

---

---

**Geographic information —  
Observations, measurements and  
samples**

*Information géographique — Observations, mesures et échantillons*

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023



STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023



**COPYRIGHT PROTECTED DOCUMENT**

© ISO 2023

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office  
CP 401 • Ch. de Blandonnet 8  
CH-1214 Vernier, Geneva  
Phone: +41 22 749 01 11  
Email: [copyright@iso.org](mailto:copyright@iso.org)  
Website: [www.iso.org](http://www.iso.org)

Published in Switzerland

# Contents

Page

<b>Foreword</b> .....	<b>ix</b>
<b>Introduction</b> .....	<b>x</b>
<b>1 Scope</b> .....	<b>1</b>
<b>2 Normative references</b> .....	<b>1</b>
<b>3 Terms and definitions</b> .....	<b>1</b>
<b>4 Document conventions</b> .....	<b>5</b>
4.1 Abbreviated terms and acronyms.....	5
4.2 Schema language.....	5
4.3 Model element names.....	6
4.4 Requirements and recommendations.....	6
4.5 Requirements classes.....	7
4.6 Conformance classes.....	7
4.7 Identifiers.....	8
4.8 Associations in UML context diagrams.....	8
<b>5 Conformance</b> .....	<b>8</b>
5.1 Overview.....	8
5.2 Conformance classes.....	9
<b>6 Packaging, requirements and dependencies</b> .....	<b>11</b>
6.1 Requirements.....	11
6.2 UML.....	12
6.2.1 UML package structure.....	12
6.2.2 UML package dependencies.....	12
6.3 Note on the use of "Any".....	14
<b>7 Fundamental characteristics of observations and samples (informative)</b> .....	<b>14</b>
7.1 Observation schema.....	14
7.1.1 Property evaluation.....	14
7.1.2 Observation.....	15
7.1.3 Properties of an Observation.....	15
7.1.4 Observation location.....	16
7.1.5 Result types.....	16
7.1.6 Use of the observation model.....	16
7.2 Sample schema.....	17
7.2.1 Role of sample features.....	17
7.2.2 Proximate vs. ultimate feature-of-interest.....	17
7.2.3 Role of samples.....	18
7.2.4 Sampling process.....	18
7.2.5 Classification of samples.....	19
7.3 Alignment between Observation, Sample and domain models.....	19
7.3.1 Model consistency.....	19
7.3.2 Relationship between Sample and domain features.....	22
<b>8 Conceptual Observation schema</b> .....	<b>25</b>
8.1 General.....	25
8.1.1 Conceptual Observation model.....	25
8.1.2 Conceptual Observation schema package Requirements Class.....	26
8.1.3 Association relatedObservation.....	26
8.2 Observation.....	27
8.2.1 Observation Requirements Class.....	27
8.2.2 Interface Observation.....	27
8.2.3 Attribute phenomenonTime.....	28
8.2.4 Attribute resultTime.....	28
8.2.5 Attribute validTime.....	28

8.2.6	Association featureOfInterest	29
8.2.7	Association observedProperty	29
8.2.8	Association result	29
8.2.9	Association observingProcedure	30
8.2.10	Association observer	30
8.2.11	Association host	30
8.2.12	Constraint Observer or Host	30
8.2.13	Constraint ObservableProperty characteristic associated with featureOfInterest	30
8.2.14	Constraint suitable ObservableProperty	30
8.2.15	Constraint suitable result type	30
8.2.16	Constraint unit of measure	31
8.3	ObservableProperty	31
8.3.1	ObservableProperty Requirements Class	31
8.3.2	Interface ObservableProperty	31
8.3.3	Association observer	32
8.4	Procedure	32
8.4.1	Procedure Requirements Class	32
8.4.2	Interface Procedure	32
8.5	ObservingProcedure	32
8.5.1	ObservingProcedure Requirements Class	32
8.5.2	Interface ObservingProcedure	32
8.5.3	Association observer	33
8.6	Observer	33
8.6.1	Observer Requirements Class	33
8.6.2	Interface Observer	33
8.6.3	Association observableProperty	34
8.6.4	Association observingProcedure	34
8.6.5	Association deployment	34
8.7	Host	34
8.7.1	Host Requirements Class	34
8.7.2	Interface Host	34
8.7.3	Association deployment	35
8.7.4	Association relatedHost	35
8.8	Deployment	35
8.8.1	Deployment Requirements Class	35
8.8.2	Interface Deployment	35
8.8.3	Association observer	35
8.8.4	Association host	35
<b>9</b>	<b>Abstract Observation Core</b>	<b>36</b>
9.1	General	36
9.1.1	Abstract Observation Core Package Requirements Class	36
9.1.2	Association metadata	36
9.2	AbstractObservationCharacteristics	36
9.2.1	AbstractObservationCharacteristics Requirements Class	36
9.2.2	Feature type AbstractObservationCharacteristics	38
9.2.3	Attribute observationType	39
9.2.4	Attribute parameter	39
9.2.5	Attribute resultQuality	39
9.2.6	Association proximateFeatureOfInterest	40
9.2.7	Association ultimateFeatureOfInterest	40
9.2.8	Association collection	41
9.3	AbstractObservation	41
9.3.1	AbstractObservation Requirements Class	41
9.3.2	Constraint observationType	42
9.3.3	Constraint resultTime instant	42
9.3.4	Constraint parameter unique name	42
9.3.5	Constraint proximate or ultimate featureOfInterest	42

9.3.6	Constraint Observer or Host.....	42
9.3.7	Constraint ObservableProperty characteristic associated with featureOfInterest.....	42
9.3.8	Constraint suitable ObservableProperty.....	42
9.3.9	Constraint suitable result type.....	42
9.4	AbstractObservableProperty.....	42
9.4.1	AbstractObservableProperty Requirements Class.....	42
9.5	AbstractObservingProcedure.....	43
9.5.1	AbstractObservingProcedure Requirements Class.....	43
9.6	AbstractObserver.....	45
9.6.1	AbstractObserver Requirements Class.....	45
9.7	AbstractHost.....	46
9.7.1	AbstractHost Requirements Class.....	46
9.8	AbstractDeployment.....	47
9.8.1	AbstractDeployment Requirements Class.....	47
9.8.2	Attribute deploymentReason.....	48
9.8.3	Attribute deploymentTime.....	49
9.9	AbstractObservationCollection.....	49
9.9.1	AbstractObservationCollection Requirements Class.....	49
9.9.2	Feature type AbstractObservationCollection.....	50
9.9.3	Attribute collectionType.....	50
9.9.4	Association member.....	51
9.9.5	Association memberCharacteristics.....	51
9.9.6	Association relatedCollection.....	51
9.10	NamedValue.....	51
9.10.1	NamedValue Requirements Class.....	51
9.10.2	Data type NamedValue.....	51
9.10.3	Attribute name.....	51
9.10.4	Attribute value.....	52
9.11	Codelists.....	52
9.11.1	AbstractObservationType.....	52
9.11.2	AbstractObservationCollectionType.....	52
<b>10</b>	<b>Basic Observations.....</b>	<b>52</b>
10.1	General.....	52
10.1.1	Basic Observations Package Requirements Class.....	52
10.1.2	Attribute link.....	53
10.1.3	Attribute location.....	53
10.2	Observation.....	53
10.2.1	Observation Requirements Class.....	53
10.3	ObservationCharacteristics.....	55
10.3.1	ObservationCharacteristics Requirements Class.....	55
10.4	ObservationCollection.....	55
10.4.1	ObservationCollection Requirements Class.....	55
10.5	ObservingCapability.....	55
10.5.1	ObservingCapability Requirements Class.....	55
10.5.2	Feature type ObservingCapability.....	57
10.6	ObservableProperty.....	58
10.6.1	ObservableProperty Requirements Class.....	58
10.7	ObservingProcedure.....	59
10.7.1	ObservingProcedure Requirements Class.....	59
10.8	Observer.....	61
10.8.1	Observer Requirements Class.....	61
10.9	Host.....	62
10.9.1	Host Requirements Class.....	62
10.10	Deployment.....	65
10.10.1	Deployment Requirements Class.....	65
10.11	GenericDomainFeature.....	66
10.11.1	GenericDomainFeature Requirements Class.....	66

10.11.2	Feature type GenericDomainFeature.....	69
10.12	Codelists.....	69
10.12.1	ObservationCollectionType.....	69
10.12.2	ObservationTypeByResultType.....	71
<b>11</b>	<b>Conceptual Sample schema.....</b>	<b>72</b>
11.1	General.....	72
11.1.1	Conceptual Sample schema model.....	72
11.1.2	Conceptual Sample Schema package Requirements Class.....	73
11.2	Sample.....	74
11.2.1	Sample Requirements Class.....	74
11.2.2	Interface Sample.....	74
11.2.3	Association sampling.....	74
11.2.4	Association preparationStep.....	75
11.2.5	Association sampledFeature.....	75
11.2.6	Association relatedSample.....	75
11.3	Sampling.....	75
11.3.1	Sampling Requirements Class.....	75
11.3.2	Interface Sampling.....	76
11.3.3	Association sample.....	76
11.3.4	Association featureOfInterest.....	76
11.3.5	Association sampler.....	76
11.3.6	Association samplingProcedure.....	77
11.3.7	Association relatedSampling.....	77
11.4	Sampler.....	77
11.4.1	Sampler Requirements Class.....	77
11.4.2	Interface Sampler.....	77
11.4.3	Association sampling.....	77
11.4.4	Association implementedProcedure.....	78
11.5	PreparationStep.....	78
11.5.1	PreparationStep Requirements Class.....	78
11.5.2	Interface PreparationStep.....	78
11.5.3	Association processingDetails.....	78
11.5.4	Association preparedSample.....	78
11.6	PreparationProcedure.....	78
11.6.1	PreparationProcedure Requirements Class.....	78
11.6.2	Interface PreparationProcedure.....	79
11.6.3	Association samplePreparationStep.....	79
11.7	SamplingProcedure.....	79
11.7.1	SamplingProcedure Requirements Class.....	79
11.7.2	Interface SamplingProcedure.....	79
11.7.3	Association sampling.....	79
11.7.4	Association sampler.....	80
<b>12</b>	<b>Abstract Sample Core.....</b>	<b>80</b>
12.1	General.....	80
12.1.1	Abstract Sample Core Package Requirements.....	80
12.2	AbstractSample.....	80
12.2.1	AbstractSample Requirements Class.....	80
12.2.2	Attribute sampleType.....	82
12.2.3	Attribute parameter.....	82
12.3	AbstractSampling.....	82
12.3.1	AbstractSampling Requirements Class.....	82
12.3.2	Attribute samplingLocation.....	83
12.3.3	Attribute time.....	83
12.3.4	Attribute parameter.....	83
12.4	AbstractSampler.....	84
12.4.1	AbstractSampler Requirements Class.....	84
12.4.2	Attribute samplerType.....	85

12.5	AbstractSamplingProcedure	86
12.5.1	AbstractSamplingProcedure Requirements Class	86
12.6	AbstractPreparationProcedure	87
12.6.1	AbstractPreparationProcedure Requirements Class	87
12.7	AbstractPreparationStep	88
12.7.1	AbstractPreparationStep Requirements Class	88
12.7.2	Attribute description	89
12.7.3	Attribute time	89
12.8	Codelists	89
12.8.1	AbstractSampleType	89
12.8.2	AbstractSamplerType	89
<b>13</b>	<b>Basic Samples</b>	<b>90</b>
13.1	General	90
13.1.1	Basic Samples Package Requirements Class	90
13.2	Sample	90
13.2.1	Sample Requirements Class	90
13.3	SpatialSample	92
13.3.1	SpatialSample Requirements Class	92
13.3.2	Feature type SpatialSample	92
13.3.3	Attribute shape	92
13.3.4	Attribute horizontalPositionalAccuracy	93
13.3.5	Attribute verticalPositionalAccuracy	93
13.4	MaterialSample	93
13.4.1	MaterialSample Requirements Class	93
13.4.2	Feature type MaterialSample	93
13.4.3	Attribute size	94
13.4.4	Attribute storageLocation	94
13.4.5	Attribute sourceLocation	94
13.5	StatisticalSample	94
13.5.1	StatisticalSample Requirements Class	94
13.5.2	Feature type StatisticalSample	95
13.5.3	Attribute classification	95
13.6	Sampling	95
13.6.1	Sampling Requirements Class	95
13.7	Sampler	96
13.7.1	Sampler Requirements Class	96
13.8	SamplingProcedure	97
13.8.1	SamplingProcedure Requirements Class	97
13.9	PreparationProcedure	99
13.9.1	PreparationProcedure Requirements Class	99
13.10	PreparationStep	100
13.10.1	PreparationStep Requirements Class	100
13.11	SampleCollection	102
13.11.1	SampleCollection Requirements Class	102
13.11.2	Feature type SampleCollection	102
13.11.3	Association member	103
13.11.4	Association relatedCollection	103
13.12	PhysicalDimension	103
13.12.1	PhysicalDimension Requirements Class	103
13.12.2	Data type PhysicalDimension	103
13.12.3	Attribute dimension	103
13.12.4	Attribute value	103
13.13	NamedLocation	104
13.13.1	NamedLocation Requirements Class	104
13.13.2	Data type NamedLocation	104
13.13.3	Attribute address	104
13.13.4	Attribute name	104
13.13.5	Attribute representativeGeometry	104

13.14	StatisticalClassification	104
13.14.1	StatisticalClassification Requirements Class	104
13.14.2	Data type StatisticalClassification	105
13.14.3	Attribute concept	105
13.14.4	Attribute classification	105
13.15	Codelists	105
13.15.1	SampleTypeByGeometryType	105
<b>Annex A</b>	<b>(normative) Abstract test suite</b>	<b>107</b>
<b>Annex B</b>	<b>(informative) Common usage of OMS concepts</b>	<b>117</b>
<b>Annex C</b>	<b>(informative) Changes in the Observation and Sample modelsbetween ISO 19156:2011 and ISO 19156:2023 (this document)</b>	<b>121</b>
<b>Annex D</b>	<b>(informative) Best practices in use of the Observation and Sampling models</b>	<b>139</b>
<b>Annex E</b>	<b>(informative) Detailed package overview diagrams</b>	<b>147</b>
<b>Bibliography</b>		<b>150</b>

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

## 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](http://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](http://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](http://www.iso.org/iso/foreword.html).

This document was prepared by Technical Committee ISO/TC 211, *Geographic information/Geomatics*, in collaboration with the European Committee for Standardization (CEN) Technical Committee CEN/TC 287, *Geographic Information*, in accordance with the Agreement on technical cooperation between ISO and CEN (Vienna Agreement), and in collaboration with the Open Geospatial Consortium (OGC).

This second edition cancels and replaces the first edition (ISO 19156:2011), which has been technically revised.

The main changes are as follows:

- the UML model and 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 ISO 19156:2011, but certain carefully designed improvements and clarifications for the intended use have been included;
- the Sample model has been refined: given the integral nature of the Sample model, it has been decided to include that term in the name of the document;
- [Annex C](#) has been added listing the changes between ISO 19156:2011 and this document.

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](http://www.iso.org/members.html).

## Introduction

This document 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 W3C Spatial Data on the Web Best Practices.<sup>[31]</sup> This includes several new standards for describing and publishing sensors and observations, such as the OGC SensorThings API<sup>[22]</sup> and the W3C/OGC Semantic Sensor Network Ontology.<sup>[28]</sup> This second edition of ISO 19156 (now named "Observations, Measurements and Samples", or abbreviated to "OMS") is informed by these recent developments. The focus of revising ISO 19156:2011 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 in this document is derived from the previous edition published by Open Geospatial Consortium as OGC 10-004r3, and also ISO 19156:2011. A technical note describing the changes in comparison to ISO 19156:2011 is provided in [Annex C](#).

The name and contact information of the maintenance agency for this document can be found at [www.iso.org/maintenance\\_agencies](http://www.iso.org/maintenance_agencies).

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023(E)

# Geographic information — Observations, measurements and samples

## 1 Scope

This document 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 document 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 document 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 19103, *Geographic information — Conceptual schema language*

ISO 19107, *Geographic information — Spatial schema*

ISO 19108, *Geographic information — Temporal schema*

## 3 Terms and definitions

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

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

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

— IEC Electropedia: available at <https://www.electropedia.org/>

### 3.1 application schema

conceptual schema for data required by one or more applications

[SOURCE: ISO 19101-1:2014, 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

[SOURCE: ISO 19123-1:—,<sup>1)</sup> 3.1.8]

1) Under preparation. Stage at the time of publication: ISO/FDIS 19123-1:2023.

### 3.3

#### **data type**

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

EXAMPLE Integer, Real, Boolean, String and Date.

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

[SOURCE: ISO 19103:2015, 4.14]

### 3.4

#### **domain**

well-defined set

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

[SOURCE: ISO 19109:2015, 4.8, modified — Original Note 1 to entry has been replaced with a new Note 1 to entry.]

### 3.5

#### **domain feature**

feature of a type defined within a particular application domain

Note 1 to entry: This can be contrasted with observations and sampling features, which are features of types defined for cross-domain purposes.

### 3.6

#### **ex situ**

#### **off-site**

referring to the study, maintenance or conservation of a specimen or population away from its natural surroundings

Note 1 to entry: Opposite of in situ (on-site).

Note 2 to entry: An example of ex situ and direct is measuring a patient's temperature with a mercury thermometer in a blood sample.

Note 3 to entry: An example of ex situ and remote is measuring a patient's temperature with an infra-red thermometer pointed at the blood sample.

### 3.7

#### **feature**

abstraction of real-world phenomena

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

[SOURCE: ISO 19101-1:2014, 4.1.11, modified — Note 1 to entry has been modified.]

### 3.8

#### **feature-of-interest**

subject of the observation

### 3.9

#### **feature type**

class of features having common characteristics

**3.10**  
**in situ**  
**on-site**

referring to the study, maintenance or conservation of a specimen or population without removing it from its natural surroundings

Note 1 to entry: Opposite of ex situ (off-site).

Note 2 to entry: An example of in situ and direct is measuring a patient's temperature with a mercury thermometer in the patient's rectum.

Note 3 to entry: An example of in situ and remote is measuring a patient's temperature with an infra-red thermometer at a distance.

**3.11**  
**measure**

<GML> value described using a numeric amount with a scale or using a scalar reference system

Note 1 to entry: When used as a noun, measure is a synonym for physical quantity.

[SOURCE: ISO 19136-1:2020, 3.1.41]

**3.12**  
**measurement**

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

[SOURCE: ISO 19101-2:2018, 3.21]

**3.13**  
**observation**

act carried out by an observer to determine the value of an observable property of an object (feature-of-interest) by using a procedure, with the value provided as the result

**3.14**  
**observation result**

estimate of the value of a property determined through a known observation procedure

**3.15**  
**observer**

identifiable entity that can generate observations pertaining to an observable property by implementing a procedure

Note 1 to entry: An observer is an instance of a sensor, instrument, implementation of an algorithm or a being such as a person.

**3.16**  
**procedure**

specified way to carry out an activity or a process

[SOURCE: ISO 9000:2015, 3.4.5, modified — Note 1 to entry has been deleted.]

**3.17**  
**process**

set of interrelated or interacting activities that use inputs to deliver an intended result

[SOURCE: ISO 9000:2015, 3.4.1, modified — Notes 1-6 have been deleted.]

**3.18**  
**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.

Note 1 to entry: In some communities, the observed property is referred to as the measurand.

[SOURCE: ISO 19143:2010, 4.21, modified — Example and note have been added to the entry.]

### 3.19

#### **property type**

characteristic of a feature type

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

### 3.20

#### **proximate feature-of-interest**

entity that is directly of interest in the act of observing

Note 1 to entry: This is a specialized form of the feature-of-interest.

### 3.21

#### **range**

<coverage> 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 ISO 19107.

[SOURCE: ISO 19123-1:—, 3.1.47]

### 3.22

#### **sample**

object that is representative of a concept, real-world object or phenomenon

### 3.23

#### **sampler**

device or entity (including humans) that is used by, or implements, a sampling procedure to create or transform one or more sample(s)

### 3.24

#### **sensor**

element of a measuring system that is directly affected by a phenomenon, body, or substance carrying a quantity to be measured

[SOURCE: JCGM 200:2012, 3.8, modified — EXAMPLES and NOTE deleted.]

### 3.25

#### **ultimate feature-of-interest**

entity that is ultimately of interest in the act of observing

Note 1 to entry: This is a specialized form of the feature-of-interest.

### 3.26

#### **unit of measure**

reference quantity chosen from a unit equivalence group

Note 1 to entry: In positioning services, the usual units of measurement are either angular units or linear units. Implementations of positioning services shall clearly distinguish between SI units and non-SI units. When non-SI units are employed, it is required that their relation to SI units be specified.

[SOURCE: ISO 19116:2019, 3.29]

### 3.27 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.

[SOURCE: ISO/IEC 19501:2005, 0000\_5, modified — Notes 1- 3 to entry have been added.]

## 4 Document conventions

### 4.1 Abbreviated terms and acronyms

CIS	Coverage Implementation Schema
EO	Earth observation
GFM	General Feature Model
GML	Geography Markup Language
INSPIRE	Infrastructure for Spatial Information in Europe
O&M	Observations and measurements
OMS	Observations, measurements and samples (this document)
OGC	Open Geospatial Consortium
RDA	Research Data Alliance
SensorML	OGC Sensor Model Language
SOS	OGC Sensor Observation Service
STA	OGC SensorThings API
SWE	OGC Sensor Web Enablement
UML	Unified Modeling Language
UoM	unit of measure
URI	Uniform Resource Identifier
XML	Extensible Markup Language
2-D	two-dimensional
3-D	three-dimensional

### 4.2 Schema language

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

The UML in Abstract Core and Basic packages is conformant with the profile described in ISO 19136-1:2020, Annex E. Use of this restricted idiom supports direct transformation into a GML Application Schema. The stereotype «FeatureType» states that a class is an instance of the «metaclass» 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.

### 4.3 Model element names

This document 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 be applied 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 document to common terminology in related application domains is provided in [Annex B](#).

### 4.4 Requirements and recommendations

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

<b>Requirement /req/{pkg}/{classM}/{reqN}</b>	[Normative statement]
---	-----------------------

where **/req/{pkg}/{classM}/{reqN}** identifies the requirement. The use of this layout convention allows the normative provisions of this document to be easily located by implementers.

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

The following base (/req/{pkg}/) 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.

In the lines below, the base (/req/{pkg}/) has been left out for better readability.

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 attributes or associations, the following syntax is used.

- **{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 apply to cardinality 0..\*.

- **{Attribute/Association Name}-type**: Type information pertaining to the attribute or association when the type is constrained within one model package.
- **{Attribute/Association Name}-card**: Cardinality information pertaining to the attribute or association, when the cardinality is constrained within one model package.
- **{Attribute/Association Name}-con**: Additional constraints. As these sometimes pertain to multiple attributes or associations, this part of the name can become more complex.

Individual requirements are case-sensitive, following UML naming conventions. Requirements pertaining to classes contain the class name in UpperCamelCase. Requirements pertaining to attributes or associations utilize the attribute name or association role name in lowerCamelCase.

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

<b>Recommendation</b> /rec/{pkg}/{classM}/{recO}	[Informative statement]
--	-------------------------

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

#### 4.5 Requirements classes

Each statement (requirement or recommendation) in this document 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:

<b>Requirements class</b>	<b>/req/{pkg}/{classM}</b>
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 shall be satisfied. Hence, the requirements class is the unit of re-use and dependency.

Dependency to another requirements 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.

#### 4.6 Conformance classes

Conformance to this document is possible at a number of levels, specified by conformance classes in accordance with [Annex A](#). Each conformance class is summarized using the following template:

<b>Conformance class</b>	<b>/conf/{pkg}/{classM}</b>
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 shall be passed. Each conformance class tests conformance to a set of requirements packaged in a requirements class.

### 4.7 Identifiers

Each requirements class, requirement and recommendation is 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 Uniform Resource Identifier (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:

<http://www.opengis.net/spec/om/3.0/req/pkg/classM>

The URI for each requirement has the form:

<http://www.opengis.net/spec/om/3.0/req/pkg/classM/reqN>

The URI for each recommendation has the form:

<http://www.opengis.net/spec/om/3.0/rec/pkg/classM/recO>

The URI for each conformance class has the form:

<http://www.opengis.net/spec/om/3.0/conf/pkg/classM>

### 4.8 Associations in UML context diagrams

The UML model described in this document is rather complex. To keep the text size readable in the UML, context diagrams of this document only display certain associations of each class. Please refer to the context diagram of a particular class to see all associations of that class. All associations of the classes in each package are also shown in the detailed package overview diagrams in [Annex E](#).

## 5 Conformance

### 5.1 Overview

[Clauses 8](#) to [13](#) of this document use 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 document.

The UML model in this document 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.

## 5.2 Conformance classes

The conformance rules for Models in general are described in ISO 19109. Application Schemas also claiming conformance to this document shall also conform to the rules specified in [Clauses 8](#) 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, measurements and samples (OMS) conformance. [Table 1](#), [Table 2](#), [Table 3](#), [Table 4](#), [Table 5](#) and [Table 6](#) list all of these classes by package, including their relative identifiers and the corresponding subclauses of the Abstract Test Suite. The full URIs of the conformance classes are formed by prefixing the relative URI path as described in [4.7](#).

**Table 1 — Conceptual Observation schema conformance classes**

Conformance class	Identifier	Subclause
Conceptual Observation schema package	/conf/obs-cpt	<a href="#">A.1.1</a>
Conceptual Observation – Deployment	/conf/obs-cpt/Deployment	<a href="#">A.1.2</a>
Conceptual Observation – Host	/conf/obs-cpt/Host	<a href="#">A.1.3</a>
Conceptual Observation – ObservableProperty	/conf/obs-cpt/ObservableProperty	<a href="#">A.1.4</a>
Conceptual Observation – Observation	/conf/obs-cpt/Observation	<a href="#">A.1.5</a>
Conceptual Observation – Observer	/conf/obs-cpt/Observer	<a href="#">A.1.6</a>
Conceptual Observation – ObservingProcedure	/conf/obs-cpt/ObservingProcedure	<a href="#">A.1.7</a>
Conceptual Observation – Procedure	/conf/obs-cpt/Procedure	<a href="#">A.1.8</a>

**Table 2 — Abstract Observation Core conformance classes**

Conformance class	Identifier	Subclause
Abstract Observation Core package	/conf/obs-core	<a href="#">A.2.1</a>
Abstract Observation Core – AbstractDeployment	/conf/obs-core/AbstractDeployment	<a href="#">A.2.2</a>
Abstract Observation Core – AbstractHost	/conf/obs-core/AbstractHost	<a href="#">A.2.3</a>
Abstract Observation Core – AbstractObservableProperty	/conf/obs-core/AbstractObservableProperty	<a href="#">A.2.4</a>
Abstract Observation Core – AbstractObservation	/conf/obs-core/AbstractObservation	<a href="#">A.2.5</a>
Abstract Observation Core – AbstractObservationCharacteristics	/conf/obs-core/AbstractObservationCharacteristics	<a href="#">A.2.6</a>
Abstract Observation Core – AbstractObserver	/conf/obs-core/AbstractObserver	<a href="#">A.2.7</a>
Abstract Observation Core – AbstractObservingProcedure	/conf/obs-core/AbstractObservingProcedure	<a href="#">A.2.8</a>
Abstract Observation Core – NamedValue	/conf/obs-core/NamedValue	<a href="#">A.2.9</a>
Abstract Observation Core – AbstractObservationCollection	/conf/obs-core/AbstractObservationCollection	<a href="#">A.2.10</a>

**Table 3 — Basic Observations conformance classes**

Conformance class	Identifier	Subclause
Basic Observations package	/conf/obs-basic	<a href="#">A.3.1</a>
Basic Observations – Deployment	/conf/obs-basic/Deployment	<a href="#">A.3.2</a>

**Table 3 (continued)**

Conformance class	Identifier	Subclause
Basic Observations – GenericDomainFeature	/conf/obs-basic/GenericDomainFeature	<a href="#">A.3.3</a>
Basic Observations – Host	/conf/obs-basic/Host	<a href="#">A.3.4</a>
Basic Observations – ObservableProperty	/conf/obs-basic/ObservableProperty	<a href="#">A.3.5</a>
Basic Observations – Observation	/conf/obs-basic/Observation	<a href="#">A.3.6</a>
Basic Observations – ObservationCharacteristics	/conf/obs-basic/ObservationCharacteristics	<a href="#">A.3.7</a>
Basic Observations – ObservationCollection	/conf/obs-basic/ObservationCollection	<a href="#">A.3.8</a>
Basic Observations – Observer	/conf/obs-basic/Observer	<a href="#">A.3.9</a>
Basic Observations – ObservingCapability	/conf/obs-basic/ObservingCapability	<a href="#">A.3.10</a>
Basic Observations – ObservingProcedure	/conf/obs-basic/ObservingProcedure	<a href="#">A.3.11</a>

**Table 4 — Conceptual Sample schema conformance classes**

Conformance class	Identifier	Subclause
Conceptual Sample schema package	/conf/sam-cpt	<a href="#">A.4.1</a>
Conceptual Sample – PreparationProcedure	/conf/sam-cpt/PreparationProcedure	<a href="#">A.4.2</a>
Conceptual Sample – PreparationStep	/conf/sam-cpt/PreparationStep	<a href="#">A.4.3</a>
Conceptual Sample – Sample	/conf/sam-cpt/Sample	<a href="#">A.4.4</a>
Conceptual Sample – Sampler	/conf/sam-cpt/Sampler	<a href="#">A.4.5</a>
Conceptual Sample – Sampling	/conf/sam-cpt/Sampling	<a href="#">A.4.6</a>
Conceptual Sample – SamplingProcedure	/conf/sam-cpt/SamplingProcedure	<a href="#">A.4.7</a>

**Table 5 — Abstract Sample Core conformance classes**

Conformance class	Identifier	Subclause
Abstract Sample Core package	/conf/sam-core	<a href="#">A.5.1</a>
Abstract Sample Core – AbstractPreparationProcedure	/conf/sam-core/AbstractPreparationProcedure	<a href="#">A.5.2</a>
Abstract Sample Core – AbstractPreparationStep	/conf/sam-core/AbstractPreparationStep	<a href="#">A.5.3</a>
Abstract Sample Core – AbstractSample	/conf/sam-core/AbstractSample	<a href="#">A.5.4</a>
Abstract Sample Core – AbstractSampler	/conf/sam-core/AbstractSampler	<a href="#">A.5.5</a>
Abstract Sample Core – AbstractSampling	/conf/sam-core/AbstractSampling	<a href="#">A.5.6</a>
Abstract Sample Core – AbstractSamplingProcedure	/conf/sam-core/AbstractSamplingProcedure	<a href="#">A.5.7</a>

**Table 6 — Basic Samples conformance classes**

Conformance class	Identifier	Subclause
Basic Samples package	/conf/sam-basic	<a href="#">A.6.1</a>
Basic Samples – MaterialSample	/conf/sam-basic/MaterialSample	<a href="#">A.6.2</a>
Basic Samples – NamedLocation	/conf/sam-basic/NamedLocation	<a href="#">A.6.3</a>
Basic Samples – PhysicalDimension	/conf/sam-basic/PhysicalDimension	<a href="#">A.6.4</a>
Basic Samples – Sample	/conf/sam-basic/Sample	<a href="#">A.6.5</a>
Basic Samples – SampleCollection	/conf/sam-basic/SampleCollection	<a href="#">A.6.6</a>
Basic Samples – Sampler	/conf/sam-basic/Sampler	<a href="#">A.6.7</a>
Basic Samples – Sampling	/conf/sam-basic/Sampling	<a href="#">A.6.8</a>
Basic Samples – SpatialSample	/conf/sam-basic/SpatialSample	<a href="#">A.6.9</a>

Table 6 (continued)

Conformance class	Identifier	Subclause
Basic Samples – StatisticalClassification	/conf/sam-basic/StatisticalClassification	<a href="#">A.6.10</a>
Basic Samples – StatisticalSample	/conf/sam-basic/StatisticalSample	<a href="#">A.6.11</a>

## 6 Packaging, requirements and dependencies

### 6.1 Requirements

As OMS implementations often seamlessly integrate with existing data ecosystems, a very flexible requirements and conformance structure is defined. This structure enables users to selectively mix and match elements as required for their purposes from the OMS 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** Some providers only serve information on Observable Properties or Monitoring Facilities, while relying on other partners to provide information on measurement procedures. These can 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<sup>[25]</sup> 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.

- a) For each UML Package, both a Requirements Class as well as a conformance class have been defined.
- b) Additional Requirements Classes have been created for each Class appearing in the data model. Conformance Classes are added accordingly to enable grouping of the former and support references to either a group or an individual Requirement Class depending on the need.
- c) Thematic domains can 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, stipulating that all aspects of an information system conform explicitly to the same underlying standards becomes increasingly difficult. 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 Observations and Measurements (O&M) Model, the observable properties concept was only included as a metaclass, with the assumption that a reference to an existing code list will be provided. Within the current OMS Model, the observable property has been upgraded to a featureType. This is because 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 (I-ADOPT). The same rationale can be applied to most concepts from the OMS Model.

In order to expose this flexibility beyond the package structure described above, a fine-grained hierarchical requirements class structure was 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.

## 6.2 UML

### 6.2.1 UML package structure

OMS 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.

- a) **Conceptual:** Within the Conceptual Model Packages, only Interfaces are provided. These models provide a very abstract view of the individual concepts they contain without reference to specific implementations. This approach allows for the inclusion of semantically aligned objects from external sources, which, although they have not been created under the OMS model, do provide concepts sharing the same semantic meaning as the concepts from the Conceptual models.
- b) **Abstract Core:** Within the Abstract Core Model Packages, only abstract featureTypes are provided following the semantic structure of the Conceptual model (i.e. realizing the interfaces provided by the Conceptual Model Packages). A consistent approach to metadata provision is introduced. All associations from the abstract featureTypes reference the conceptual interfaces for greater implementation flexibility. The Abstract Core Model Packages are foreseen for the creation of domain models providing an Abstract Core ready for Extension.
- c) **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.

### 6.2.2 UML package dependencies

Some model elements used in the schema are defined in other International Standards. [Table 7](#) lists the dependencies between the UML packages defined in this document and other International Standards, and [Figure 1](#) shows the dependencies of the entire OMS UML model package to the other International Standards in a graphical form.

**Table 7 — UML package level dependencies**

OMS Package	Package	International Standard	Classes
Conceptual Observation schema	Any type	ISO 19103:2015	Any
Conceptual Observation schema	Temporal Objects	ISO 19108:2002	TM_Object
Conceptual Observation schema	Name types	ISO 19103:2015	GenericName
Abstract Observation Core	Conceptual Observation schema	ISO 19156:2023 (this document)	TM_Instant, TM_Period via the Temporal Objects dependency
Abstract Observation Core	General Feature Model	ISO 19109:2015	Feature concepts
Abstract Observation Core	Text	ISO 19103:2015	CharacterString
Basic Observations	Abstract Observation Core	ISO 19156:2023 (this document)	
Basic Observations	Web environment	ISO 19103:2015	URI
Basic Observations	Geometry	ISO 19107:2019	Geometry

Table 7 (continued)

OMS Package	Package	International Standard	Classes
Conceptual Sample schema	Any type	ISO 19103:2015	Any
Conceptual Sample schema	Temporal Objects	ISO 19108:2002	TM_Object
Conceptual Sample schema	Name types	ISO 19103:2015	GenericName
Conceptual Sample schema	Conceptual Observation schema	ISO 19156:2023 (this document)	Observation, Procedure
Abstract Sample Core	Conceptual Sample schema	ISO 19156:2023 (this document)	
Abstract Sample Core	General Feature Model	ISO 19109:2015	Feature concepts
Abstract Sample Core	Geometry	ISO 19107:2019	Geometry
Abstract Sample Core	Text	ISO 19103:2015	CharacterString
Abstract Sample Core	Abstract Observation Core	ISO 19156:2023 (this document)	NamedValue
Basic Samples	Abstract Sample core	ISO 19156:2023 (this document)	
Basic Samples	Web environment	ISO 19103:2015	URI
Basic Samples	Measure types	ISO 19103:2015	Measure

pkg Figure 1: External package dependencies

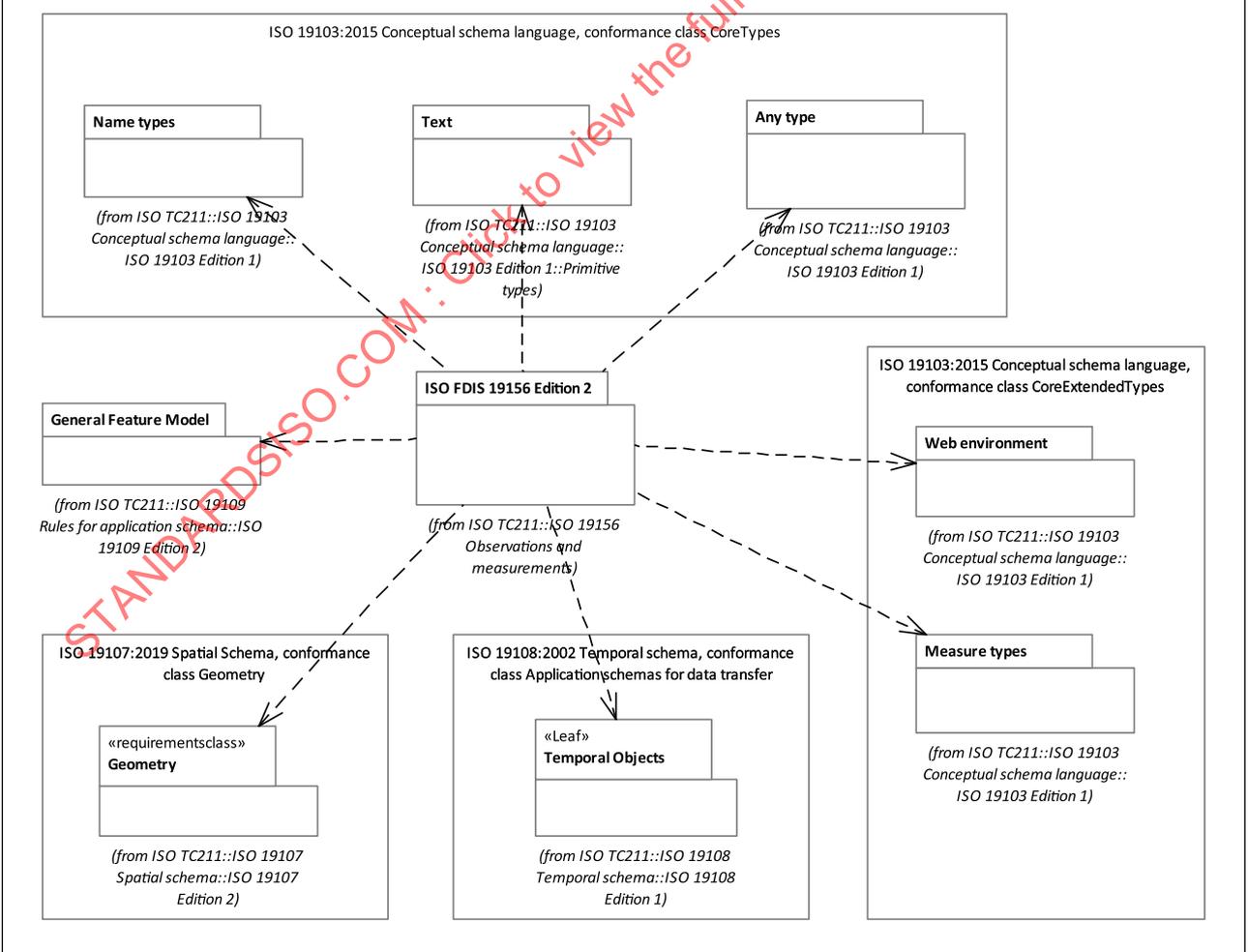


Figure 1 — External UML package dependencies

### 6.3 Note on the use of "Any"

The UML models defined in this document make extensive use of the Any interface defined in the 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 a digital representation of an appropriate concept. If 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

EXAMPLE 1 Reference to SWEET Ontology: <http://sweetontology.net/realmAtmoBoundaryLayer#planetaryboundarylayer>

EXAMPLE 2 Reference to SensorThings deployment: [https://lubw-frost.docker01.ilt-dmz.iosb.fraunhofer.de/v1.1/Locations\(269\)](https://lubw-frost.docker01.ilt-dmz.iosb.fraunhofer.de/v1.1/Locations(269))

EXAMPLE 3 Reference to the ISO 19115 series Metadata: [https://inspire-geoportal.ec.europa.eu/resources/INSPIRE-61494ff5-6fad-11e8-b649-52540023a883\\_20210415-080302/services/1/PullResults/701-750/43\\_iso19139.xml](https://inspire-geoportal.ec.europa.eu/resources/INSPIRE-61494ff5-6fad-11e8-b649-52540023a883_20210415-080302/services/1/PullResults/701-750/43_iso19139.xml)

EXAMPLE 4 Reference to an instance of borehole: <https://data.geoscience.fr/id/borehole/BSS001REWW>

EXAMPLE 5 Reference to a hydrostation: <https://iddata.eaufrance.fr/id/HydroStation/Y251002001>

EXAMPLE 6 Reference to a river segment: <https://iddata.eaufrance.fr/id/WatercourseLinkSequence/A0080300>

EXAMPLE 7 An (embedded) Boolean value as Result.

EXAMPLE 8 An (embedded) SWE DataRecord.

EXAMPLE 9 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](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)

EXAMPLE 10 OMS MaterialSample -> Reference to a rock sample: [https://www.geodata.rocks/Samples/SD-5054\\_1\\_A\\_564\\_7WR\\_20-40](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.

- 1) Value (e.g. name, price, legal boundary) assigned by some authority. These are exact.
- 2) 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 characteristic. This act 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 a given feature; an observation is a property-value-provider for a feature-of-interest. Use of a common model allows observation data using different procedures to be combined unambiguously.

In conventional measurement theory (e.g. References [11], [14], [15], [16], [18], [23]) the term “measurement” is used. However, a distinction between measurement and category-observation has been adopted in more recent work [12],[17],[24] 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 commonly provided via catalogue services (e.g. the ISO 19115 series or another community agreed one).

### 7.1.3 Properties of an Observation

An observation results in a value being assigned to a characteristic. The characteristic 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 [11],[12] but may be a process chain, human observer, an algorithm, a computation or a simulator. The key idea is that the observation result is an estimate of the value of some quality (property, characteristic) of the feature-of-interest, and the other properties of the observation provide context or metadata to support evaluation, interpretation and use of the result. Figure 2 (based on representation work done under INSPIRE [29]) provides a visual overview of this.

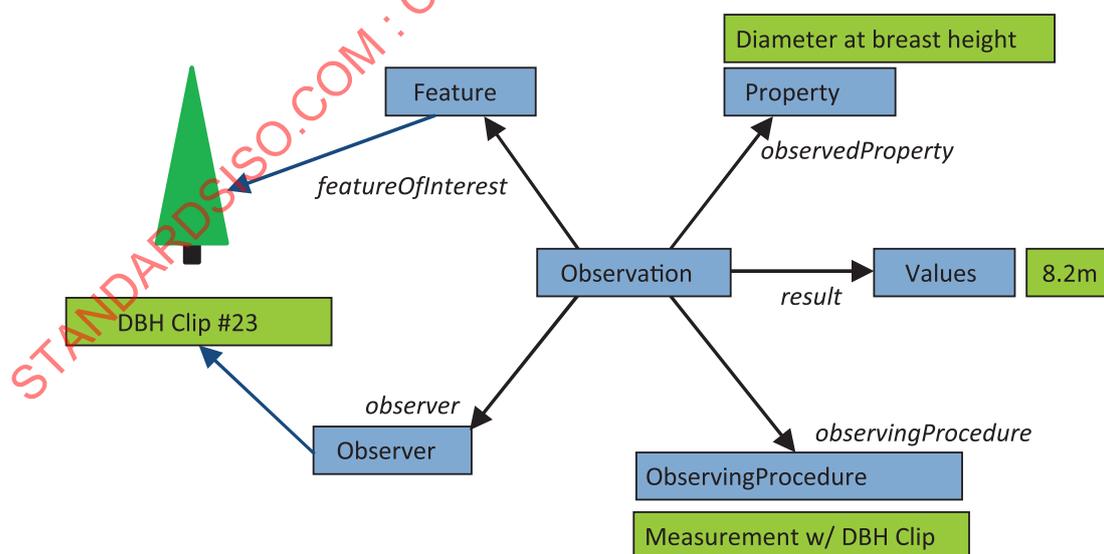


Figure 2 — Properties of an Observation

The relationship between the properties of an observation and those of its feature-of-interest is key to the semantics of the data model elaborated in this document. This is further elaborated in [Clause D.3](#).

#### 7.1.4 Observation location

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

However, the location of the feature-of-interest can potentially not be readily 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 can 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. The proximate feature-of-interest is the object upon which the measurement is directly performed, whereas the ultimate feature-of-interest is the object for which this measurement is ultimately seen as representative of.

Other locations may be relevant 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 such as 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 (8.2.3; 8.2.4). This is due to 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 may also have 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 reality, the result will typically be sampled discretely on the domain and may be represented as a discrete coverage.

Building on this, specialized observation types can be defined by communities to describe the type of result provided, expressed using a terminology common to that community.

#### 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 a provider-centric viewpoint.

The digital representation of 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 19115-1).

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 feature-of-interest. In this case the sampled-feature association of the sample would point upwards in the chain of sampled features leading to the ultimate feature-of-interest of the observation. The sample may also be associated with observations, both those being made directly on the sample as well as observations on other samples.

### 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 is not always 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

In some cases, for various reasons, the domain feature is not 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. In contrast to the well in the EXAMPLE 1, the flight-line does not represent a real-world object.

**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. Again, a virtual feature serves as proximate feature-of-interest.

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 can be utilized as a proxy for the sun's intensity.

In some cases, 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 feature-of-interest of this well is an aquifer. However, the final target of the observation is the fluid body 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 can potentially 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 can be quantified as a voltage, whereas the observed property represented by this voltage is the physiochemical value being observed by the sensor (e.g. 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 can 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 samples

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" and "station" are encountered. These usually correspond to an identifiable locality where a monitoring facility (host, platform, etc.) has been established, or sensors or other measurement devices (observers) 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, and so forth that are potentially described by the Host.

**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. Examples include biodiversity studies, crop seed preservation, etc. In those cases, the material samples considered are called specimen. It is for this reason that the class named "SE Specimen" in the previous edition of this document has been renamed "MaterialSample" in this edition.

**EXAMPLE 3** Statistical samples usually apply to populations or other sets, of which a certain subset can be of specific interest.

**NOTE** A transient spatial sample, such as a ship-track or flight-line, can 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 is not known when observations are made using the sample.

### 7.2.4 Sampling process

Understanding the process by which samples are obtained is often essential to understanding the context of subsequent measurements on this object (feature-of-interest). 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.

As a dependence on the sampling strategy, a sampling procedure appropriate to the sampling act to be performed needs to 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 act. Multiple sampling procedures can also be required for the case where one sampling process classifies samples in accordance with multiple criteria.

**EXAMPLE 1** When performing observations on populations, these can potentially 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 can involve very different samples, whereby some of these samples can serve purely to provide contextual information pertaining to the sampling event.

**EXAMPLE 2** When sampling water from a river, information on the meteorology at the time of sampling can be relevant for the interpretation of measurements obtained on the water sample.

### 7.2.5 Classification of samples

A small number of common sampling patterns, similar across domains, provide a basis for processing and portrayal tools, and depend particularly on the geometry of the sample design. 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 can 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.

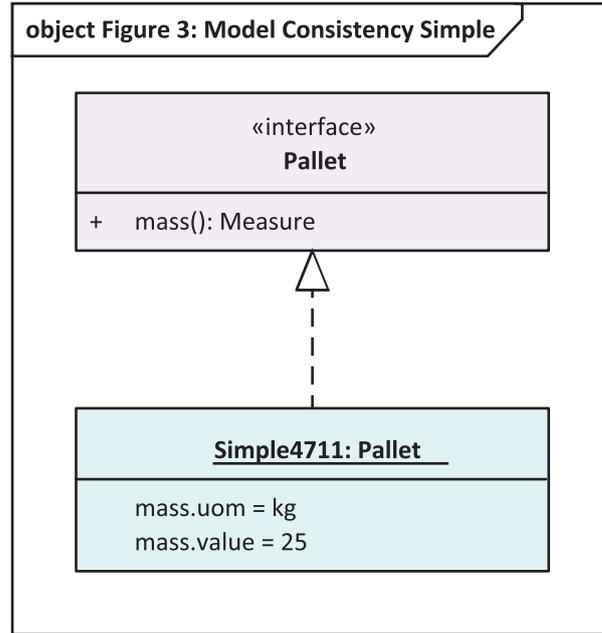
Additional information on provenance, curation and methods of archiving a sample has been delegated to external standards that may be referenced via the 'metadata' association that can be provided for all types contained within the Sampling model.

## 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 3](#)).

**EXAMPLE** A pallet with the characteristic mass is to be described via a feature model. In the simplest form, an interface "Pallet" can be defined as having the attribute "mass" of type "Measure" describing the mass characteristic of the pallet being described ([Figure 3](#)). However, when using this direct approach, no further measurement metadata is available; only the numeric mass is provided together with the unit of measurement.

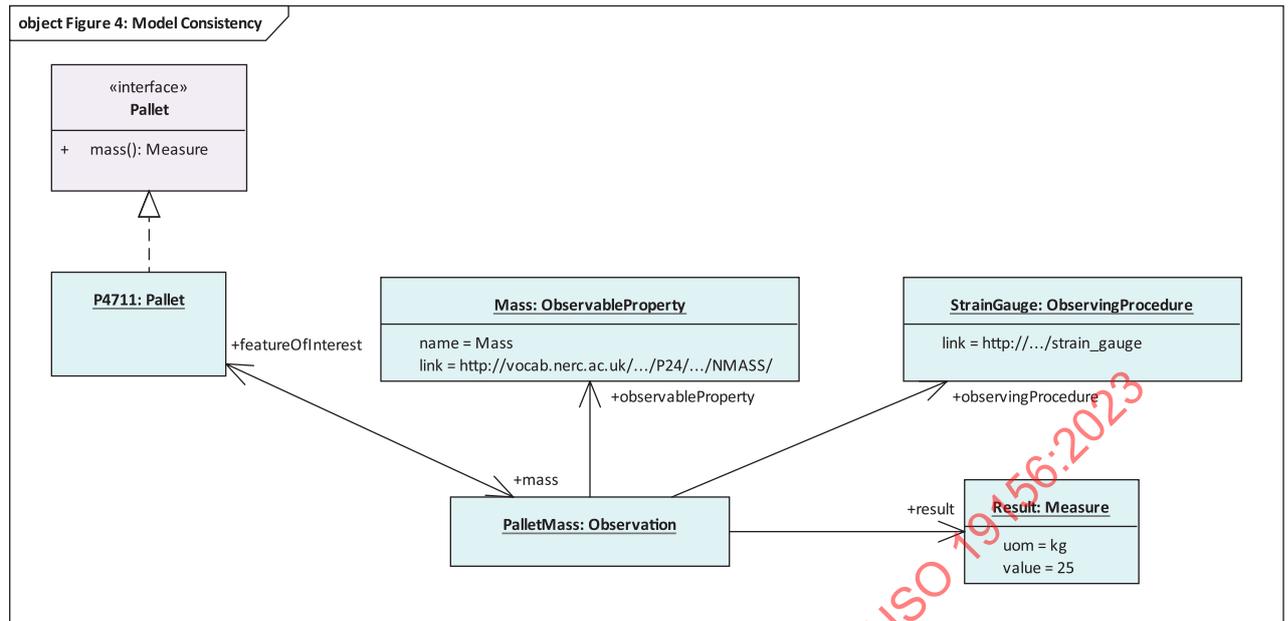


NOTE Simple example for model consistency.

**Figure 3 — (Example) Pallet interface**

Alternatively, through utilization of the OMS 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.

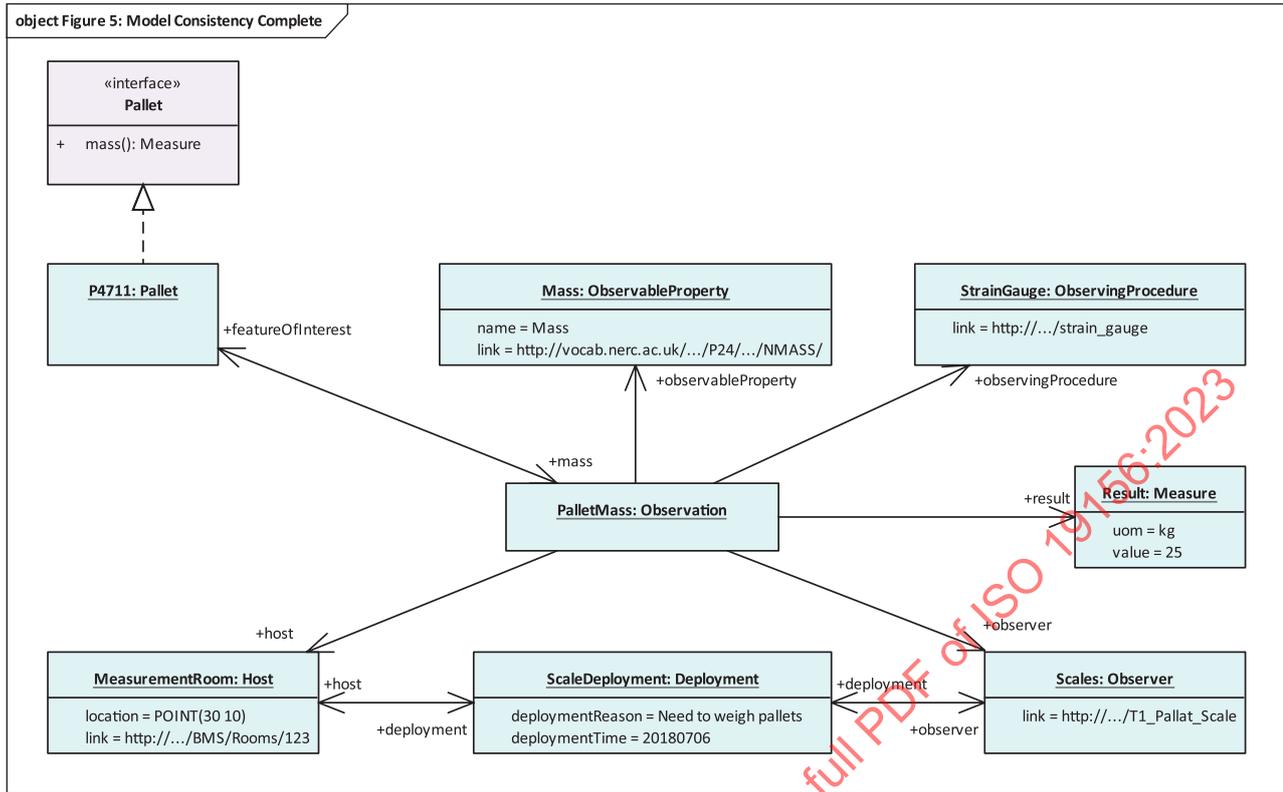
For this purpose, the observation shall have observedProperty "mass", the result shall be of the type "Measure" and the scale (unit of measure) shall be suitable for mass measurements. Thus, the requirements ensuing from the Pallet Interface are fulfilled, while additional relevant measurement meta-information is also provided; model consistency has been ensured. This approach is illustrated in [Figure 4](#).



NOTE The observed property (mass) is a characteristic associated with the type of the feature-of-interest (Pallet) and the procedure and result type are also suitable.

**Figure 4 — (Example) An observation with consistent properties**

[Figure 5](#) shows a complete representation of a mass observation. In addition to the basic information provided with the observation in [Figure 4](#), information on the specific measurement device used is provided together with information on where this device was deployed as the observation was performed.



NOTE For additional context, the Observer, Host and Deployment have been added.

Figure 5 — (Example) An observation with complete properties

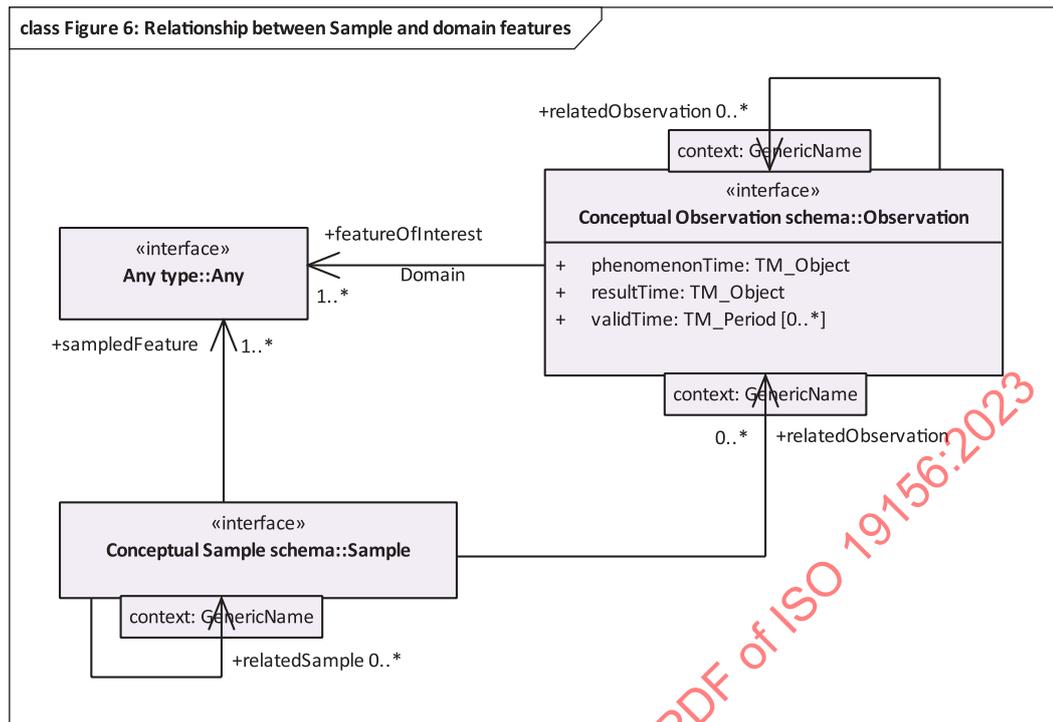
An attribute from within the conceptual model can be instantiated as an Observation in the concrete realization. The attributes that have been defined for the domain feature within the interface, in the example “mass” and “uom”, can be realized through the association of an Observation carrying this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

Based on the use case, when modelling, it is necessary to decide whether solely providing information of type "Measure" with UoM is sufficient for the domain considered. In some cases, the full OMS 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, etc.) 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 “sampledFeature” links the Sample feature to the feature which this Sample feature was designed to sample. The target of this association 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 feature and the domain feature can potentially 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 OMS model can interact with a domain model.

In this example, Well, Aquifer and FluidBody are modelled outside the OMS model (in OGC:GWML2 respectively under GW\_Well, GW\_Aquifer and GW\_FluidBody) 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. The Well is thus considered as the proximateFeatureOfInterest of the Observation. The sampledFeature (the aquifer) of the Well is the ultimateFeatureOfInterest.

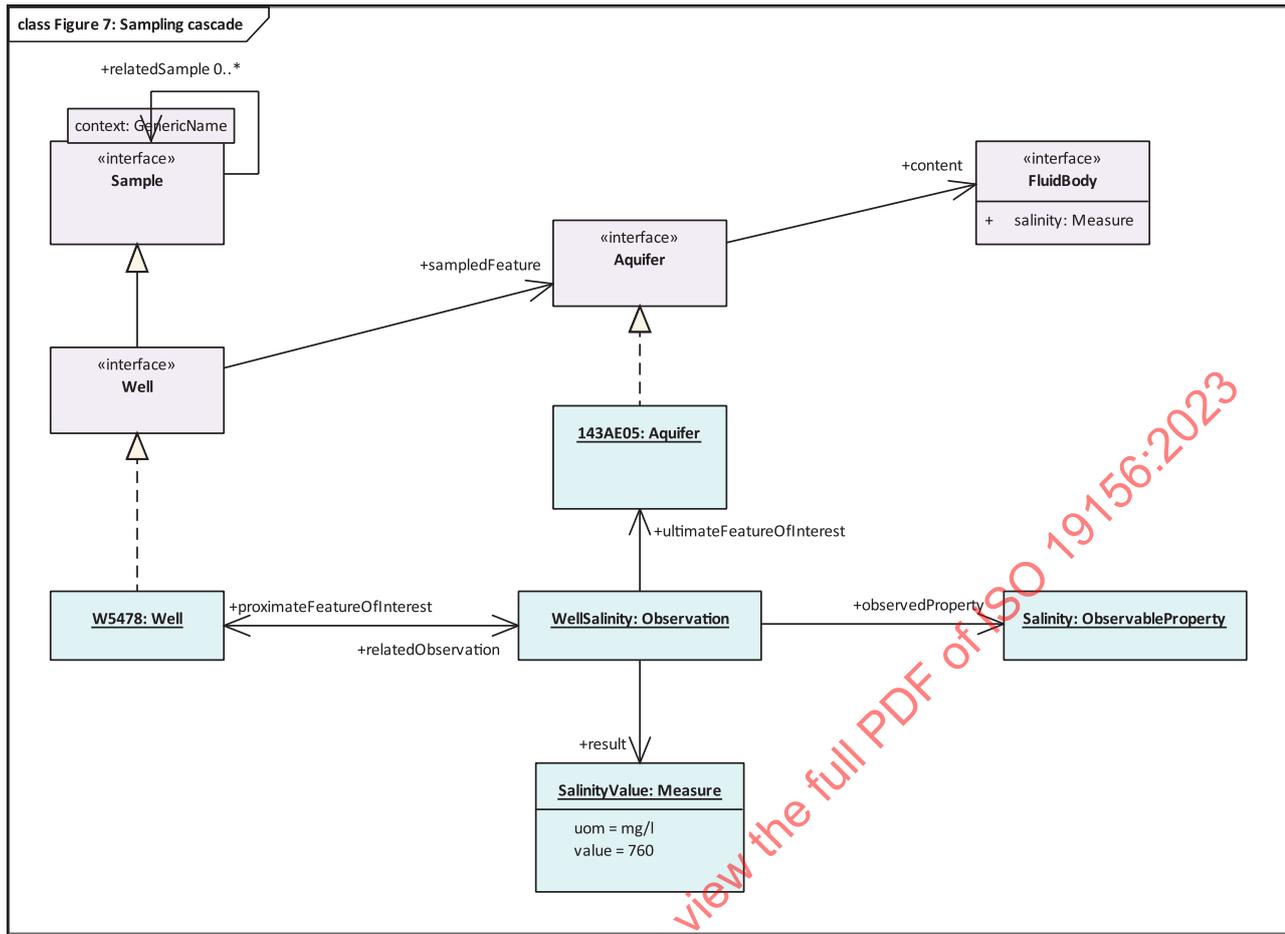


Figure 7 — (Example) Sampling Cascade example including domain features

Depending on the use case, it is advisable to push the modelling choice a step further and instantiate a `FluidBody` in the system according to the semantics of the domain model (`Well`, `Aquifer`, `FluidBody`). The example is further refined accordingly 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`.

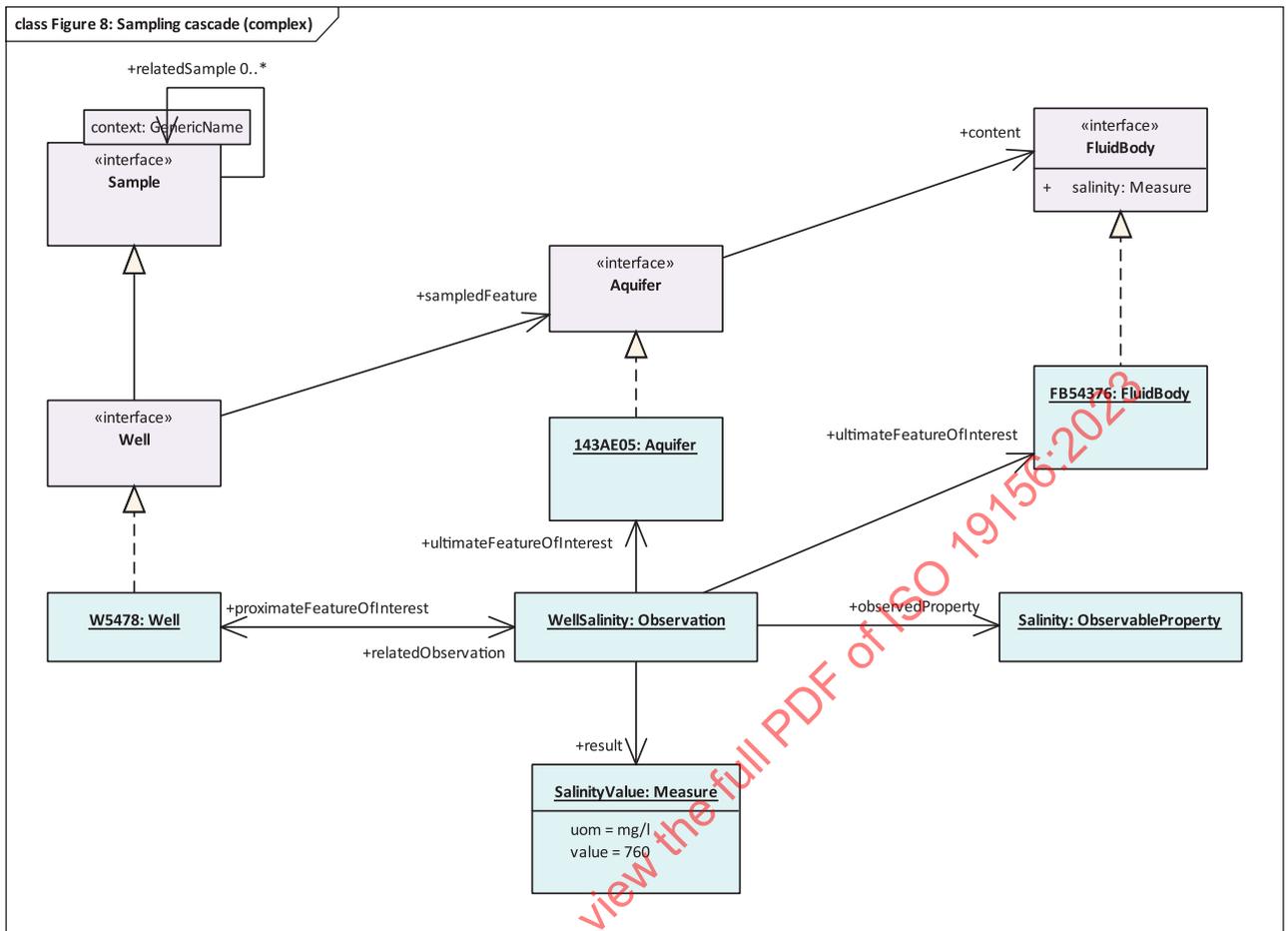


Figure 8 — (Example) Complex Sampling Cascade example referencing external domain feature

## 8 Conceptual Observation schema

### 8.1 General

#### 8.1.1 Conceptual Observation model

The Conceptual Observation schema is described as a class diagram in [Figure 9](#). The schema is fully described in [8.1.2](#).



## 8.2 Observation

### 8.2.1 Observation Requirements Class

<b>Requirements Class</b>	/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, Core-Types 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/observingProcedure-sem
Requirement	/req/obs-cpt/Observation/observingProcedure-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/observingProcedure-con
Recommendation	/rec/obs-cpt/Observation/result-con
Recommendation	/rec/obs-cpt/Observation/phenomenonTimeResult-con
Requirement	/req/obs-cpt/gen/relatedObservation-sem
Requirement	/req/obs-cpt/Observation/uom
Recommendation	/rec/obs-cpt/Observation/uom-con

### 8.2.2 Interface Observation

<b>Requirement</b> /req/obs-cpt/Observation/Observation-sem	An <b>Observation</b> shall be defined as an act carried out by an <b>Observer</b> to determine the value of an <b>ObservableProperty</b> of an object ( <b>featureOfInterest</b> ) by using an <b>ObservingProcedure</b> ; the value is provided as the <b>result</b> .
--	--

NOTE It is important to note that the terms ‘observation’, ‘interpretation’, ‘forecast’ and ‘simulation’ do correspond to this definition. This aspect is further clarified in [Clause 7](#).

### 8.2.3 Attribute phenomenonTime

<b>Requirement</b> /req/obs-cpt/Observation/phenomenonTime-sem	The time for which the <b>result</b> applies to the characteristic of the <b>FeatureOfInterest</b> being observed.  If the <b>phenomenonTime</b> is described, this shall be provided by the attribute <b>phenomenonTime:TM_Object</b>
---	--

NOTE 1 The phenomenonTime is often the time of interaction with a real-world feature either by a SamplingProcedure (time at which a Sample has been taken) or by an ObservingProcedure.

NOTE 2 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.

<b>Requirement</b> /req/obs-cpt/Observation/phenomenonTime-card	An <b>Observation</b> shall have exactly 1 <b>phenomenonTime</b> .
--	--

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

### 8.2.4 Attribute resultTime

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

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 can be used to estimate the values for phenomena in the future or past. The phenomenonTime is the time to which the result applies, while the resultTime is the time at which the simulation was executed.

<b>Requirement</b> /req/obs-cpt/Observation/resultTime-card	An <b>Observation</b> shall have exactly 1 <b>resultTime</b> .
--	--

### 8.2.5 Attribute validTime

<b>Requirement</b> /req/obs-cpt/Observation/validTime-sem	The time interval during which the <b>result</b> is assumed to be applicable for use.  If <b>validTime(s)</b> are described, they shall be provided by the attribute <b>validTime:TM_Period</b>
--	---

NOTE This attribute is commonly required in forecasting applications.

### 8.2.6 Association featureOfInterest

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

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 1 The featureOfInterest can be of Any type.

NOTE 2 This object is either the real-world object whose properties are under observation, or it is an object used as a proxy for a real-world object that is not directly observable, as described in 7.2. An observation instance serves as a propertyValueProvider for its feature-of-interest.

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

### 8.2.7 Association observedProperty

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

<b>Requirement</b> /req/obs-cpt/Observation/observedProperty-card	An <b>Observation</b> shall have exactly 1 <b>observedProperty</b> .
--	--

### 8.2.8 Association result

<b>Requirement</b> /req/obs-cpt/Observation/result-sem	The <b>result</b> of the <b>Observation</b> . If a reference to a <b>result</b> is provided, the association <b>Range</b> with the role <b>result</b> shall be used.
---	---

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

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

<b>Requirement</b> /req/obs-cpt/Observation/result-card	An <b>Observation</b> shall have exactly 1 <b>result</b> .
--	--

### 8.2.9 Association observingProcedure

<b>Requirement</b> /req/obs-cpt/Observation/observingProcedure-sem	The <b>ObservingProcedure</b> is used by the <b>Observation</b> to determine the value of the <b>ObservableProperty</b> provided by the <b>result</b> .  If a reference to an <b>ObservingProcedure</b> is provided, the association with the role <b>procedure</b> 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.

<b>Requirement</b> /req/obs-cpt/Observation/observingProcedure-card	An <b>Observation</b> shall have exactly 1 <b>observingProcedure</b> .
--	--

### 8.2.10 Association observer

<b>Requirement</b> /req/obs-cpt/Observation/observer-sem	An <b>Observer</b> that is involved in the creation of this <b>Observation</b> .  If a reference to an <b>Observer</b> is provided, the association with the role <b>observer</b> shall be used.
---	--

### 8.2.11 Association host

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

### 8.2.12 Constraint Observer or Host

<b>Recommendation</b> /rec/obs-cpt/Observation/observerhost-con	At least one <b>Observer</b> or <b>Host</b> should be provided.
--	---

### 8.2.13 Constraint ObservableProperty characteristic associated with featureOfInterest

<b>Recommendation</b> /rec/obs-cpt/Observation/observedProperty-con	The <b>ObservableProperty</b> referenced by <b>observedProperty</b> should correspond to a characteristic associated with the <b>featureOfInterest</b> .
--	--

### 8.2.14 Constraint suitable ObservableProperty

<b>Recommendation</b> /rec/obs-cpt/Observation/observingProcedure-con	The <b>ObservingProcedure</b> referenced by <b>procedure</b> should be suitable for the associated <b>ObservableProperty</b> .
--	--

### 8.2.15 Constraint suitable result type

<b>Recommendation</b> /rec/obs-cpt/Observation/result-con	The type of the result provided by the <b>result</b> association should be suitable for the associated <b>ObservableProperty</b> .
--	--

### 8.2.16 Constraint unit of measure

<b>Requirement</b> /req/obs-cpt/Observation/uom	The <b>Observation</b> shall provide a unit of measure (UoM) if the result is measurable. If the UoM is not contained in the result, it shall be provided in the context of the <b>Observation</b> ; the provision modality is to be defined by communities.
--	--

<b>Recommendation</b> /rec/obs-cpt/Observation/uom-con	The unit of measure should be suitable for the associated <b>ObservableProperty</b> and <b>ObservingProcedure</b> .
---	---

NOTE 1 In the case where the result of the Observation is a classification, for which no unit exists, the UoM can be declared as unitless (e.g. referencing the QUDT<sup>[27]</sup> <https://qudt.org/vocab/unit/UNITLESS> or UCUM<sup>[19]</sup> entry for “no units”).

## 8.3 ObservableProperty

### 8.3.1 ObservableProperty Requirements Class

<b>Requirements Class</b>	/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

### 8.3.2 Interface ObservableProperty

<b>Requirement</b> /req/obs-cpt/ObservableProperty/ObservableProperty-sem	An <b>ObservableProperty</b> shall be defined as a quality (property, characteristic) of the <b>feature-of-interest</b> that can 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, the:

- a) Groundwater Level (1 observable property)

is monitored

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

In addition, the groundwater well is revisited in the context of physical campaigns, where the:

- c) Groundwater Level (still the same observable property as above)

is measured, but

- d) with a manual probe (this is a different procedure than used above).

This allows for checking whether the probe needs recalibration.

8.3.3 Association observer

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

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

8.4.2 Interface Procedure

<p><b>Requirement</b> /req/obs-cpt/Procedure/Procedure-sem</p>	<p>A <b>Procedure</b> shall be defined as a description of steps performed.</p>
--	---

NOTE 1 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.

NOTE 2 The term "process" that was used in ISO 19156:2011 was purposely dropped in this document 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

8.5.2 Interface ObservingProcedure

<p><b>Requirement</b> /req/obs-cpt/ObservingProcedure/ObservingProcedure-sem</p>	<p>An <b>ObservingProcedure</b> shall be defined as the description of steps performed in order to determine the value of an <b>observableProperty</b> by an <b>Observer</b>.</p>
--	---

NOTE 1 Depending on the complexity of the use case, the procedure will be more or less explicitly described. Especially pertaining to historical data, there can be very little or no information available. Information on the recipe that the observer (cook) follows to generate the Observation is ideally included.

NOTE 2 The procedure is often referred to as the method.

NOTE 3 Different observers can follow the same (reusable) procedure for the creation of different observations.

NOTE 4 The procedure is a workflow, protocol, plan, algorithm, or computational method specifying how to make an observation.

NOTE 5 The observing procedure cannot describe a sensor instance, but it can describe the sensor type.

NOTE 6 The term "process" that was used in ISO 19156:2011 has been purposely dropped in this document 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. Procedure could further describe the algorithms behind simulators or models used to generate a result from other inputs.

### 8.5.3 Association observer

<p><b>Requirement</b> /req/obs-cpt/ObservingProcedure/observer-sem</p>	<p>An <b>Observer</b> capable of performing this <b>Observing-Procedure</b>. If a reference to an <b>Observer</b> is provided, the association with the role <b>observer</b> shall be used.</p>
--	---

## 8.6 Observer

### 8.6.1 Observer Requirements Class

<b>Requirements Class</b>	/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

### 8.6.2 Interface Observer

<p><b>Requirement</b> /req/obs-cpt/Observer/Observer-sem</p>	<p>An <b>Observer</b> shall be defined as an identifiable entity that can generate <b>Observations</b> pertaining to an <b>observableProperty</b> by implementing an <b>ObservingProcedure</b>.</p>
--	---

NOTE 1 Different Observers can follow the same (reusable) observing Procedure for the creation of different Observations.

NOTE 2 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.

NOTE 3 An Observer is closely linked with an observableProperty for which it generates results for.

NOTE 4 An Observer can be hosted by one or more Host.

NOTE 5 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, etc. are Observers 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

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

### 8.6.4 Association observingProcedure

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

### 8.6.5 Association deployment

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

## 8.7 Host

### 8.7.1 Host Requirements Class

<b>Requirements Class</b>	/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

### 8.7.2 Interface Host

<b>Requirement</b> /req/obs-cpt/Host/Host-sem	A <b>Host</b> shall be defined as a grouping of <b>Observers</b> for a specific reason.
--	---

NOTE 1 In many use cases, the Host is the environmental monitoring facility.

NOTE 2 The Host can be a platform that hosts a set of sensors.

NOTE 3 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

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

### 8.7.4 Association relatedHost

<b>Requirement</b> /req/obs-cpt/Host/relatedHost-sem	<b>A Host</b> the <b>Host</b> is related to. If a reference to a related <b>Host</b> is provided, the association with role <b>relatedHost</b> shall be used. The <b>context: GenericName</b> 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

<b>Requirements Class</b>	/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

### 8.8.2 Interface Deployment

<b>Requirement</b> /req/obs-cpt/Deployment/Deployment-sem	<b>A Deployment</b> shall be defined as information on the assignment of an <b>Observer</b> to a <b>Host</b> .
--	--

EXAMPLE 1 Information regarding a sensor being attached to a pole.

EXAMPLE 2 The monitoring facilities pertaining to an environmental monitoring network.

EXAMPLE 3 The description of a ship cruise linking a research vessel with a marine network.

EXAMPLE 4 The participation of a citizen in a citizen-science project involving crowd sensing.

### 8.8.3 Association observer

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

### 8.8.4 Association host

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

## 9 Abstract Observation Core

### 9.1 General

#### 9.1.1 Abstract Observation Core Package Requirements Class

<b>Requirements Class</b>	/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
Imports	/req/obs-core/AbstractObservationCollection

#### 9.1.2 Association metadata

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

NOTE When providing metadata, using the classes, attributes and associations explicitly modelled in the OMS greatly improves the interoperability compared to using the generic metadata association to include the same information.

## 9.2 AbstractObservationCharacteristics

#### 9.2.1 AbstractObservationCharacteristics Requirements Class

<b>Requirements class</b>	/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, Core-Types 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
Recommendation	/rec/obs-core/AbstractObservationCharacteristics/parameter-procedure
Recommendation	/rec/obs-core/AbstractObservationCharacteristics/parameter-redundant
Requirement	/req/obs-core/AbstractObservationCharacteristics/resultQuality-sem
Requirement	/req/obs-cpt/Observation/phenomenonTime-sem

<b>Requirements class</b>	/req/obs-core/AbstractObservationCharacteristics
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-basic/ObservationCharacteristics/collection-sem
Requirement	/req/obs-cpt/Observation/observedProperty-sem
Requirement	/req/obs-cpt/Observation/result-sem
Requirement	/req/obs-cpt/Observation/observingProcedure-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

AbstractObservationCharacteristics and AbstractObservation from the Abstract Observation Core are described as a class diagram in [Figure 10](#). The schema is fully described in [9.2](#) and [9.3](#).

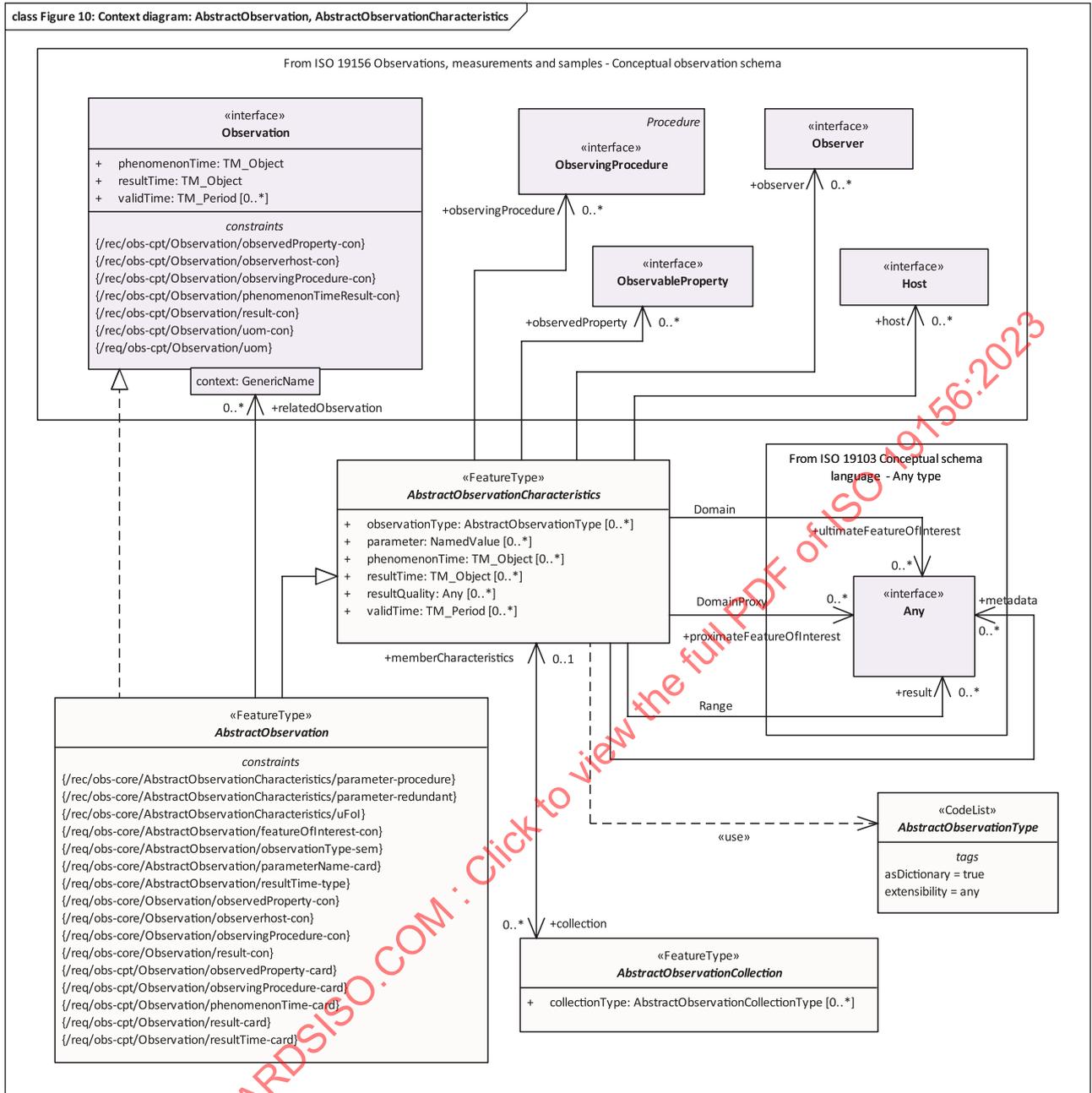


Figure 10 — Context diagram for Abstract Observation Core — AbstractObservationCharacteristics and AbstractObservation

9.2.2 Feature type AbstractObservationCharacteristics

<p><b>Requirement</b> /req/obs-core/AbstractObservationCharacteristics/AbstractObservationCharacteristics-sem</p>	<p>An <b>AbstractObservationCharacteristics</b> shall be defined as a set of common characteristics used for describing an <b>Observation</b> or a collection of Observations.</p>
---	--

### 9.2.3 Attribute observationType

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

NOTE 1 Observation type allows describing the formalism, encoding, etc. to be expected when accessing objects associated to the Observation.

NOTE 2 Multiple types can be applied to one Observation, such as in the case where the Observation is being typed both by the Domain (feature-of-interest geometry) as well as Range (result type).

### 9.2.4 Attribute parameter

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

<b>Recommendation</b> /rec/obs-core/AbstractObservationCharacteristics/parameter-procedure	<b>Parameter</b> should not be used instead of the procedure to describe the steps performed in order to determine the value of the ObservableProperty.
---	---

<b>Recommendation</b> /rec/obs-core/AbstractObservationCharacteristics/parameter-redundant	<b>Parameter</b> should not be utilized to provide information already contained in the model by existing attributes or associations.
---	---

The ObservingProcedure 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 or operator may augment the description of a standard procedure.

EXAMPLE A time sequence of observations of water quality in a well can 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.

NOTE 1 This can 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.

NOTE 2 Parameters that are tightly bound to the procedure can be recorded as part of the procedure description.

### 9.2.5 Attribute resultQuality

<b>Requirement</b> /req/obs-core/AbstractObservationCharacteristics/resultQuality-sem	Information pertaining to the data quality of the <b>result</b> .  If additional data quality information is provided, the property <b>resultQuality:Any</b> 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 can be assessed following the procedures in the ISO 19157 series. Multiple measures can be provided.

9.2.6 Association proximateFeatureOfInterest

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

NOTE The measurement process can 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

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

EXAMPLE 1 A river, an aquifer, soil layer, outcrop, a butterfly, a survey area, a room, Abby’s car, a specific human being, this document.

EXAMPLE 2 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 3 Pertaining to documents and observations on the consistency of documents, for the Observation “This clause is inconsistent”, the ultimateFeatureOfInterest is the entire document, while the proximateFeatureOfInterest is the specific clause being addressed.

EXAMPLE 4 The determination of the species of the butterfly, in this case the butterfly is the ultimateFeatureOfInterest, no proximateFeatureOfInterest need be provided.

NOTE 1 The measurement process can 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.

<p><b>Recommendation</b> /rec/obs-core/AbstractObservationCharacteristics/uFol</p>	<p>In the case where ultimate and proximate features-of-interest are the same object, the association should be provided using the <b>ultimateFeatureOfInterest</b> association role.</p>
--	---

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

### 9.2.8 Association collection

<b>Requirement</b> /req/obs-basic/ObservationCharacteristics/ collection-sem	An <b>AbstractObservationCollection</b> that is described by these <b>AbstractObservationCharacteristics</b> .  If a reference to an <b>AbstractObservationCollection</b> from the <b>AbstractObservationCharacteristics</b> is provided, the association with the role <b>collection</b> shall be used.
--	--

## 9.3 AbstractObservation

### 9.3.1 AbstractObservation Requirements Class

<b>Requirements Class</b>	/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-cpt/Observation/phenomenonTime-card
Requirement	/req/obs-cpt/Observation/resultTime-card
Requirement	/req/obs-cpt/Observation/observedProperty-card
Requirement	/req/obs-cpt/Observation/observingProcedure-card
Requirement	/req/obs-cpt/Observation/result-card
Requirement	/req/obs-cpt/Observation/Observation-sem
Requirement	/req/obs-core/AbstractObservation/observationType-sem
Requirement	/req/obs-core/AbstractObservationType/AbstractObservationType-sem
Requirement	/req/obs-core/AbstractObservation/resultTime-type
Requirement	/req/obs-core/AbstractObservation/featureOfInterest-con
Requirement	/req/obs-core/AbstractObservation/parameterName-card
Requirement	/req/obs-core/Observation/observerhost-con
Requirement	/req/obs-core/Observation/observedProperty-con
Requirement	/req/obs-core/Observation/observingProcedure-con
Requirement	/req/obs-core/Observation/result-con
Requirement	/req/obs-cpt/Observation/uom
Recommendation	/rec/obs-cpt/Observation/uom-con
Recommendation	/rec/obs-cpt/Observation/observedProperty-con
Recommendation	/rec/obs-cpt/Observation/observerhost-con
Recommendation	/rec/obs-cpt/Observation/observingProcedure-con
Recommendation	/rec/obs-cpt/Observation/result-con
Recommendation	/rec/obs-cpt/Observation/phenomenonTimeResult-con

### 9.3.2 Constraint observationType

<b>Requirement</b> /req/obs-core/AbstractObservation/observationType-sem	If information on the type of <b>Observation</b> is provided, the constraints defined in the referenced codelist shall be used.
---	---

### 9.3.3 Constraint resultTime instant

<b>Requirement</b> /req/obs-core/AbstractObservation/resultTime-type	If the result time of the <b>Observation</b> is provided, the <b>resultTime</b> attribute shall be of type <b>TM_Instant</b> .
---	--

### 9.3.4 Constraint parameter unique name

<b>Requirement</b> /req/obs-core/AbstractObservation/parameterName-card	The <b>name</b> attribute of a <b>parameter NamedValue</b> shall be unique within an <b>Observation</b> .
--	---

### 9.3.5 Constraint proximate or ultimate featureOfInterest.

<b>Requirement</b> /req/obs-core/AbstractObservation/featureOfInterest-con	At least one <b>proximateFeatureOfInterest</b> or <b>ultimateFeatureOfInterest</b> shall be given.
---	--

### 9.3.6 Constraint Observer or Host

<b>Requirement</b> /req/obs-core/Observation/observerhost-con	At least one <b>Observer</b> or <b>Host</b> shall be provided
--	---

### 9.3.7 Constraint ObservableProperty characteristic associated with featureOfInterest

<b>Requirement</b> /req/obs-core/Observation/observedProperty-con	The <b>ObservableProperty</b> referenced by <b>observedProperty</b> shall correspond to a characteristic associated with the <b>featureOfInterest</b> .
--	---

### 9.3.8 Constraint suitable ObservableProperty

<b>Requirement</b> /req/obs-core/Observation/observingProcedure-con	The <b>ObservingProcedure</b> referenced by <b>procedure</b> shall be suitable for the associated <b>ObservableProperty</b> .
--	---

### 9.3.9 Constraint suitable result type

<b>Requirement</b> /req/obs-core/Observation/result-con	The type of the result provided by the <b>result</b> association shall be suitable for the associated <b>ObservableProperty</b> .
--	---

## 9.4 AbstractObservableProperty

### 9.4.1 AbstractObservableProperty Requirements Class

<b>Requirements Class</b>	/req/obs-core/AbstractObservableProperty
Target type	Logical model
Name	Abstract Observation Core – AbstractObservableProperty

Requirements Class	/req/obs-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

AbstractObservableProperty from the Abstract Observation Core is described as a class diagram in [Figure 11](#). The schema is fully described in [9.4](#).

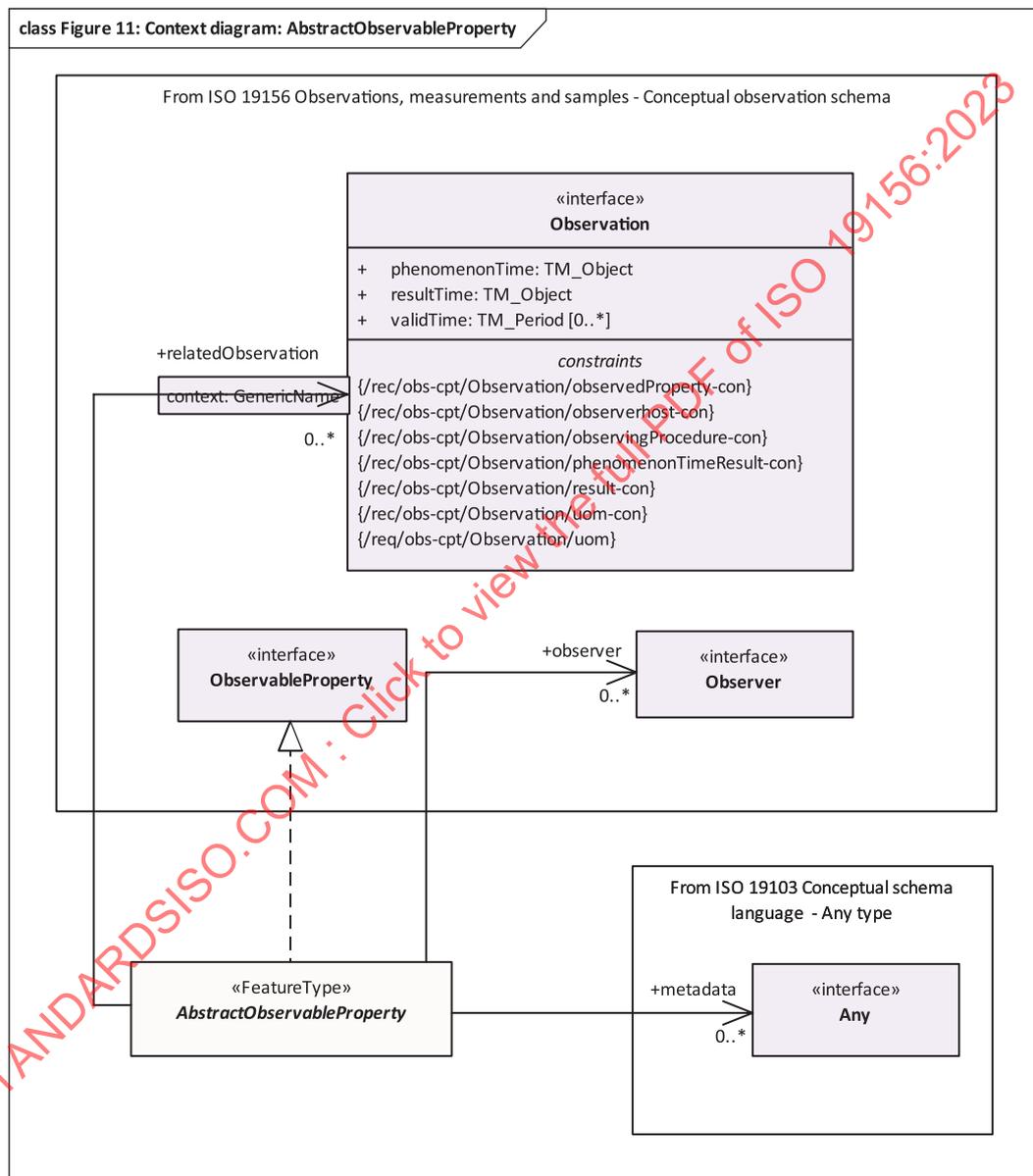


Figure 11 — Context diagram for Abstract Observation Core — AbstractObservableProperty

## 9.5 AbstractObservingProcedure

### 9.5.1 AbstractObservingProcedure Requirements Class

Requirements Class	/req/obs-core/AbstractObservingProcedure
Target type	Logical model

<b>Requirements Class</b>	/req/obs-core/AbstractObservingProcedure
<b>Name</b>	Abstract Observation Core – AbstractObservingProcedure
<b>Dependency</b>	ISO 19103:2015 Geographic information — Conceptual schema language, UML2 conformance class
<b>Imports</b>	/req/obs-cpt/ObservingProcedure
<b>Requirement</b>	/req/obs-core/gen/metadata-sem

AbstractObservingProcedure from the Abstract Observation Core is described as a class diagram in [Figure 12](#). The schema is fully described in [9.5](#).

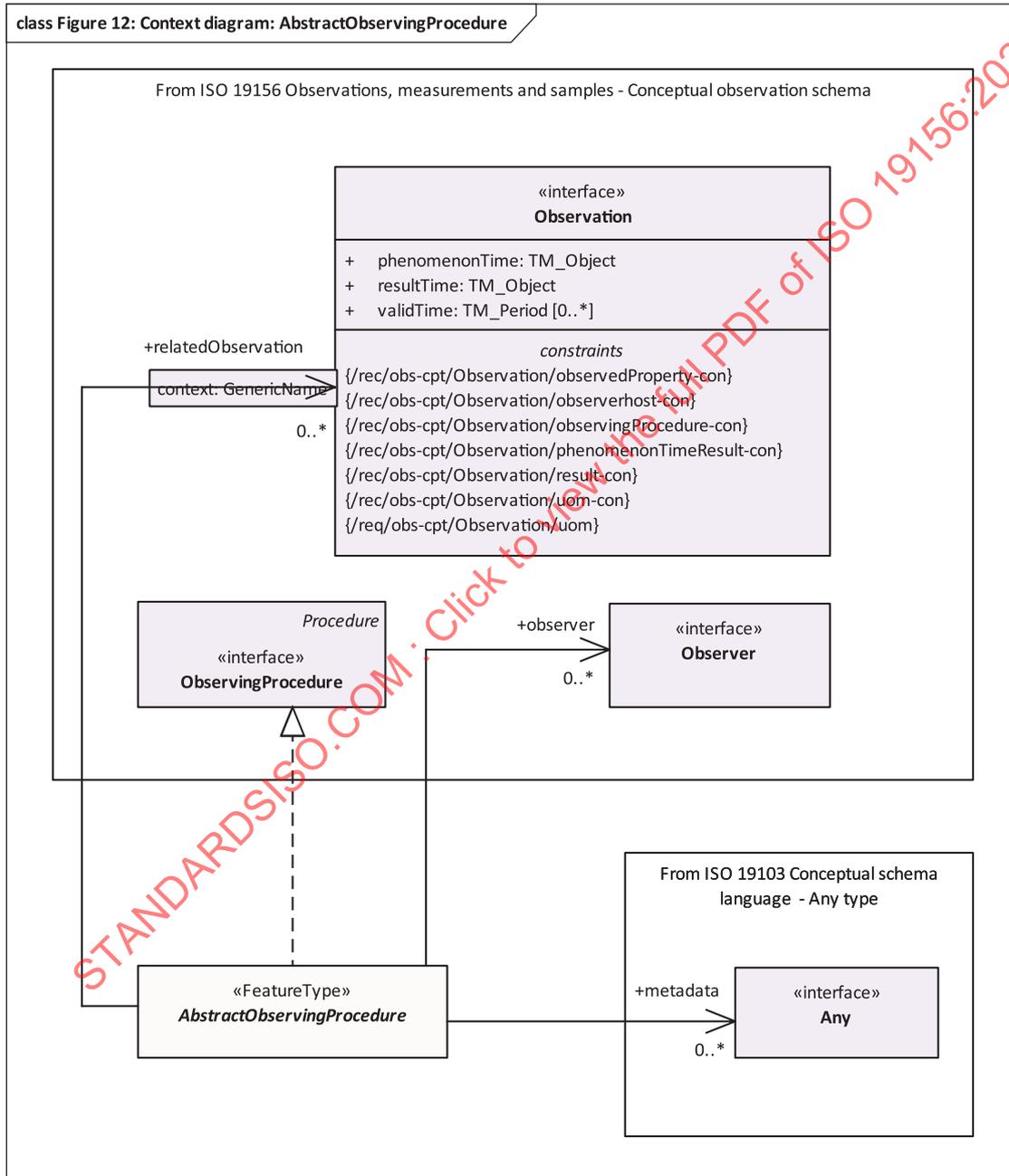


Figure 12 — Context diagram for Abstract Observation Core — AbstractObservingProcedure

## 9.6 AbstractObserver

### 9.6.1 AbstractObserver Requirements Class

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

AbstractObserver from the Abstract Observation Core are described as a class diagram in [Figure 13](#). The schema is fully described in [9.6](#).

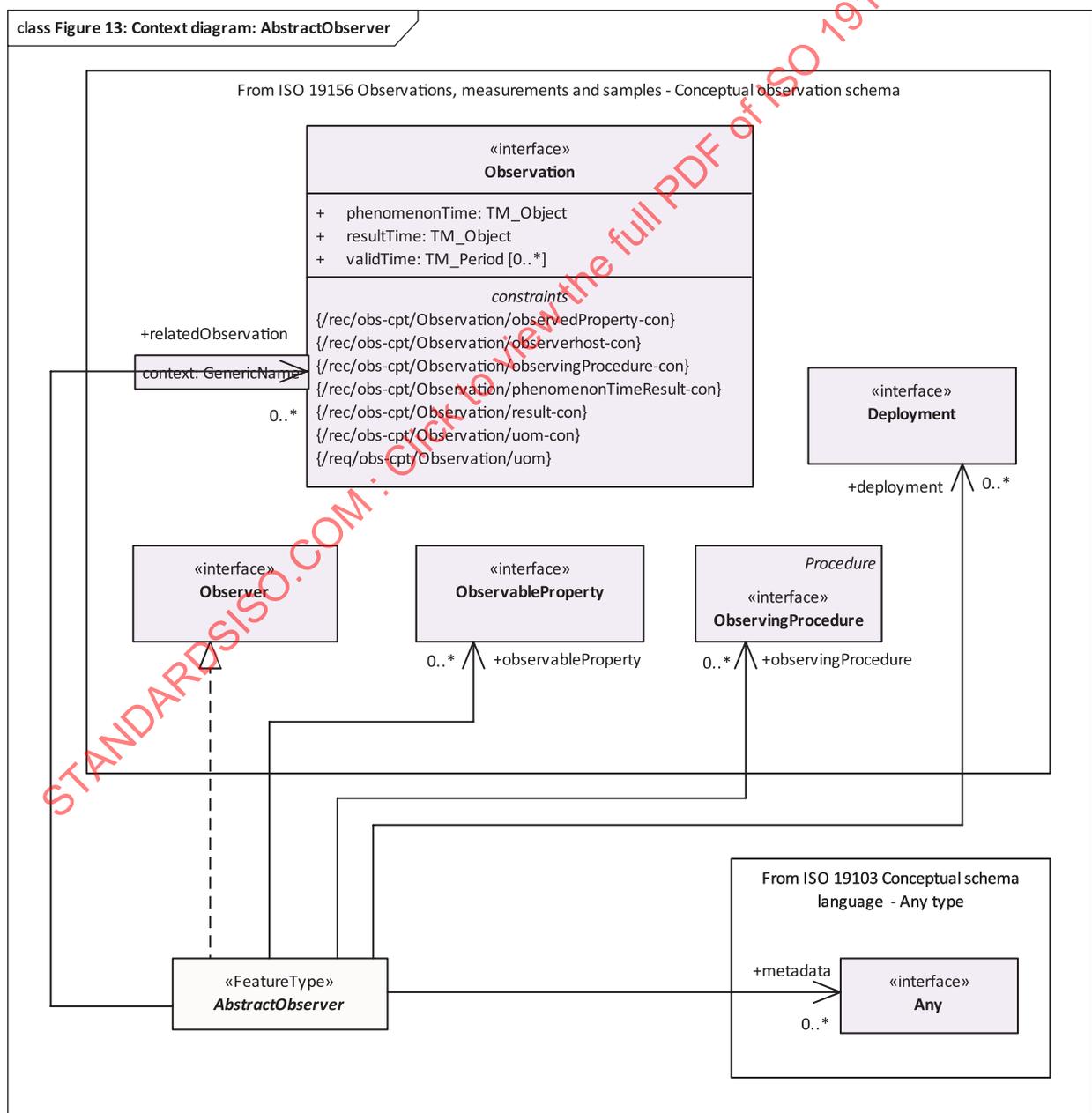


Figure 13 — Context diagram for Abstract Observation Core — AbstractObserver

## 9.7 AbstractHost

### 9.7.1 AbstractHost Requirements Class

<b>Requirements Class</b>	/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

AbstractHost from the Abstract Observation Core is described as a class diagram in [Figure 14](#). The schema is fully described in [9.7](#).

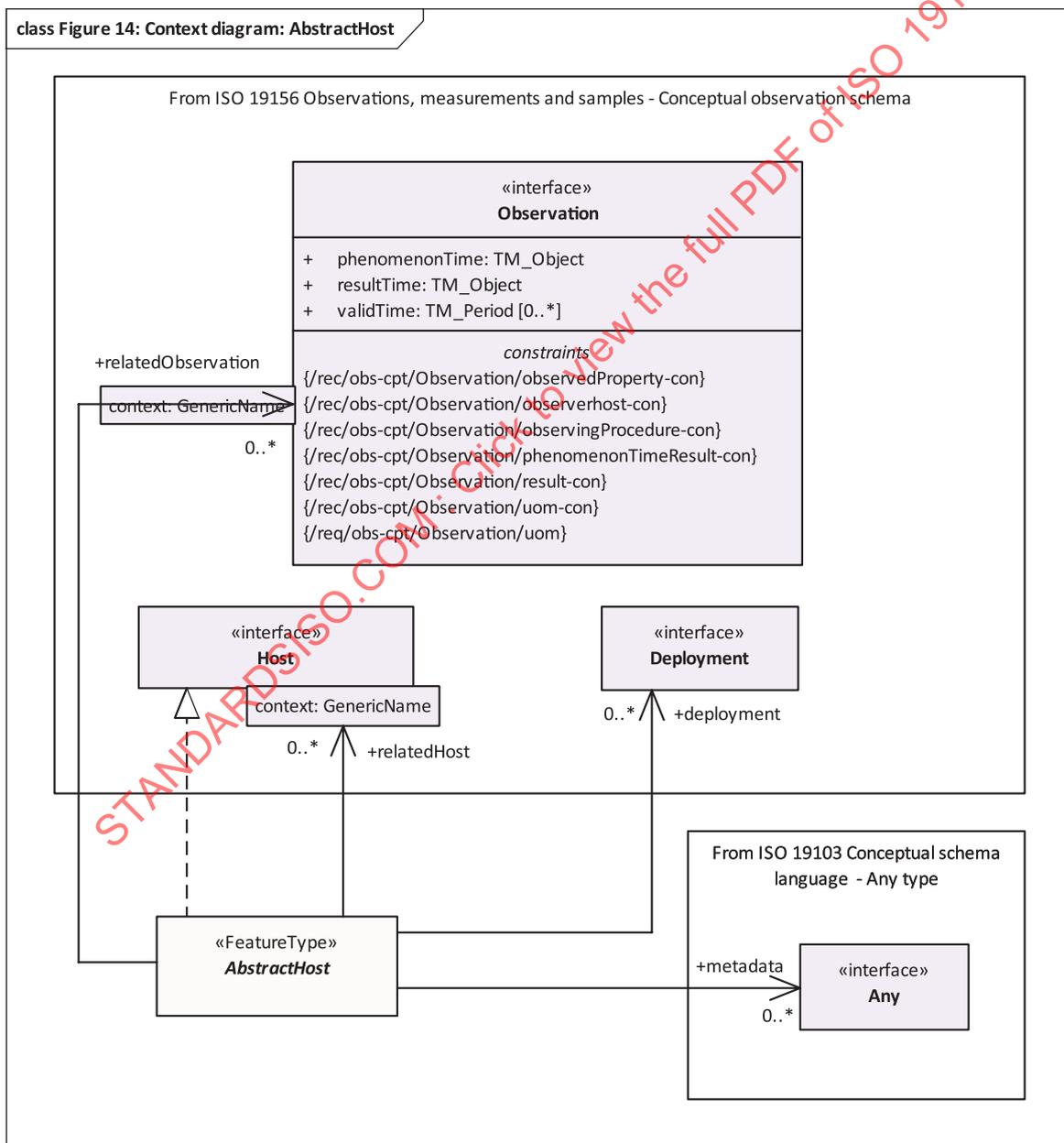


Figure 14 — Context diagram for Abstract Observation Core — AbstractHost

## 9.8 AbstractDeployment

### 9.8.1 AbstractDeployment Requirements Class

<b>Requirements Class</b>	/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

AbstractDeployment from the Abstract Observation Core are described as a class diagram in [Figure 15](#). The schema is fully described in [9.8](#).

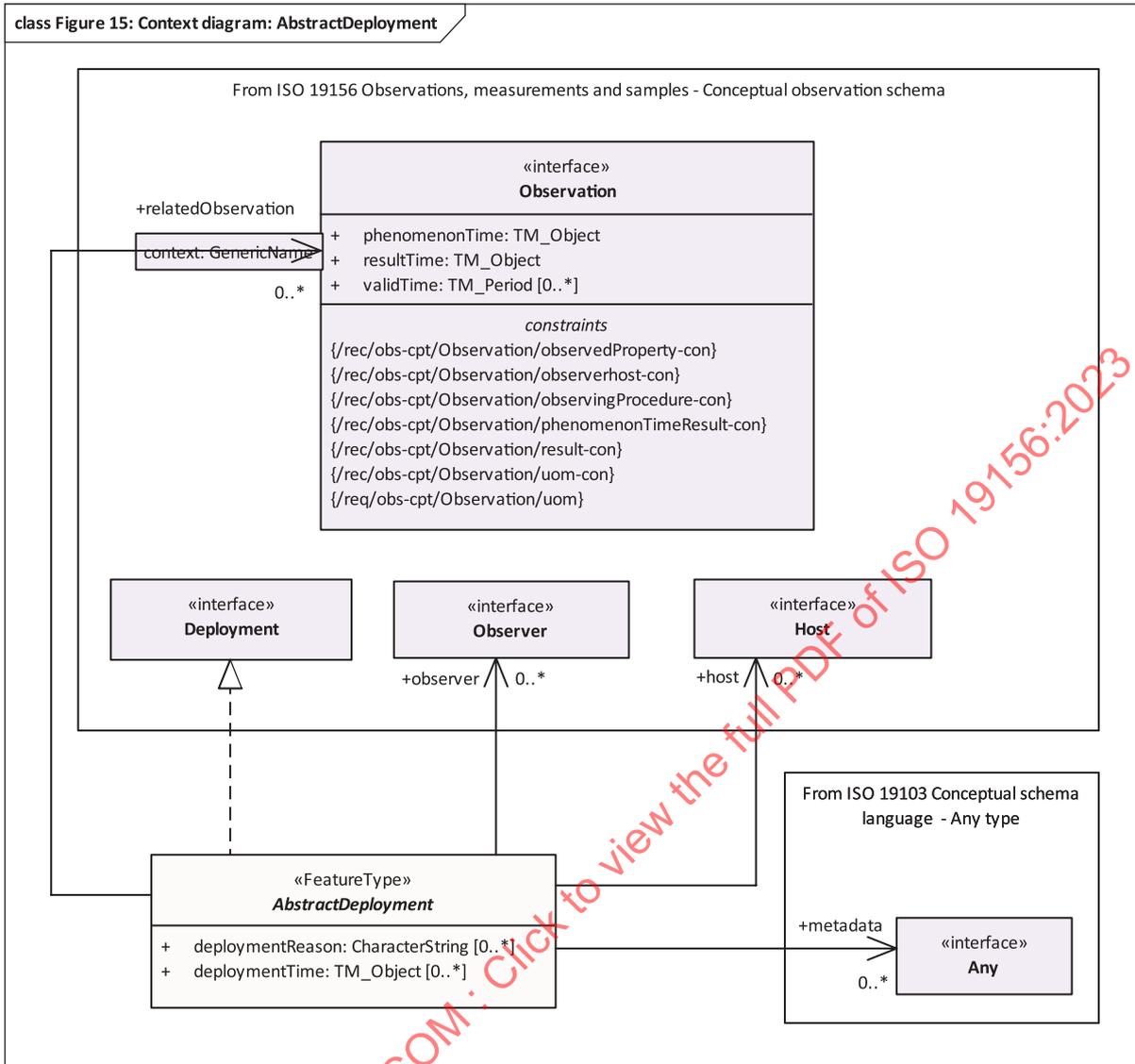


Figure 15 — Context diagram for Abstract Observation Core — AbstractDeployment

### 9.8.2 Attribute deploymentReason

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

EXAMPLE 1 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.

EXAMPLE 2 A sensor is mounted on a building to monitor seismic activities.

EXAMPLE 3 A new sensor type is rolled out within a regional or thematic network due to new legal reporting requirements.

### 9.8.3 Attribute deploymentTime

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

EXAMPLE 1 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.

EXAMPLE 2 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 AbstractObservationCollection

### 9.9.1 AbstractObservationCollection Requirements Class

<b>Requirements Class</b>	/req/obs-core/AbstractObservationCollection
Target type	Logical model
Name	Abstract Observations – AbstractObservationCollection
Dependency	ISO 19103:2015, Geographic information — Conceptual schema language, UML2 conformance class
Requirement	/req/obs-core/AbstractObservationCollection/AbstractObservationCollection-sem
Requirement	/req/obs-core/AbstractObservationCollection/collectionType-sem
Requirement	/req/obs-core/AbstractObservationCollection/collectionType-con
Requirement	/req/obs-core/AbstractObservationCollection/member-sem
Requirement	/req/obs-core/AbstractObservationCollection/memberCharacteristics-sem
Requirement	/req/obs-core/AbstractObservationCollection/relatedCollection-sem
Requirement	/req/obs-cpt/gen/relatedObservation-sem
Requirement	/req/obs-core/AbstractObservationCollectionType/AbstractObservationCollectionType-sem

AbstractObservationCollection from the Abstract Observation Core is described as a class diagram in [Figure 16](#). The schema is fully described in [9.9](#).

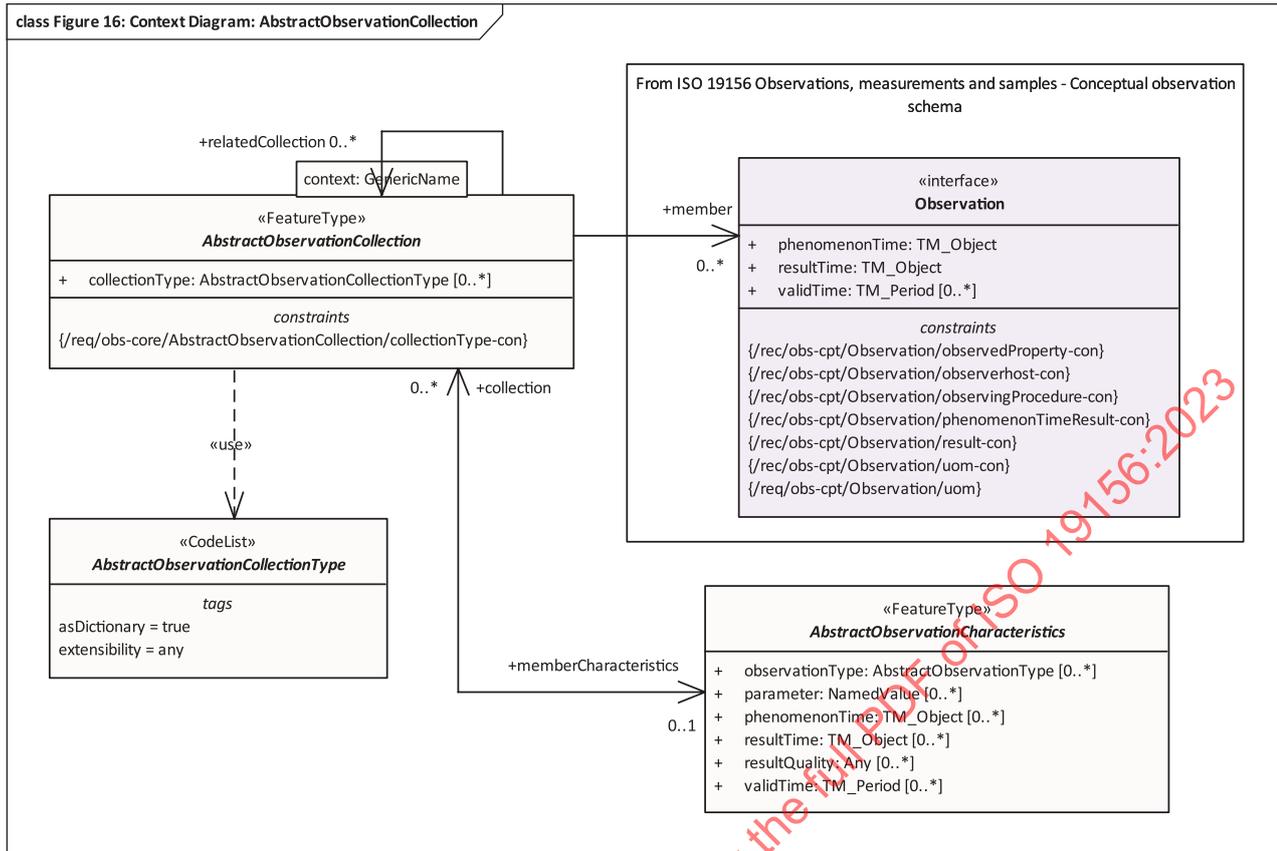


Figure 16 — Context diagram for Abstract Observation Core — AbstractObservationCollection

### 9.9.2 Feature type AbstractObservationCollection

<b>Requirement</b> /req/obs-core/AbstractObservationCollection/AbstractObservationCollection-sem	An <b>AbstractObservationCollection</b> shall be defined as a collection of similar <b>Observations</b> .
---	---

### 9.9.3 Attribute collectionType

<b>Requirement</b> /req/obs-core/AbstractObservationCollection/collectionType-sem	Information on the type of the <b>AbstractObservationCollection</b> . If information on the collection type is provided, the attribute <b>collectionType:AbstractObservationCollectionType</b> shall be used.
--	--

<b>Requirement</b> /req/obs-core/AbstractObservationCollection/collectionType-con	If the <b>collectionType</b> is provided, property values of the associated <b>Observation</b> and <b>AbstractObservationCharacteristics</b> instances shall comply with the constraints defined for this <b>collectionType</b> value.
--	--

9.9.4 Association member

<b>Requirement</b> /req/obs-core/AbstractObservationCollection/member-sem	An <b>Observation</b> that is part of this <b>AbstractObservationCollection</b> .  If a reference to a member <b>Observation</b> is provided, the association with the role <b>member</b> shall be used.
--	--

9.9.5 Association memberCharacteristics

<b>Requirement</b> /req/obs-core/AbstractObservationCollection/memberCharacteristics-sem	Information on <b>AbstractObservationCharacteristics</b> of <b>Observations</b> contained within the <b>AbstractObservationCollection</b> .  If a reference to <b>AbstractObservationCharacteristics</b> pertaining to the collection members is provided, the association with the role <b>memberCharacteristics</b> shall be used.
---	--

9.9.6 Association relatedCollection

<b>Requirement</b> /req/obs-core/AbstractObservationCollection/relatedCollection-sem	An <b>AbstractObservationCollection</b> the <b>AbstractObservationCollection</b> is related to.  If a reference to a related <b>AbstractObservationCollection</b> is provided, the association with role <b>relatedCollection</b> shall be used. The <b>context:GenericName</b> qualifier of this association may be used to provide further information as to the nature of the relation.
---	--

9.10 NamedValue

9.10.1 NamedValue Requirements Class

<b>Requirements Class</b>	/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

9.10.2 Data type NamedValue

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

9.10.3 Attribute name

<b>Requirement</b> /req/obs-core/NamedValue/name-sem	The attribute <b>name:GenericName</b> shall indicate the meaning of the named value.
---	--

NOTE Using well-governed sources for the value of the name enhances reusability.

EXAMPLE When used as the value of an Observation: parameter, the name can take values like ‘procedureOperator’, ‘detectionLimit’, ‘amplifierGain’, ‘samplingDepth’, ‘analysisIteration’, etc.

9.10.4 Attribute value

<b>Requirement</b> /req/obs-core/NamedValue/value-sem	The attribute <b>value:Any</b> shall provide the value.
--	---

NOTE In concrete realizations, the type “Any” can be substituted by a suitable concrete type, such as CI\_ ResponsibleParty or Measure.

9.11 Codelists

9.11.1 AbstractObservationType

The code list AbstractObservationType can be specialized as required to more precisely define the semantics of observation types, as done in the derived codelist ObservationTypeByResultType below.

<b>Requirement</b> /req/obs-core/AbstractObservationType/ AbstractObservationType-sem	An empty extension-point for providing various classification schemes for <b>Observations</b> .  If <b>Observation</b> classification schemes are used in the implementing application schemas, a concrete realization shall be created for the application.
---	--

9.11.2 AbstractObservationCollectionType

The code list AbstractObservationCollectionType can be specialized as required to more precisely define the semantics of collection types, as done in the derived codelist ObservationCollectionType below.

<b>Requirement</b> /req/obs-core/AbstractObservationCollection- Type/AbstractObservationCollectionType-sem	An empty extension-point for providing various classification schemes for <b>ObservationCollections</b> .  If <b>ObservationCollection</b> classification schemes are used in the implementing application schemas, a concrete realization shall be created for the application.
--	--

10 Basic Observations

10.1 General

10.1.1 Basic Observations Package Requirements Class

<b>Requirements Class</b>	/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

<b>Requirements Class</b>	/req/obs-basic
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
Requirement	/req/obs-basic/ObservationTypeByResultType/ObservationTypeByResultType-sem
Requirement	/req/obs-basic/ObservationTypeByResultType/ObservationTypeByResultType-con
Requirement	/req/obs-basic/ObservationCollectionType/ObservationCollectionType-sem
Requirement	/req/obs-basic/ObservationCollectionType/homogeneous-con
Requirement	/req/obs-basic/ObservationCollectionType/summarizing-con

### 10.1.2 Attribute link

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

### 10.1.3 Attribute location

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

## 10.2 Observation

### 10.2.1 Observation Requirements Class

<b>Requirements Class</b>	/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

Observation from the Basic Observations is described as a class diagram in [Figure 17](#). The schema is fully described in [10.2](#).

class Figure 17: Context diagram: Observation

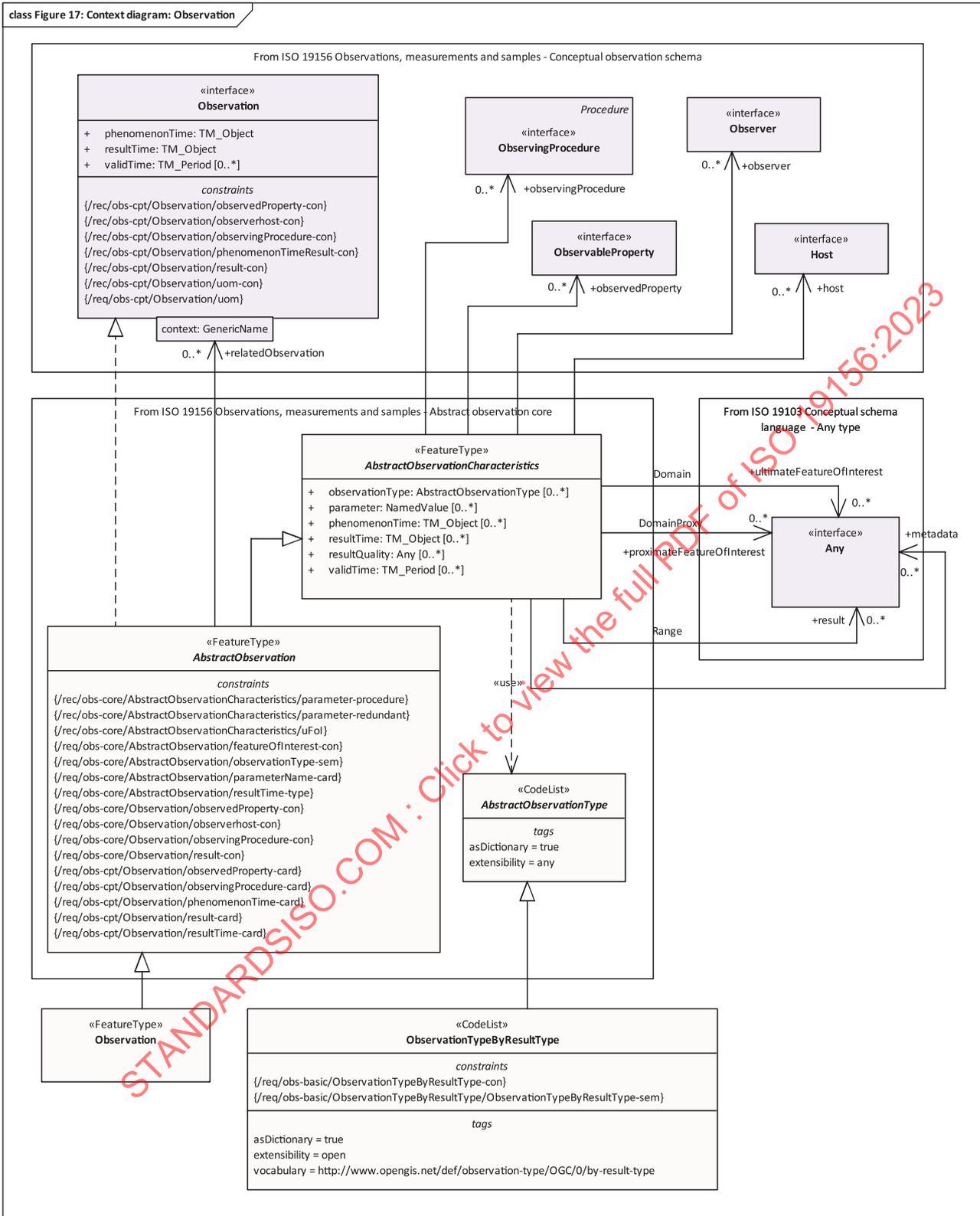


Figure 17 — Context diagram for Basic Observations — Observation

## 10.3 ObservationCharacteristics

### 10.3.1 ObservationCharacteristics Requirements Class

<b>Requirements Class</b>	/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

ObservationCharacteristics from the Basic Observations is described as a class diagram in [Figure 18](#). The schema is fully described in [10.3](#).

## 10.4 ObservationCollection

### 10.4.1 ObservationCollection Requirements Class

<b>Requirements Class</b>	/req/obs-basic/ObservationCollection
Target type	Logical model
Name	Basic Observations – ObservationCollection
Dependency	ISO 19103:2015, Geographic information — Conceptual schema language, UML2 conformance class
Imports	/req/obs-core/AbstractObservationCollection

ObservationCollection from the Basic Observations is described as a class diagram in [Figure 18](#). The schema is fully described in [10.4](#).

## 10.5 ObservingCapability

### 10.5.1 ObservingCapability Requirements Class

<b>Requirements Class</b>	/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

ObservationCollection from the Basic Observations is described as a class diagram in [Figure 18](#). The schema is fully described in [10.5](#).

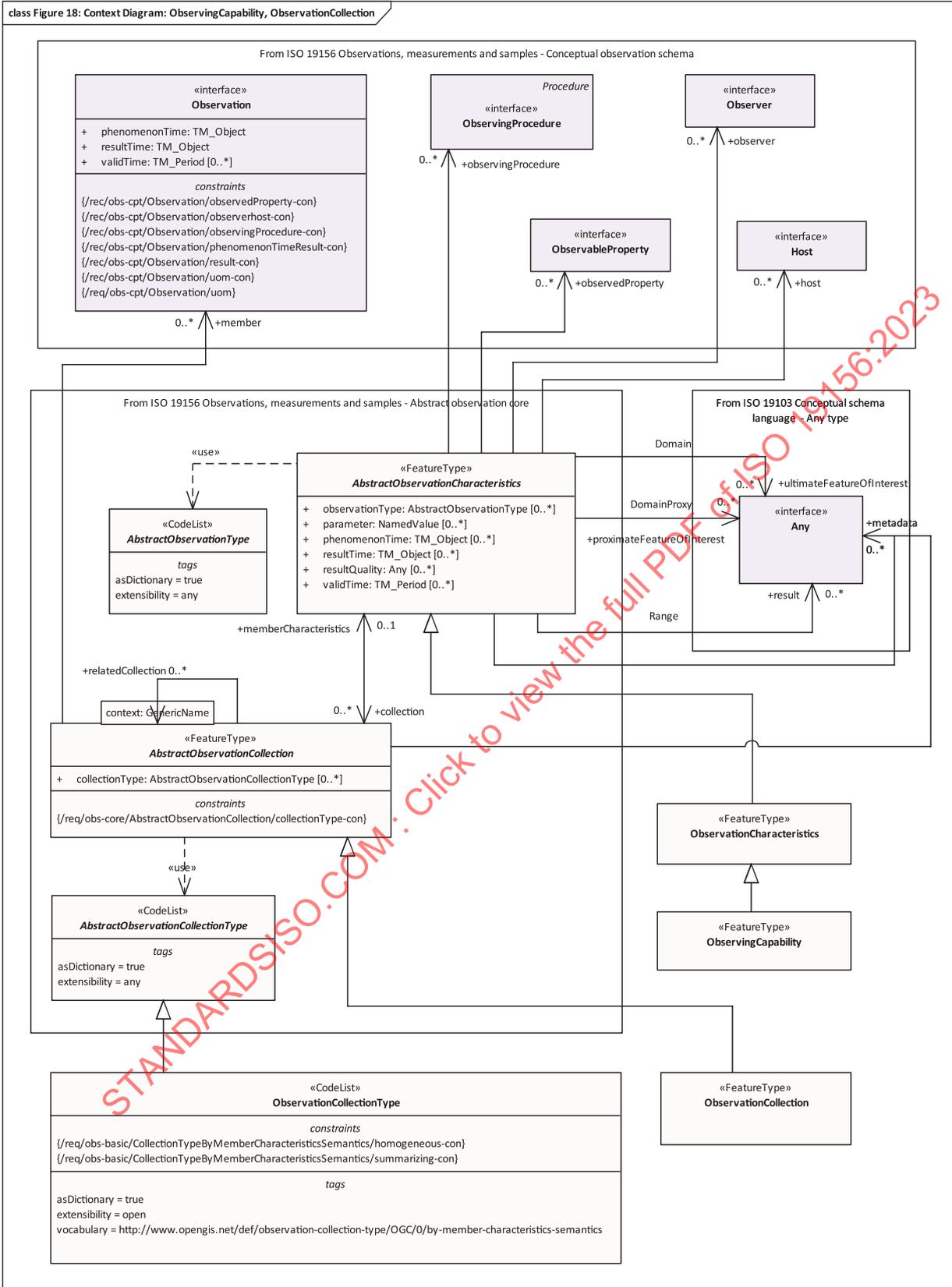


Figure 18 — Context diagram for Basic Observations — ObservationCharacteristics, ObservingCapability and ObservationCollection

### 10.5.2 Feature type ObservingCapability

<b>Requirement</b> /req/obs-basic/ObservingCapability/ObservingCapability-sem	An <b>ObservingCapability</b> shall be defined as information on <b>Observation(s)</b> that could potentially be provided.
--	--

**EXAMPLE** In order to explicitly describe the capabilities of an environmental monitoring facility, information on which Observable Properties are being measured with which methodology is provided.

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

- a) Some monitoring may have just one ObservingCapability:
  - 1) ObservingCapability:
    - i) ultimateFeatureOfInterest: 'Hydrogeological Unit 121AS';
    - ii) proximateFeatureOfInterest: 'xyz';
    - iii) procedure: 'Groundwater depth measurement by electronic probe';
    - iv) observedProperty: 'GroundWaterDepth'.
- b) Other monitoring may have several such ObservingCapabilities, for example:
  - 1) ObservingCapability 1:
    - i) ultimateFeatureOfInterest: 'Entite hydrogeologique 143AE05';
    - ii) proximateFeatureOfInterest: 'Calcaires du Muschelkalk de Lorraine à SERVIGNY-LES-RAVILLE';
    - iii) procedure: 'Groundwater depth measurement by electronic probe';
    - iv) observedProperty: 'GroundWaterDepth'.
  - 2) ObservingCapability 2:
    - i) ultimateFeatureOfInterest: 'Entite hydrogeologique 143AE05';
    - ii) proximateFeatureOfInterest : 'Calcaires du Muschelkalk de Lorraine à SERVIGNY-LES-RAVILLE';
    - iii) procedure: 'Digital recording teletransmitted';
    - iv) observedProperty: 'Water Temperature'.
  - 3) ObservingCapability 3:
    - i) ultimateFeatureOfInterest: 'Entite hydrogeologique 143AE05';
    - ii) proximateFeatureOfInterest : 'Calcaires du Muschelkalk de Lorraine à SERVIGNY-LES-RAVILLE';
    - iii) procedure: 'Digital recording teletransmitted';
    - iv) 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

<b>Requirements Class</b>	/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

ObservableProperty from the Basic Observations is described as a class diagram in [Figure 19](#). The schema is fully described in [10.6](#).

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

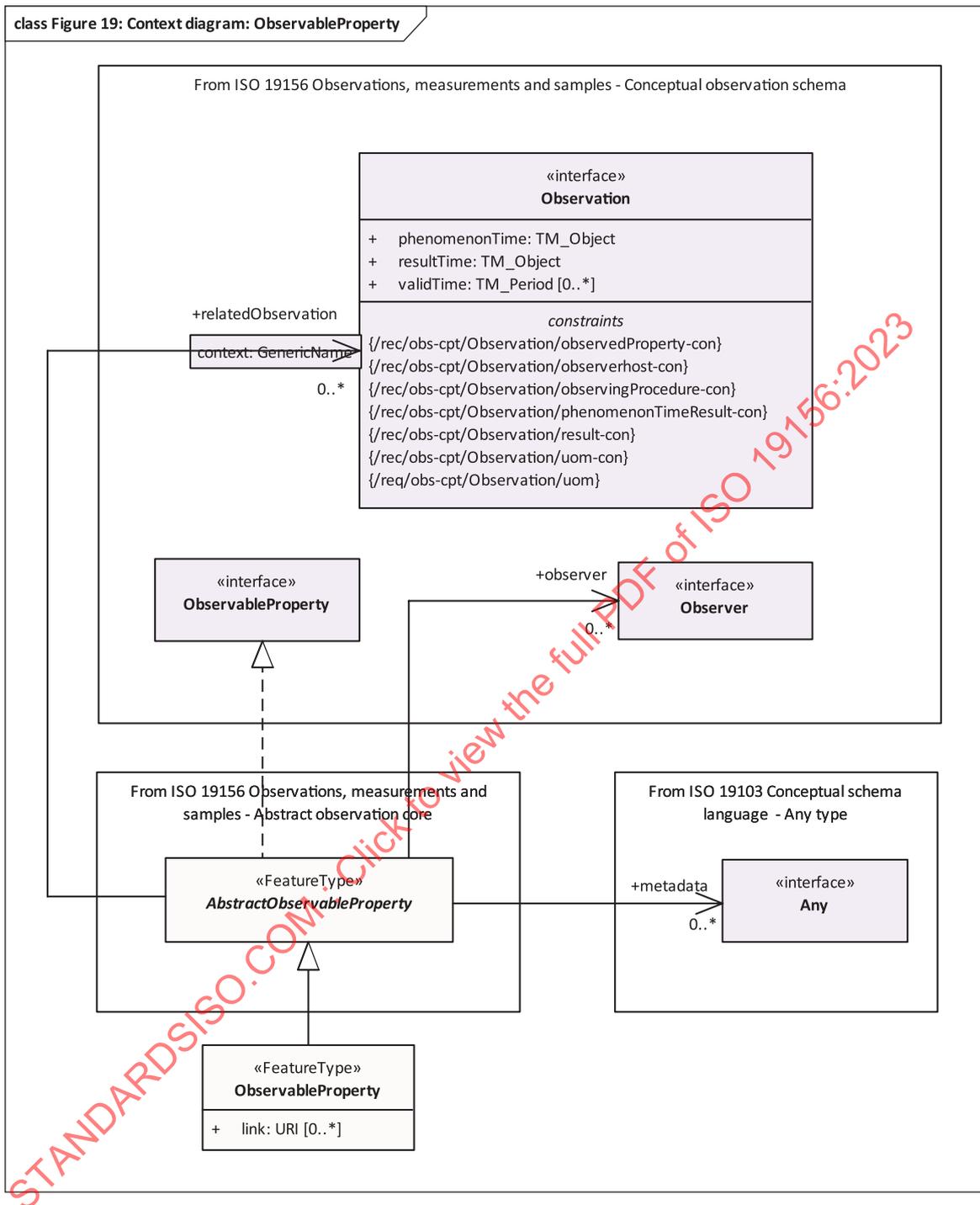


Figure 19 — Context diagram for the Basic Observations — ObservableProperty

## 10.7 ObservingProcedure

### 10.7.1 ObservingProcedure Requirements Class

Requirements Class	/req/obs-basic/ObservingProcedure
Target type	Logical model
Name	Basic Observations – ObservingProcedure



## 10.8 Observer

### 10.8.1 Observer Requirements Class

<b>Requirements Class</b>	/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

Observer from the Basic Observations is described as a class diagram in [Figure 21](#). The schema is fully described in [10.8](#).

class Figure 21: Context diagram Observer

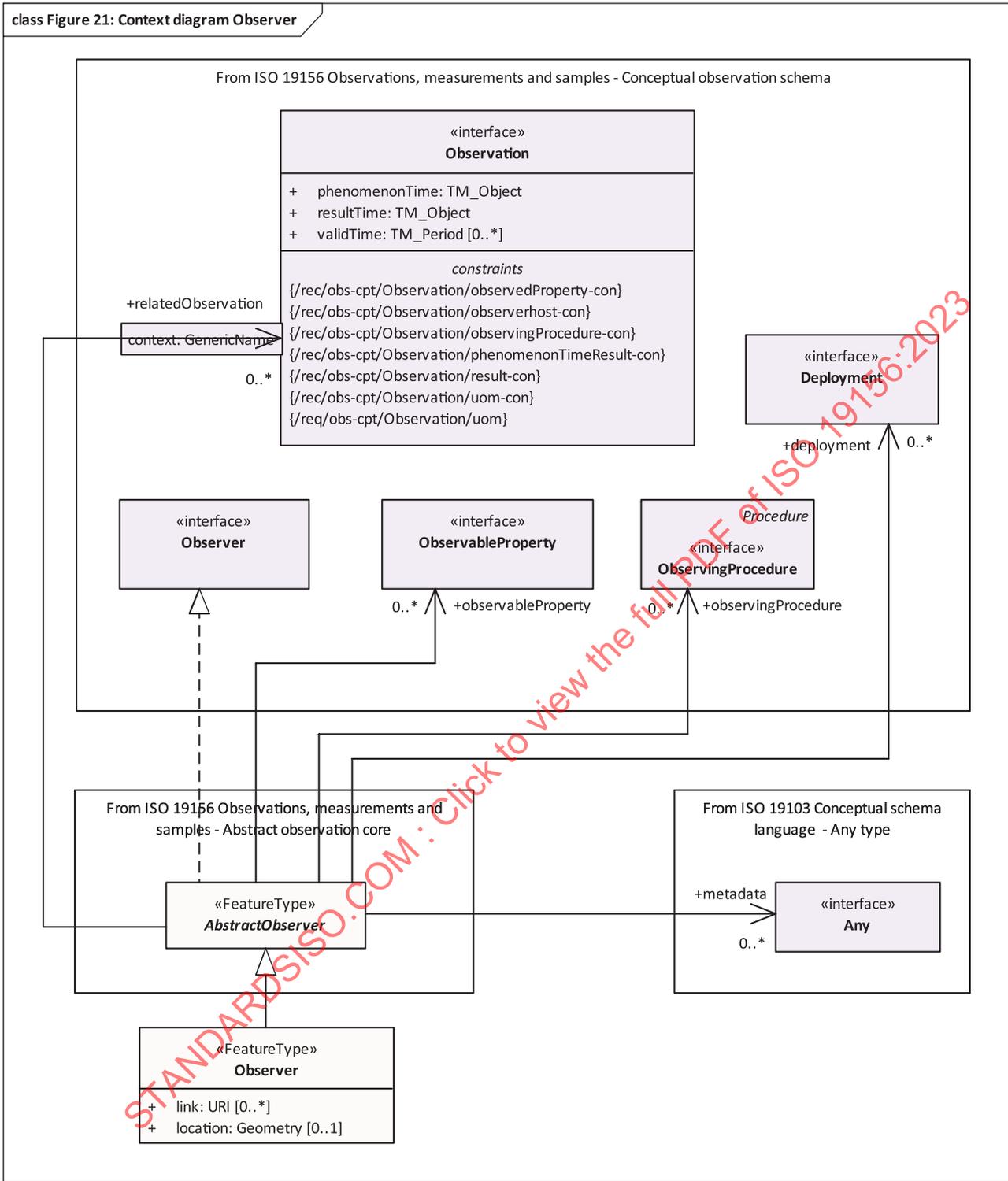


Figure 21 — Context diagram for Basic Observations — Observer

## 10.9 Host

### 10.9.1 Host Requirements Class

Requirements Class	/req/obs-basic/Host
Target type	Logical model

<b>Requirements Class</b>	/req/obs-basic/Host
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

Host from the Basic Observations is described as a class diagram in [Figure 22](#). The schema is fully described in [10.9](#).

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

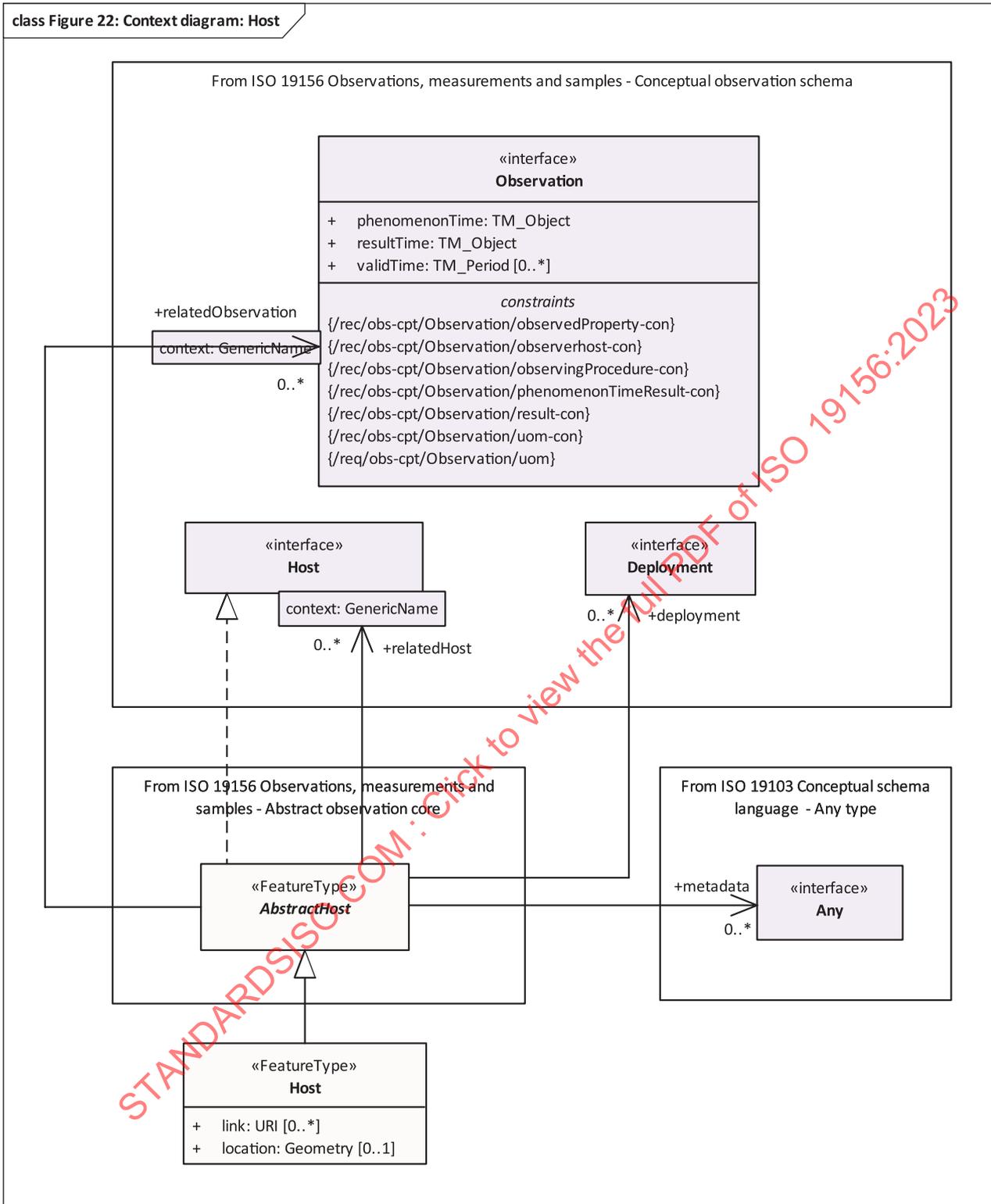


Figure 22 — Context diagram for Basic Observations — Host

## 10.10 Deployment

### 10.10.1 Deployment Requirements Class

<b>Requirements Class</b>	/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

Deployment from the Basic Observations is described as a class diagram in [Figure 23](#). The schema is fully described in [10.10](#).

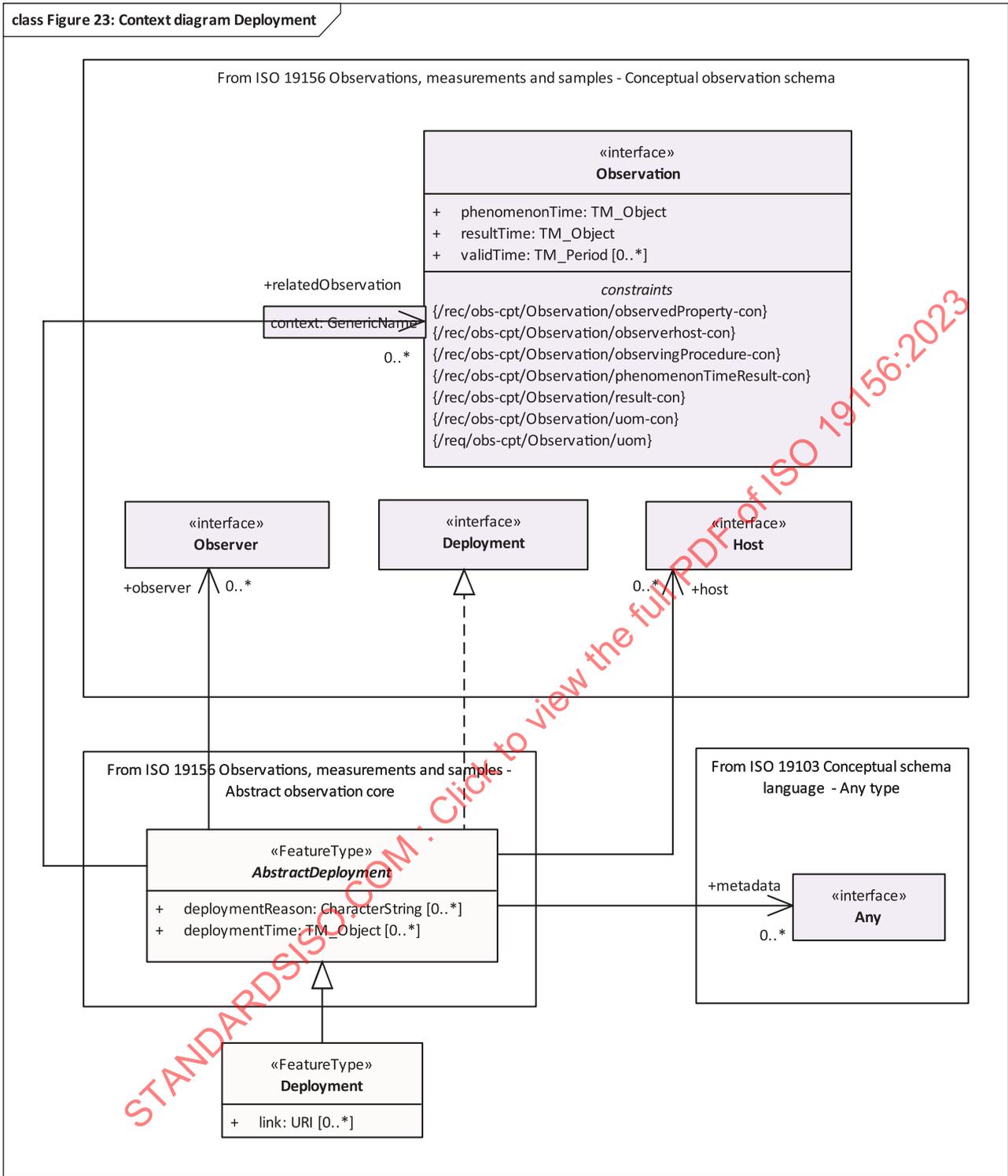


Figure 23 — Context diagram for Basic Observations — Deployment

### 10.11 GenericDomainFeature

#### 10.11.1 GenericDomainFeature Requirements Class

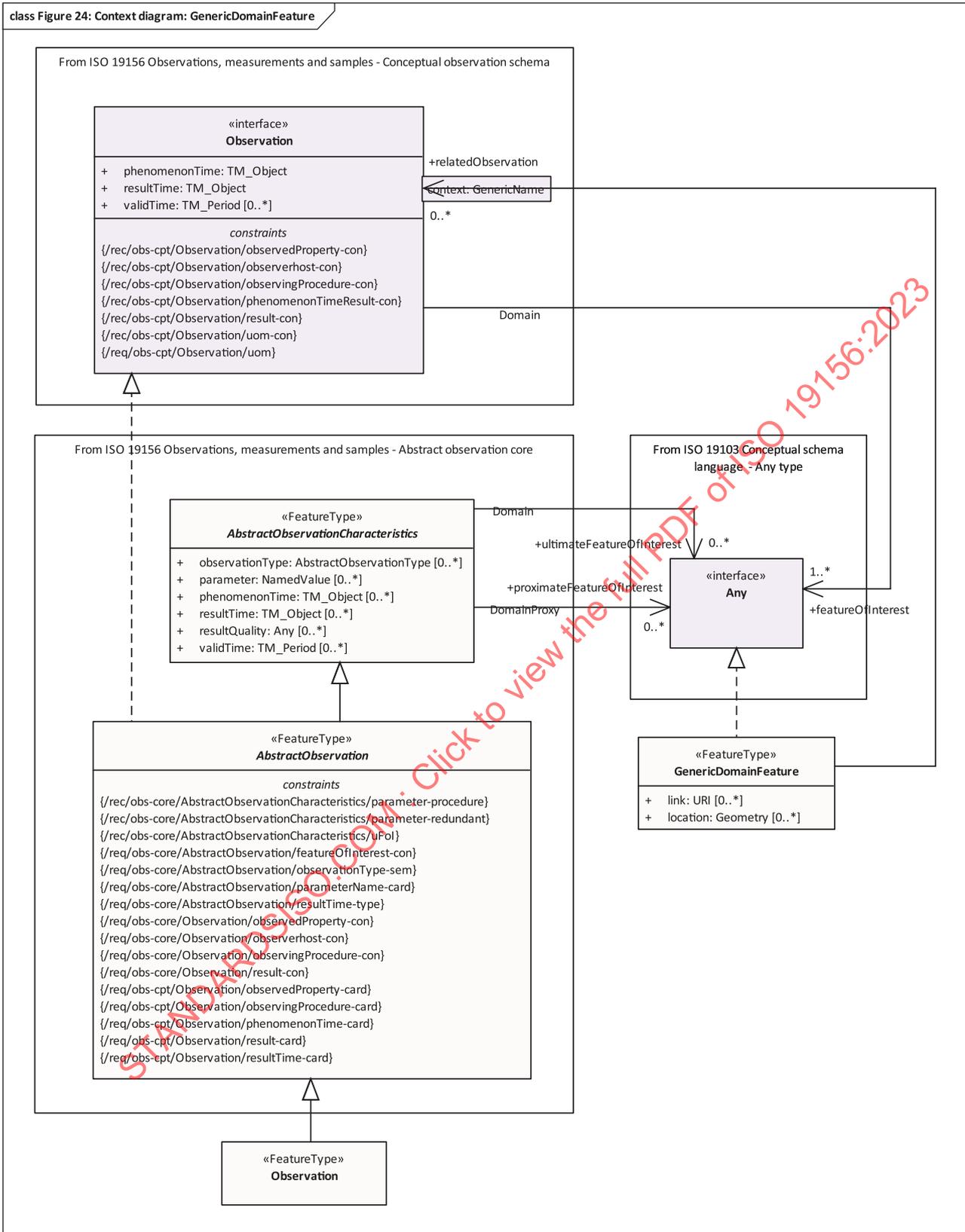
Requirements Class	/req/obs-basic/GenericDomainFeature
Target type	Logical model

<b>Requirements Class</b>	/req/obs-basic/GenericDomainFeature
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

GenericDomainFeature from the Basic Observations is described as a class diagram in [Figure 24](#). The schema is fully described in [10.11](#).

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

class Figure 24: Context diagram: GenericDomainFeature



NOTE GenericDomainFeature can be used as the target of the ultimate or proximate feature-of-interest of an Observation in the absence of an existing, more specific domain feature.

Figure 24 — Context diagram for Basic Observations — GenericDomainFeature

### 10.11.2 Feature type GenericDomainFeature

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

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

## 10.12 Codelists

### 10.12.1 ObservationCollectionType

The codelist ObservationCollectionType realizes the AbstractObservationCollectionType and has the following values defined in this document: “homogeneous” and “summarizing”.

<b>Requirement</b> /req/obs-basic/ObservationCollectionType/ObservationCollectionType-sem	The following entries shall be provided: <ul style="list-style-type: none"> <li>— <b>homogeneous</b>: all observations contained are of a similar nature;</li> <li>— <b>summarizing</b>: a wider grab-bag type of collection.</li> </ul>
--	--

<b>Requirement</b> /req/obs-basic/ObservationCollectionType/homogeneous-con	<p>If <b>collectionType</b> in the <b>ObservationCollection</b> is specified as “<b>homogeneous</b>” from this Codelist, the following constraints apply to the associated <b>ObservationCharacteristics</b> and all <b>Observation</b> instances referenced via the member association.</p> <p>If a property value is provided within the <b>ObservationCharacteristics</b>, this value applies to all <b>Observations</b> contained in the <b>ObservationCollection</b>:</p> <ul style="list-style-type: none"> <li>— property not provided – values may be provided by the <b>Observations</b> but is not provided at this level;</li> <li>— property provided but with no content – no <b>Observation</b> within the collection provides this property;</li> <li>— property = value – this value applies to all <b>Observations</b> within the collection;</li> <li>— property = value set/range – this value set/range applies to all <b>Observations</b> within the collection;</li> </ul>
--	--

NOTE The Observations need not contain attributes or associations supplied via the ObservationCharacteristics when collectionType is set to homogeneous.

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 will refer to that ObservableProperty.

<p><b>Requirement</b></p> <p>/req/obs-basic/ObservationCollectionType/ summarizing-con</p>	<p>If <b>collectionType</b> in the <b>ObservationCollection</b> is specified as “<b>summarizing</b>” from this Codelist, the following constraints apply to the associated <b>ObservationCharacteristics</b> and all <b>Observation</b> instances referenced via the member association.</p> <p>If multiple values for a property are available in the contained <b>Observations</b>, all values for this attribute (or the range of values contained in all <b>Observations</b>) are provided in the <b>ObservationCharacteristics</b>. A property may also be empty in the <b>ObservationCharacteristics</b> – in this case any value can be provided for this attribute within the contained <b>Observations</b>:</p> <ul style="list-style-type: none"> <li>— property not provided – values may be provided by the <b>Observations</b> but a summary is not provided at this level;</li> <li>— property provided but with no content – no <b>Observation</b> within the collection provides this property;</li> <li>— property = value – this value applies to all <b>Observations</b> within the collection;</li> <li>— property = value set/range – all <b>Observations</b> provide a value within this set/range.</li> </ul>
--	--

**NOTE** If a summarizing collection provides a set/range for an attribute, it can 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:

- a) Observations in the collection can have any value for the resultTime property, since it is absent from the collection;
- b) none of the Observations in the collection provide a value for validTime;

**NOTE** [empty/NIL/null] is a placeholder for the encoding specific representation of the absence of information.

- c) Observations can have any value for the phenomenonTime property that falls completely in the given time range. Valid examples would be:
  - 1) 2020-01-05T00:00:00+05:00;
  - 2) 2020-01-05T10:00:00Z/2020-01-05T11:00:00Z;
  - 3) 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:

- a) 1;
- b) 9;
- c) 2, 5 (a set with the two values);
- d) [8,1 -9,2] (a range of 8,1 to 9,2);

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

EXAMPLE 4 If the summarizing collection supplies:

- a) ultimateFeatureOfInterest=<https://example.org/collections/42/items/42>;
- b) deployment=[empty/NIL/null] (i.e. property provided but with no content);
- c) observer=[<https://example.org/v1.1/Sensors/41>, <https://example.org/v1.1/Sensors/43>].

then this means:

- the Observations in the collection all have the same ultimateFeatureOfInterest (a reference to <https://example.org/collections/42/items/42>);
- none of the Observations in the collection have a (reference to a) deployment;
- all Observations in the collection have either one, or both, of the referenced Observers;
- since the proximateFeatureOfInterest is not specified in the collection, the Observations in the collection can have any value for this field.

### 10.12.2 ObservationTypeByResultType

The codelist ObservationTypeByResultType is a specialization of AbstractObservationType created to support the legacy observation types from the previous version of this document.

<p><b>Requirement</b> /req/obs-basic/ObservationTypeByResultType/ObservationTypeByResultType-sem</p>	<p>The following entries shall be provided:</p> <ul style="list-style-type: none"> <li>— measurement: the result is of type <b>Measure</b>;</li> <li>— category-observation: the result is of type <b>ScopedName</b>;</li> <li>— truth-observation: result is a truth value;</li> <li>— count-observation: the result is of type <b>Integer</b>;</li> <li>— temporal-observation: the result is of type <b>TM_Object</b>;</li> <li>— geometry-observation: the result is of type <b>Geometry</b>;</li> <li>— complex-observation: the result is of type <b>Record</b>;</li> <li>— discrete-coverage-observation: result is a coverage that returns the same feature attribute values for every direct position within any single spatial object, temporal object, or spatiotemporal object in its domain;</li> <li>— discrete-point-coverage: result is a coverage that has a domain composed of points;</li> <li>— timeseries-observation: the result is a timeseries (a sequence of data values which are ordered in time).</li> </ul>
--	--

<p><b>Requirement</b> /req/obs-basic/ObservationTypeByResult- Type/ObservationTypeByResultType-con</p>	<p>The following constraints shall be applied to the value of the result association of the <b>Observation</b> based on the codelist value used:</p> <ul style="list-style-type: none"> <li>— If the value "measurement" is used, the value of the result shall be of type <b>Measure</b>;</li> <li>— If the value "category-observation" is used the value of the result shall be of type <b>ScopedName</b>;</li> <li>— If the value "truth-observation" is used, the value of result shall be a truth value;</li> <li>— If the value "count-observation" is used, the value of the result shall be of type <b>Integer</b>;</li> <li>— If the value "temporal-observation" is used, the value of the result shall be of type <b>TM_Object</b>;</li> <li>— If the value "geometry-observation" is used, the value of the result shall be of type <b>Geometry</b>;</li> <li>— If the value "complex-observation" is used, the value of the result shall be of type <b>Record</b>.</li> </ul>
--	---

## 11 Conceptual Sample schema

### 11.1 General

#### 11.1.1 Conceptual Sample schema model

The Conceptual Sample schema is described as a class diagram in [Figure 25](#). It is fully described in [11.1.2](#).



<b>Requirements Class</b>	/req/sam-cpt
Imports	/req/sam-cpt/PreparationStep
Imports	/req/sam-cpt/PreparationProcedure
Imports	/req/sam-cpt/SamplingProcedure

## 11.2 Sample

### 11.2.1 Sample Requirements Class

<b>Requirements Class</b>	/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, Core-Types 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/relatedSample-sem
Requirement	/req/obs-cpt/gen/relatedObservation-sem

### 11.2.2 Interface Sample

<b>Requirement</b> /req/sam-cpt/Sample/Sample-sem	A <b>Sample</b> shall be defined as an object that is representative of a concept, real-world object or phenomenon.
--	---

NOTE 1 The way the sample is taken is typically guided by a sampling strategy. Samples are often artefacts of an observational strategy, and often have no significant function outside of their role in the observation process (although specimen preservation could be considered a specific activity per se).

NOTE 2 The physical characteristics of the features themselves are of little interest, except perhaps to the manager of a sampling campaign.

NOTE 3 Typically, the Sample is a Feature which is intended to be representative of a FeatureOfInterest on which Observations can be made. As such, it can 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

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

### 11.2.4 Association preparationStep

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

### 11.2.5 Association sampledFeature

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

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

<b>Requirement</b> /req/sam-cpt/Sample/relatedSample-sem	A <b>Sample</b> the <b>Sample</b> is related to. If a reference to a related <b>Sample</b> is provided, the association with role <b>relatedSample</b> shall be used. The <b>context:GenericName</b> 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 samples 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

<b>Requirements Class</b>	/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, Core-Types 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

Requirements Class	/req/sam-cpt/Sampling
Requirement	/req/obs-basic/gen/link-sem

### 11.3.2 Interface Sampling

Requirement /req/sam-cpt/Sampling/Sampling-sem	A <b>Sampling</b> shall be defined as an act applying a <b>SamplingProcedure</b> to create or transform one or more <b>Sample(s)</b> .
---	--

EXAMPLE 1 Crushing a rock sample in a ball mill.

EXAMPLE 2 Digging a pit through a soil sequence.

EXAMPLE 3 Dividing a field site into quadrants.

EXAMPLE 4 Drawing blood from a patient.

EXAMPLE 5 Extracting water from an observation well.

EXAMPLE 6 Extracting a sample from a defined environmental monitoring station.

EXAMPLE 7 Registering an image of the landscape.

EXAMPLE 8 Sieving a powder to separate the subset finer than 100-mesh.

EXAMPLE 9 Selecting a subset of a population.

EXAMPLE 10 Splitting a piece of drill-core to create two new samples.

EXAMPLE 11 Taking a diamond-drill core from a rock outcrop.

### 11.3.3 Association sample

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

### 11.3.4 Association featureOfInterest

Requirement /req/sam-cpt/Sampling/featureOfInterest-sem	A <b>feature-of-interest</b> shall be defined as the concept, real-world object or phenomenon (feature-of-interest) the <b>Sample(s)</b> of the <b>Sampling</b> represent. Reference to the feature-of-interest shall be done using the association with the role <b>featureOfInterest</b> .
--	---

Requirement /req/sam-cpt/Sampling/featureOfInterest-card	A <b>Sampling</b> shall have at least 1 <b>featureOfInterest</b> and may have more than 1 in cases where multiple objects are sampled with the intention to create one <b>Sample</b> .
---	--

### 11.3.5 Association sampler

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

### 11.3.6 Association **samplingProcedure**

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

### 11.3.7 Association **relatedSampling**

<b>Requirement</b> /req/sam-cpt/Sampling/relatedSampling-sem	Related <b>Sampling(s)</b> . If a reference to a related <b>Sampling</b> is provided, the association with role <b>relatedSampling</b> shall be used. The <b>context:GenericName</b> 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

<b>Requirements Class</b>	/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
Requirement	/req/obs-basic/gen/link-sem

### 11.4.2 Interface **Sampler**

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

EXAMPLE 1 A ball mill, diamond drill, hammer.

EXAMPLE 2 A hypodermic syringe and needle.

EXAMPLE 3 An image sensor, a soil auger.

EXAMPLE 4 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 is not necessarily a physical device.

### 11.4.3 Association **sampling**

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

#### 11.4.4 Association implementedProcedure

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

### 11.5 PreparationStep

#### 11.5.1 PreparationStep Requirements Class

<b>Requirements Class</b>	/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
Requirement	/req/obs-basic/gen/link-sem

#### 11.5.2 Interface PreparationStep

<b>Requirement</b> /req/sam-cpt/PreparationStep/PreparationStep-sem	A <b>PreparationStep</b> shall be defined as an individual step pertaining to a <b>PreparationProcedure</b> .
--	---

#### 11.5.3 Association processingDetails

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

#### 11.5.4 Association preparedSample

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

### 11.6 PreparationProcedure

#### 11.6.1 PreparationProcedure Requirements Class

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

<b>Requirements Class</b>	/req/sam-cpt/PreparationProcedure
Imports	/req/obs-cpt/Procedure
Requirement	/req/sam-cpt/PreparationProcedure/PreparationProcedure-sem
Requirement	/req/sam-cpt/PreparationProcedure/samplePreparationStep-sem
Requirement	/req/obs-basic/gen/link-sem

### 11.6.2 Interface PreparationProcedure

<b>Requirement</b> /req/sam-cpt/PreparationProcedure/PreparationProcedure-sem	A <b>PreparationProcedure</b> shall be defined as the description of preparation steps performed on a <b>Sample</b> .
--	---

### 11.6.3 Association samplePreparationStep

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

## 11.7 SamplingProcedure

### 11.7.1 SamplingProcedure Requirements Class

<b>Requirements Class</b>	/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
Requirement	/req/obs-basic/gen/link-sem

### 11.7.2 Interface SamplingProcedure

<b>Requirement</b> /req/sam-cpt/SamplingProcedure/SamplingProcedure-sem	A <b>SamplingProcedure</b> shall be defined as the description of steps performed by a <b>Sampler</b> in order to extract a <b>Sample</b> from its <b>sampledFeature</b> in the frame of a <b>Sampling</b> .
--	--

### 11.7.3 Association sampling

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

11.7.4 Association sampler

<b>Requirement</b> /req/sam-cpt/SamplingProcedure/sampler-sem	If the <b>SamplingProcedure</b> provides information on the <b>Sampler</b> that implements this procedure, the association with the role <b>sampler</b> shall be used.
--	--

12 Abstract Sample Core

12.1 General

12.1.1 Abstract Sample Core Package Requirements

<b>Requirements Class</b>	/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

12.2 AbstractSample

12.2.1 AbstractSample Requirements Class

<b>Requirements Class</b>	/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
Requirement	/req/sam-core/AbstractSampleType/AbstractSampleType-sem

AbstractSample from the Abstract Sample Core is described as a class diagram in [Figure 26](#). The schema is fully described in [12.2](#).

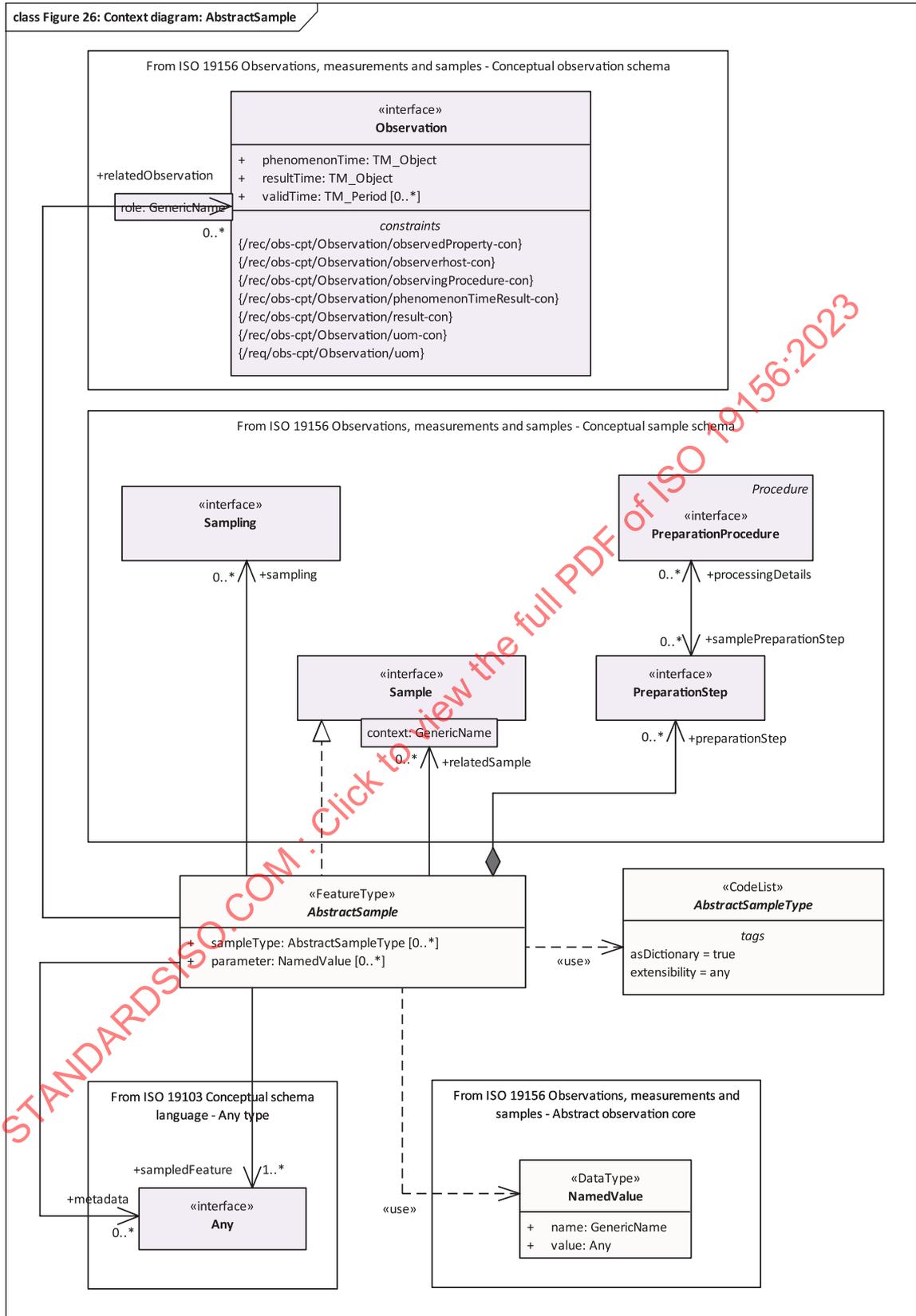


Figure 26 — Context diagram for Abstract Sample Core — AbstractSample

### 12.2.2 Attribute sampleType

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

### 12.2.3 Attribute parameter

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

**EXAMPLE** When taking water samples, the sampling procedure specifies the amount of time that needs to pass to allow sediments to settle. As reality is rarely as exact as plans, the actual waiting time applied to a specific sample can be stored in the parameter.

**NOTE** Using the classes, attributes and associations explicitly modelled in the OMS greatly improves the interoperability compared to using the generic parameter mechanism to include the same information.

## 12.3 AbstractSampling

### 12.3.1 AbstractSampling Requirements Class

<b>Requirements Class</b>	/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

AbstractSampling from the Abstract Sample Core is described as a class diagram in [Figure 27](#). The schema is fully described in [12.3](#).

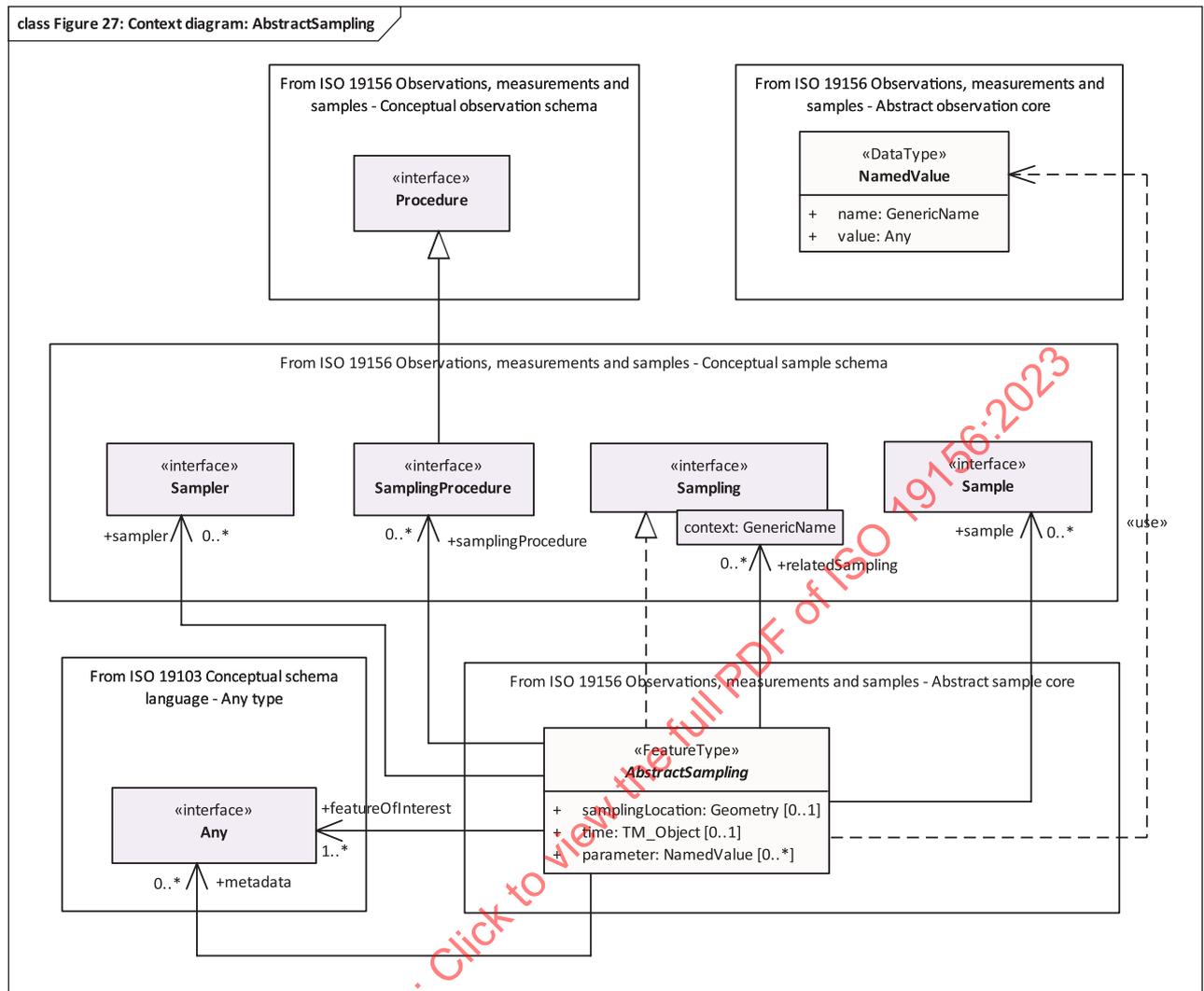


Figure 27 — Context diagram for Abstract Sample Core — AbstractSampling

### 12.3.2 Attribute samplingLocation

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

### 12.3.3 Attribute time

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

### 12.3.4 Attribute parameter

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

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

**NOTE** Using the classes, attributes and associations explicitly modelled in the OMS greatly improves the interoperability compared to using the generic parameter mechanism to include the same information.

## 12.4 AbstractSampler

### 12.4.1 AbstractSampler Requirements Class

<b>Requirements Class</b>	/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
Requirement	/req/sam-core/AbstractSamplerType/AbstractSamplerType-sem

AbstractSampler from the Abstract Sample Core is described as a class diagram in [Figure 28](#). The schema is fully described in [12.4](#).

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

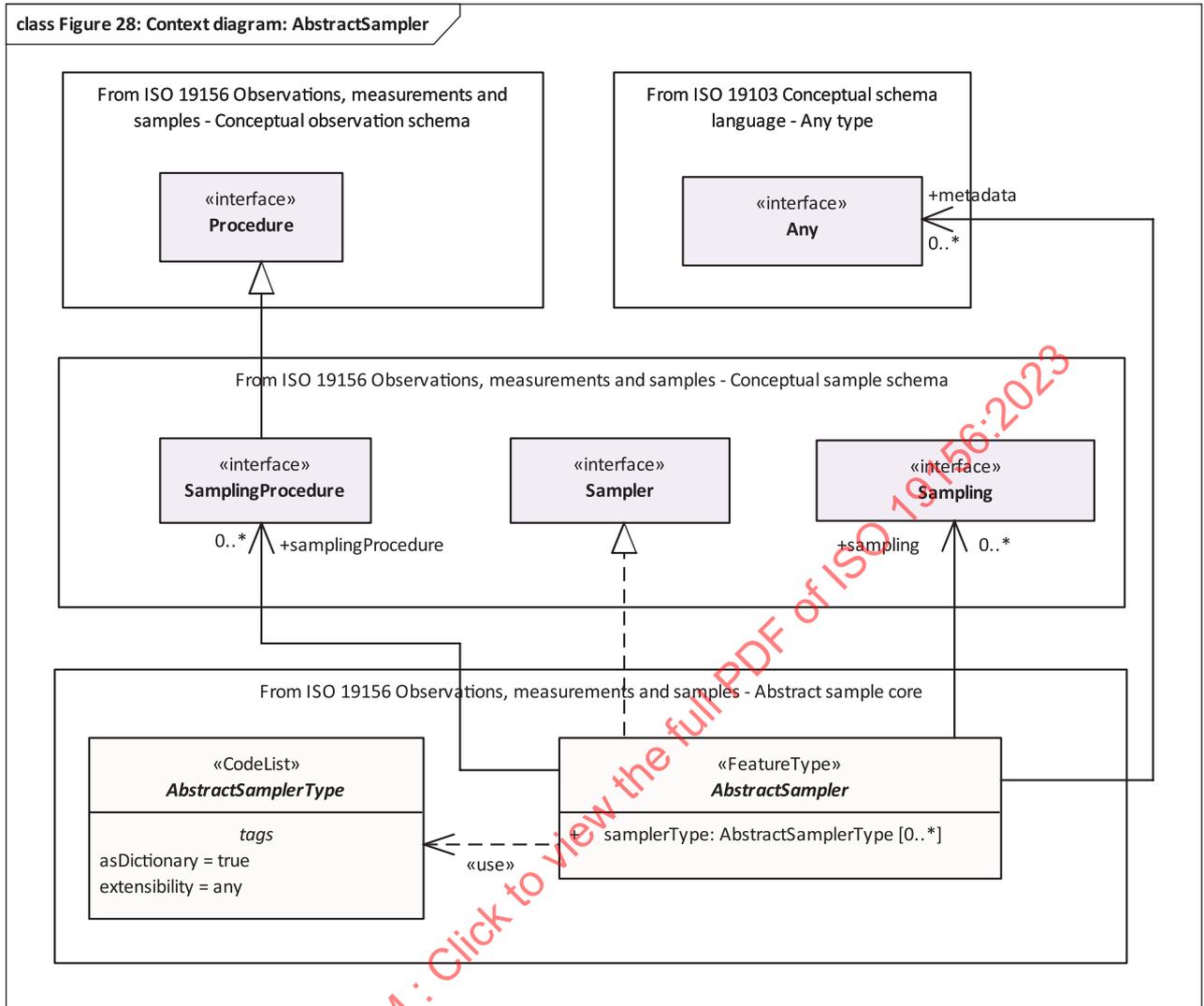


Figure 28 — Context diagram for the Abstract Sample Core — AbstractSampler

### 12.4.2 Attribute **samplerType**

<p><b>Requirement</b> /req/sam-core/AbstractSampler/sampler-Type-sem</p>	<p>The type of <b>Sampler</b> according to a community agreed typology.</p> <p>If information on the type of <b>AbstractSampler</b> is provided, the attribute <b>samplerType: AbstractSamplerType</b> shall be used.</p>
--	---

- EXAMPLE 1 A ball mill, diamond drill, hammer.
- EXAMPLE 2 A hypodermic syringe and needle.
- EXAMPLE 3 An image sensor, a soil auger.
- EXAMPLE 4 A human being.

## 12.5 AbstractSamplingProcedure

### 12.5.1 AbstractSamplingProcedure Requirements Class

<b>Requirements Class</b>	/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

AbstractSamplingProcedure from the Abstract Sample Core is described as a class diagram in [Figure 29](#). The schema is fully described in [12.5](#).

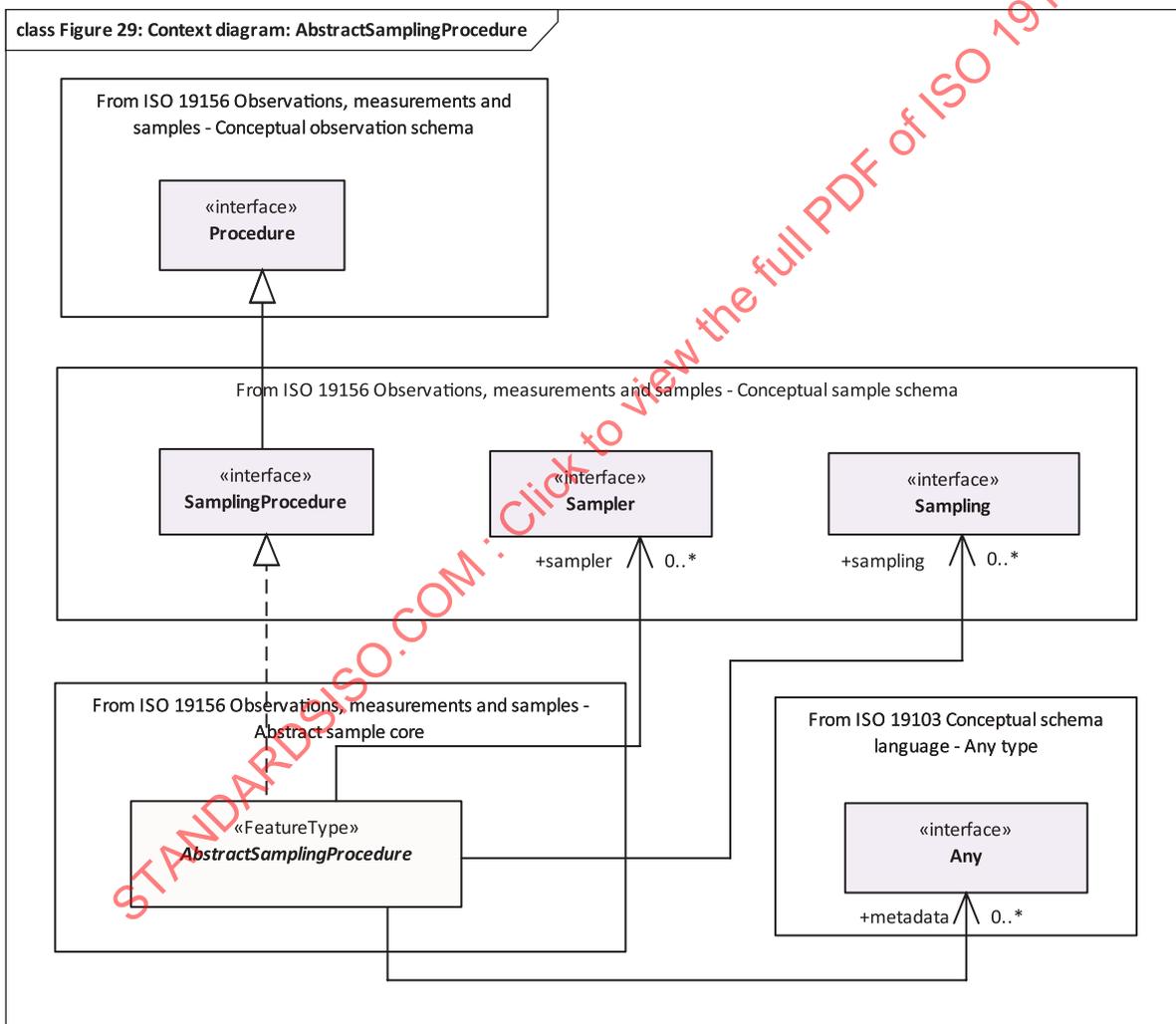


Figure 29 — Context diagram for Abstract Sample Core — AbstractSamplingProcedure

## 12.6 AbstractPreparationProcedure

### 12.6.1 AbstractPreparationProcedure Requirements Class

<b>Requirements Class</b>	/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

AbstractPreparationProcedure and AbstractPreparationStep from the Abstract Sample Core are described as a class diagram in [Figure 30](#). The schema is fully described in [12.6](#).

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

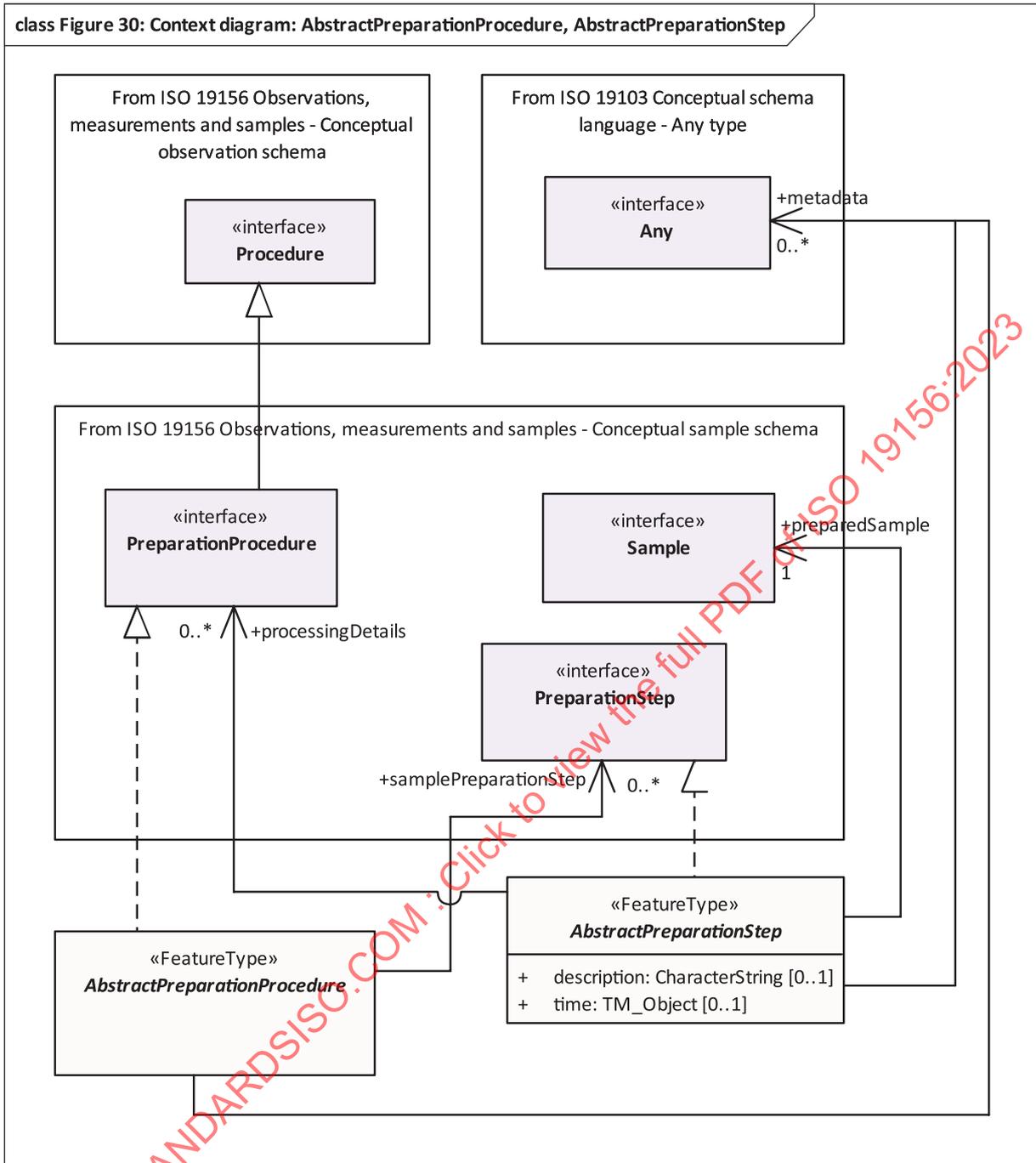


Figure 30 — Context diagram for Abstract Sample Core —AbstractPreparationProcedure and AbstractPreparationStep

## 12.7 AbstractPreparationStep

### 12.7.1 AbstractPreparationStep Requirements Class

Requirements Class	/req/sam-core/AbstractPreparationStep
Target type	Logical model
Name	Abstract Sample Core – AbstractPreparationStep

<b>Requirements Class</b>	/req/sam-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

### 12.7.2 Attribute description

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

### 12.7.3 Attribute time

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

## 12.8 Codelists

### 12.8.1 AbstractSampleType

The codelist AbstractSampleType can be specialized as required to more precisely define the semantics of sample types.

<b>Requirement</b> /req/sam-core/AbstractSampleType/AbstractSampleType-sem	An empty extension-point for providing various classification schemes for <b>Samples</b> . If <b>Sample</b> classification schemes are used in the implementing application schemas, a concrete realization shall be created for the application.
---	--

### 12.8.2 AbstractSamplerType

The codelist AbstractSamplerType can be specialized as required to more precisely define the semantics of sampler types.

<b>Requirement</b> /req/sam-core/AbstractSamplerType/AbstractSamplerType-sem	An empty extension-point for providing various classification schemes for <b>Samplers</b> . If <b>Sampler</b> classification schemes are used in the implementing application schemas, a concrete realization shall be created for the application.
---	--

## 13 Basic Samples

### 13.1 General

#### 13.1.1 Basic Samples Package Requirements Class

<b>Requirements Class</b>	/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/SamplingProcedure
Imports	/req/sam-basic/PreparationProcedure
Imports	/req/sam-basic/PreparationStep
Imports	/req/sam-basic/SampleCollection
Requirement	/req/sam-basic/SampleTypeByGeometryType/SampleTypeByGeometryType-sem
Requirement	/req/sam-basic/SampleTypeByGeometryType/SampleTypeByGeometryType-con

## 13.2 Sample

### 13.2.1 Sample Requirements Class

<b>Requirements Class</b>	/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

Sample, SpatialSample, StatisticalSample and MaterialSample from the Basic Samples are described as a class diagram in [Figure 31](#). The schema is fully described in [13.2](#), [13.3](#), [13.4](#) and [13.5](#).

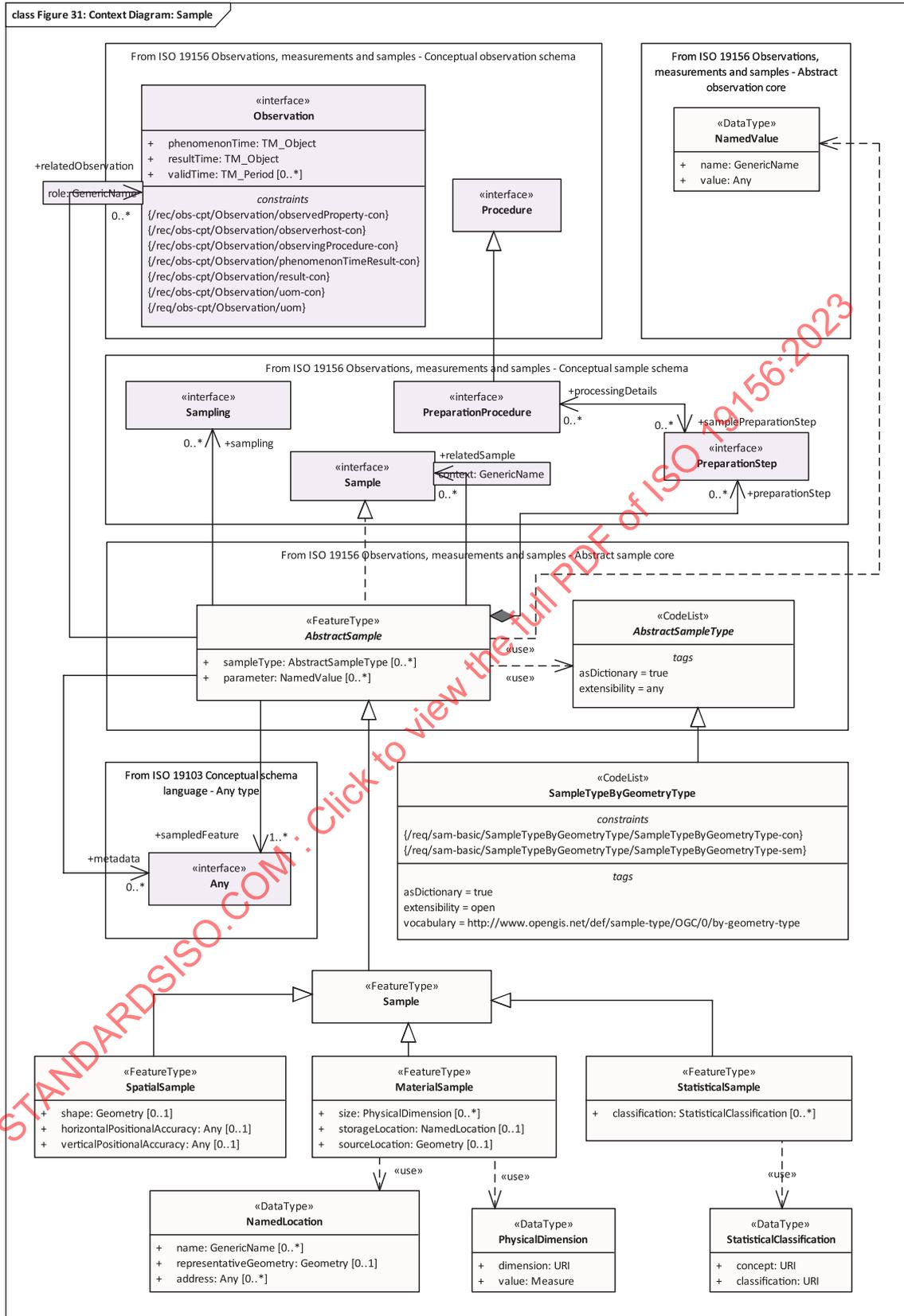


Figure 31 — Context diagram for Basic Samples — Sample, SpatialSample, StatisticalSample and MaterialSample

### 13.3 SpatialSample

#### 13.3.1 SpatialSample Requirements Class

<b>Requirements Class</b>	/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, Core-Types 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

#### 13.3.2 Feature type SpatialSample

<b>Requirement</b> /req/sam-basic/SpatialSample/SpatialSample-sem	A <b>SpatialSample</b> shall be defined as a geospatial <b>Sample</b> .
--	---

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 can be extensive in one, two or three spatial dimensions.

EXAMPLE 1 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.

EXAMPLE 2 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

<b>Requirement</b> /req/sam-basic/SpatialSample/shape-sem	The <b>shape</b> is the Geometry of the <b>SpatialSample</b> . If location information pertaining to the <b>SpatialSample</b> is provided, the attribute <b>shape:Geometry</b> 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 can use different intervals, and sub-samples can 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

<b>Requirement</b> /req/sam-basic/SpatialSample/horizontalPositionalAccuracy-sem	The positional accuracy of the horizontal component of the Geometry of the <b>SpatialSample</b> .  If horizontal positional accuracy information pertaining to the <b>SpatialSample</b> is provided, the attribute <b>horizontalPositionalAccuracy:Any</b> shall be used.
---	---

### 13.3.5 Attribute verticalPositionalAccuracy

<b>Requirement</b> /req/sam-basic/SpatialSample/verticalPositionalAccuracy-sem	The positional accuracy of the vertical component of the Geometry of the <b>SpatialSample</b> .  If horizontal positional accuracy information pertaining to the <b>SpatialSample</b> is provided, the attribute <b>verticalPositionalAccuracy:Any</b> shall be used.
---	---

## 13.4 MaterialSample

### 13.4.1 MaterialSample Requirements Class

<b>Requirements Class</b>	/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

### 13.4.2 Feature type MaterialSample

<b>Requirement</b> /req/sam-basic/MaterialSample/MaterialSample-sem	A <b>MaterialSample</b> shall be defined as a physical, tangible <b>Sample</b> .
--	--

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

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

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

### 13.4.4 Attribute storageLocation

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

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

### 13.4.5 Attribute sourceLocation

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

NOTE 1 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 such location information.

NOTE 2 The attribute sourceLocation of the MaterialSample can be unnecessary in the case that the related Sampling act samplingLocation attribute is provided.

## 13.5 StatisticalSample

### 13.5.1 StatisticalSample Requirements Class

<b>Requirements Class</b>	/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

### 13.5.2 Feature type StatisticalSample

<b>Requirement</b> /req/sam-basic/StatisticalSample/Statistical-Sample-sem	A <b>StatisticalSample</b> shall be defined as a statistical subset of a feature-of-interest, defined for the purpose of creating <b>Observation(s)</b> .
---	---

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

EXAMPLE The male or female subset of a population.

### 13.5.3 Attribute classification

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

NOTE The classification can 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

<b>Requirements Class</b>	/req/sam-basic/Sampling
Target type	Logical model
Name	Basic Samples – Sampling
Dependency	ISO 19103:2015, Geographic information — Conceptual schema language, UML2 conformance class
Imports	/req/sam-core/AbstractSampling
Requirement	/req/obs-basic/gen/link-sem

Sampling from the Basic Samples is described as a class diagram in [Figure 32](#).

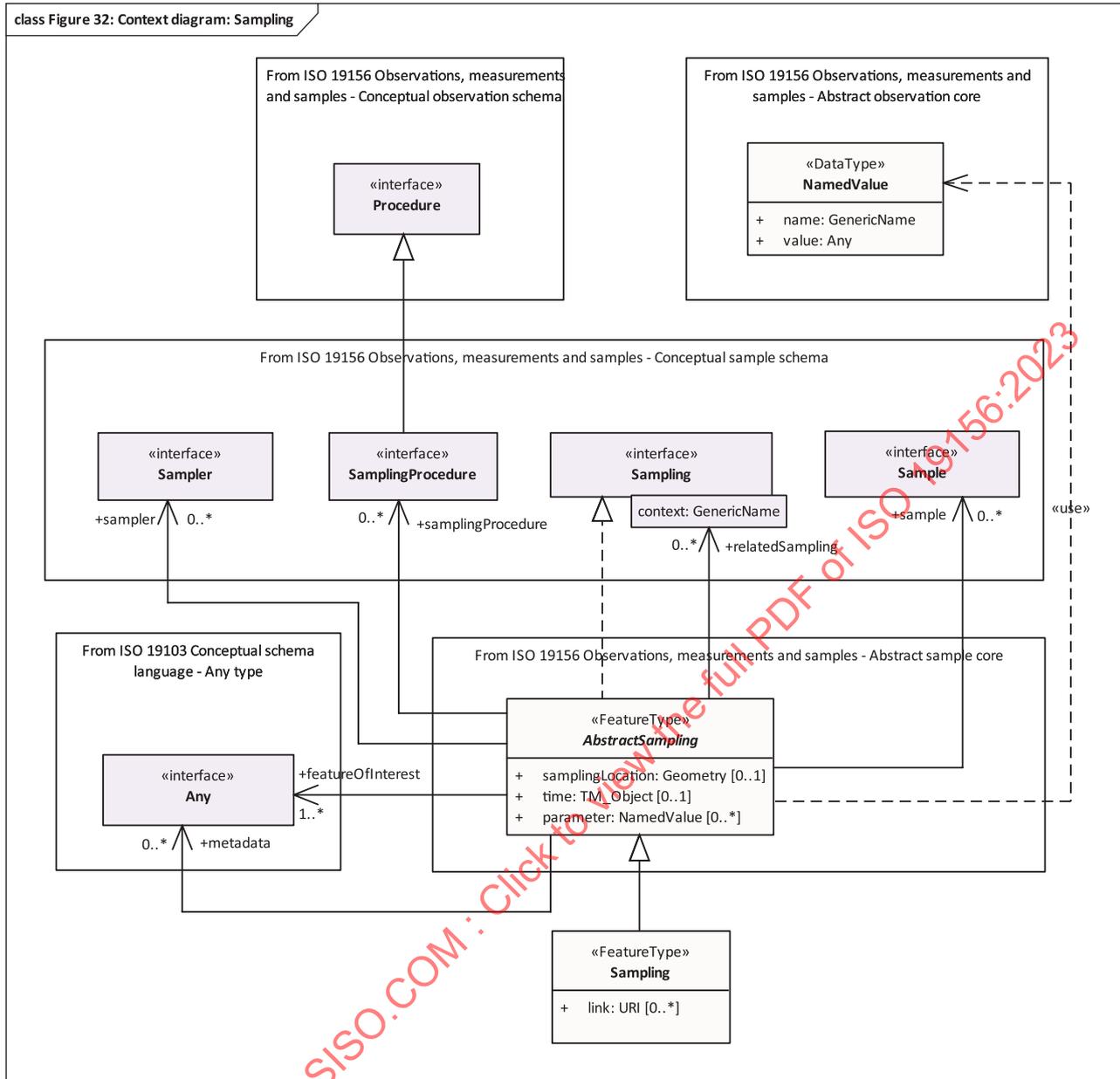


Figure 32 — Context diagram for Basic Samples — Sampling

## 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
Requirement	/req/obs-basic/gen/link-sem

Sampler from the Basic Samples is described as a class diagram in [Figure 33](#).

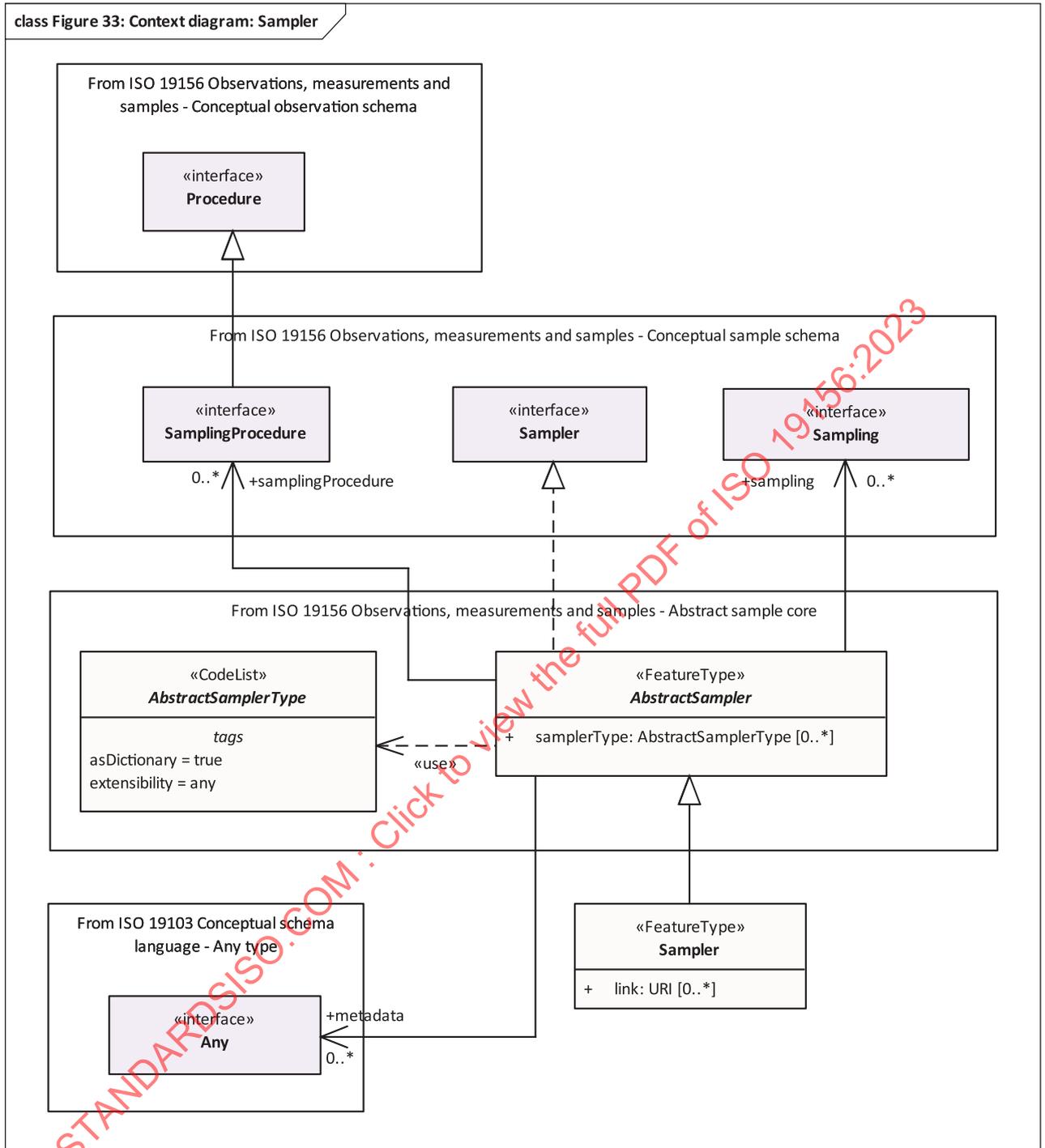


Figure 33 — Context diagram for Basic Samples — Sampler

## 13.8 SamplingProcedure

### 13.8.1 SamplingProcedure Requirements Class

Requirements Class	/req/sam-basic/SamplingProcedure
Target type	Logical model
Name	Basic Samples – SamplingProcedure

Requirements Class	/req/sam-basic/SamplingProcedure
Dependency	ISO 19103:2015, Geographic information — Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015, Geographic information — Conceptual schema language, Core-Types conformance class
Imports	/req/sam-core/AbstractSamplingProcedure
Requirement	/req/obs-basic/gen/link-sem

SamplingProcedure from the Basic Samples is described as a class diagram in [Figure 34](#).

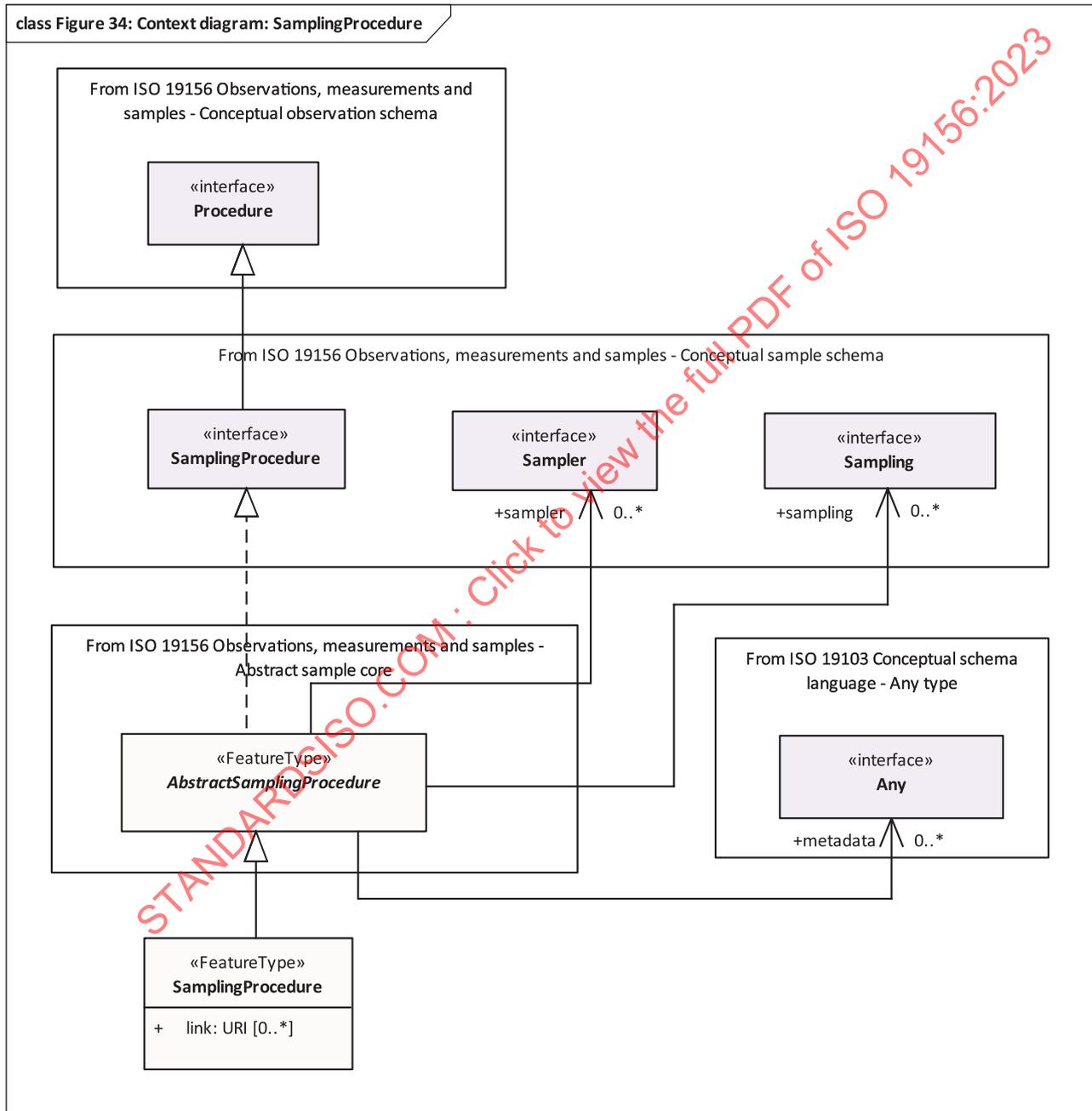


Figure 34 — Context diagram for Basic Samples — SamplingProcedure

## 13.9 PreparationProcedure

### 13.9.1 PreparationProcedure Requirements Class

<b>Requirements Class</b>	/req/sam-basic/PreparationProcedure
Target type	Logical model
Name	Basic Samples – PreparationProcedure
Dependency	ISO 19103:2015, Geographic information — Conceptual schema language, UML2 conformance class
Dependency	ISO 19103:2015, Geographic information — Conceptual schema language, Core-Types conformance class
Imports	/req/sam-core/AbstractPreparationProcedure
Requirement	/req/obs-basic/gen/link-sem

PreparationProcedure from the Basic Samples is described as a class diagram in [Figure 35](#).

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

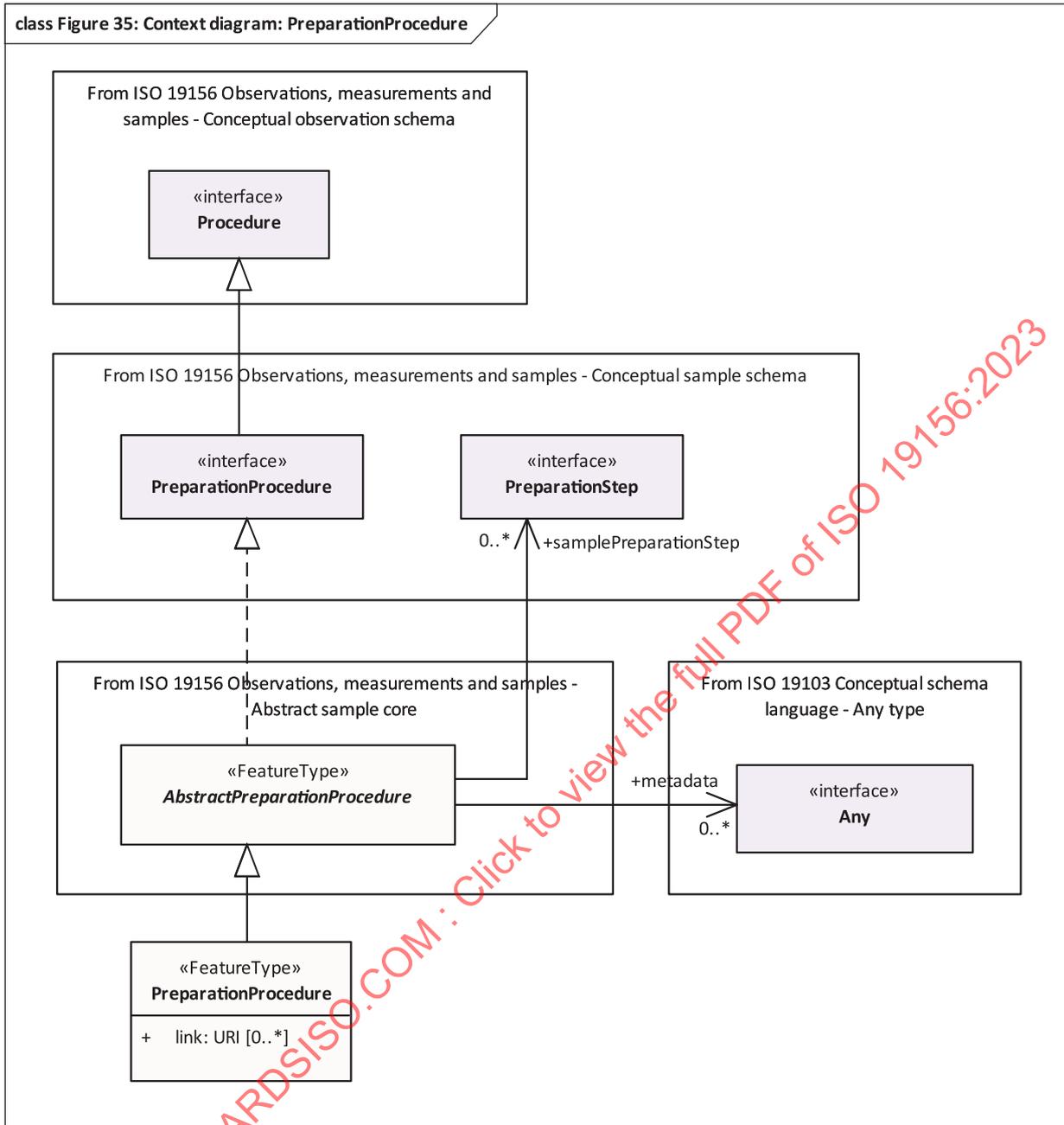


Figure 35 — Context diagram for Basic Samples — PreparationProcedure

### 13.10 PreparationStep

#### 13.10.1 PreparationStep Requirements Class

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

Requirements Class	/req/sam-basic/PreparationStep
Imports	/req/sam-core/AbstractPreparationStep
Requirement	/req/obs-basic/gen/link-sem

PreparationStep from the Basic Samples is described as a class diagram in [Figure 36](#).

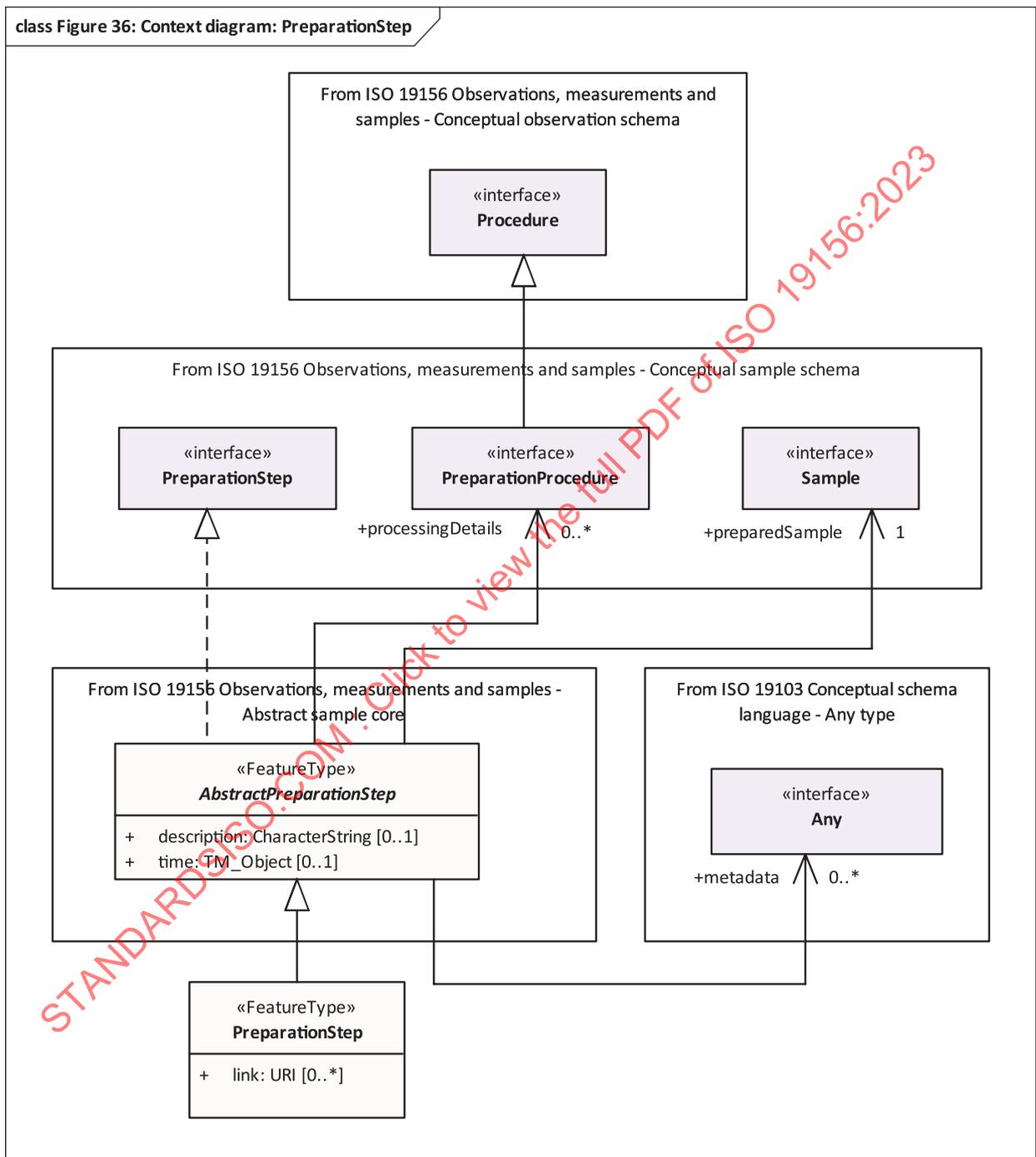


Figure 36 — Context diagram for Basic Samples — PreparationStep

### 13.11 SampleCollection

#### 13.11.1 SampleCollection Requirements Class

<b>Requirements Class</b>	/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, Core-Types 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

SampleCollection from the Basic Samples is described as a class diagram in [Figure 37](#). The schema is fully described in [13.11.2](#), [13.11.3](#) and [13.11.4](#).

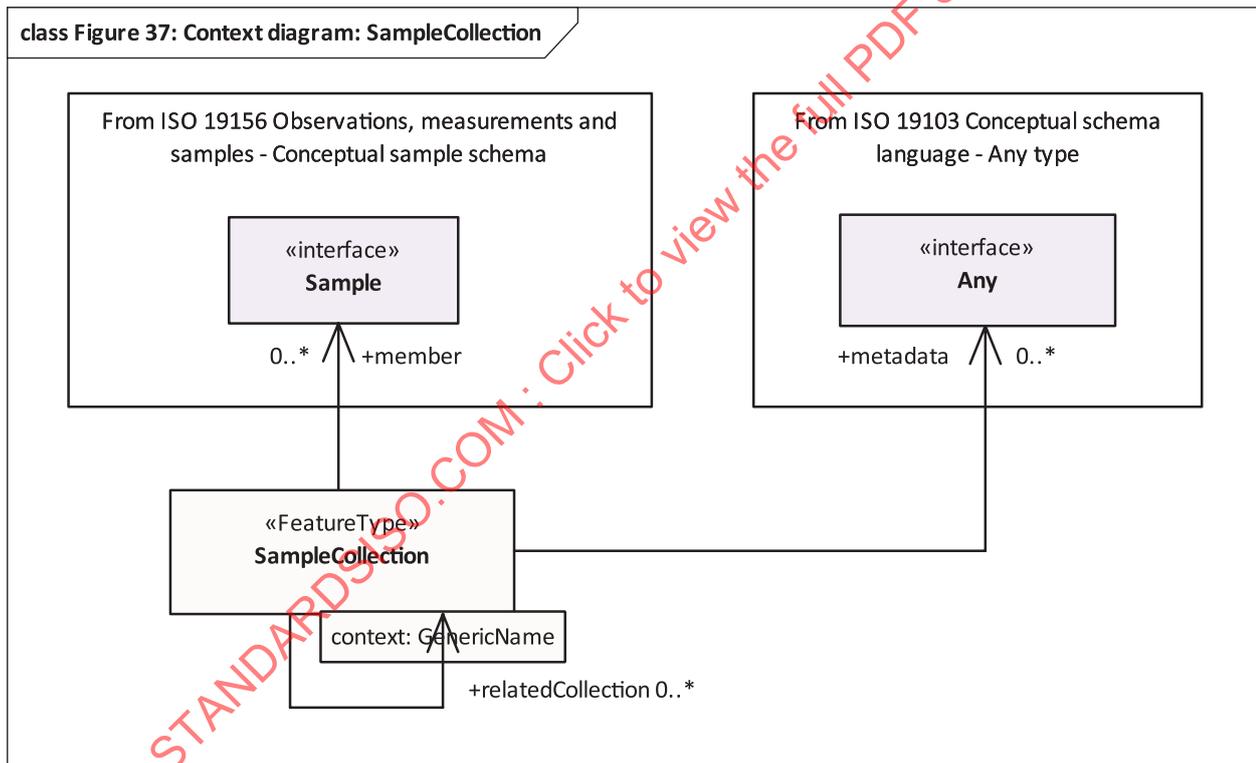


Figure 37 — Context diagram for Basic Samples — SampleCollection

#### 13.11.2 Feature type SampleCollection

<b>Requirement</b> /req/sam-basic/SampleCollection/SampleCollection-sem	A <b>SampleCollection</b> shall be defined as a collection of <b>Samples</b> .
--	--

### 13.11.3 Association member

<b>Requirement</b> /req/sam-basic/SampleCollection/member-sem	A <b>Sample</b> that is part of this <b>SampleCollection</b> . If the <b>SampleCollection</b> has members, the association with the role <b>member</b> shall be used.
--	--

### 13.11.4 Association relatedCollection

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

## 13.12 PhysicalDimension

### 13.12.1 PhysicalDimension Requirements Class

<b>Requirements Class</b>	/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

### 13.12.2 Data type PhysicalDimension

<b>Requirement</b> /req/sam-basic/PhysicalDimension/PhysicalDimension-sem	A <b>PhysicalDimension</b> shall be defined as a <b>dataType</b> for the provision of various size quantities.
--	--

### 13.12.3 Attribute dimension

<b>Requirement</b> /req/sam-basic/PhysicalDimension/dimension-sem	The name of the <b>PhysicalDimension</b> about which a <b>value</b> is provided. The identifier of the physical dimension shall be provided in the attribute <b>dimension:URI</b> .
--	--

### 13.12.4 Attribute value

<b>Requirement</b> /req/sam-basic/PhysicalDimension/value-sem	The <b>value</b> of the <b>PhysicalDimension</b> . The measure of the quantity being provided shall be provided in the attribute <b>value:Measure</b>
--	--

### 13.13 NamedLocation

#### 13.13.1 NamedLocation Requirements Class

<b>Requirements Class</b>	/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

#### 13.13.2 Data type NamedLocation

<b>Requirement</b> /req/sam-basic/NamedLocation/NamedLocation-sem	A <b>NamedLocation</b> shall be defined as a location identified by its name, address, spatial geometry or a combination of any of these three.
--	---

#### 13.13.3 Attribute address

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

#### 13.13.4 Attribute name

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

#### 13.13.5 Attribute representativeGeometry

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

### 13.14 StatisticalClassification

#### 13.14.1 StatisticalClassification Requirements Class

<b>Requirements Class</b>	/req/sam-basic/StatisticalClassification
Target type	Logical model
Name	Basic Samples – StatisticalClassification

<b>Requirements Class</b>	/req/sam-basic/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

### 13.14.2 Data type StatisticalClassification

<b>Requirement</b> /req/sam-basic/StatisticalClassification/StatisticalClassification-sem	A <b>StatisticalClassification</b> shall be defined as a <b>dataType</b> for the provision of information on statistical classifications.
--	---

### 13.14.3 Attribute concept

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

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

### 13.14.4 Attribute classification

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

EXAMPLE 1 The classification for a statistical classification could be:

- Age Brackets: [0-10], [10-20];
- Genders: Male, Female, Other;
- Color: Red, Green, Blue.

## 13.15 Codelists

### 13.15.1 SampleTypeByGeometryType

The codelist **SampleTypeByGeometryType** is a specialization of **AbstractSampleType** created to support the legacy sample types from ISO 19156:2011.

<p><b>Requirement</b> /req/sam-basic/SampleTypeByGeometryType/Sample-TypeByGeometryType-sem</p>	<p>The following entries shall be provided:</p> <ul style="list-style-type: none"> <li>— point: the provided geometry is of type <b>Point</b>.</li> <li>— curve: the provided geometry is of type <b>Curve</b>.</li> <li>— surface: the provided geometry is of type <b>Surface</b>.</li> <li>— solid: the provided geometry is of type <b>Solid</b>.</li> </ul>
---	--

<p><b>Requirement</b> /req/sam-basic/SampleTypeByGeometryType/Sample-TypeByGeometryType-con</p>	<p>The following constraints shall be applied to the value of the result association of the <b>Observation</b> based on the codelist value used:</p> <ul style="list-style-type: none"> <li>— If value “point” is used, the provided geometry shall be of type <b>Point</b>.</li> <li>— If value “curve” is used, the provided geometry shall be of type <b>Curve</b>.</li> <li>— If value “surface” is used, the provided geometry shall be of type <b>Surface</b>.</li> <li>— If value “solid” is used, the provided geometry shall be of type <b>Solid</b>.</li> </ul>
---	---

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

## 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**

<b>Conformance class</b>	<b>/conf/obs-cpt/Observer</b>
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**

<b>Conformance class</b>	<b>/conf/obs-cpt/ObservingProcedure</b>
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**

<b>Conformance class</b>	<b>/conf/obs-cpt/Procedure</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core/AbstractDeployment</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core/AbstractHost</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core/AbstractObservableProperty</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core/AbstractObservation</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core/AbstractObservationCharacteristics</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core/AbstractObserver</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core/AbstractObservingProcedure</b>
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**

<b>Conformance class</b>	<b>/conf/obs-core/NamedValue</b>
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.2.10 Abstract Observation Core – AbstractObservationCollection**

<b>Conformance class</b>	<b>/conf/obs-core/AbstractObservationCollection</b>
Requirements	/req/obs-core/AbstractObservationCollection
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**

<b>Conformance class</b>	<b>/conf/obs-basic</b>
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**

<b>Conformance class</b>	<b>/conf/obs-basic/Deployment</b>
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**

<b>Conformance class</b>	<b>/conf/obs-basic/GenericDomainFeature</b>
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**

<b>Conformance class</b>	<b>/conf/obs-basic/Host</b>
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**

<b>Conformance class</b>	<b>/conf/obs-basic/ObservableProperty</b>
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

<b>Conformance class</b>	<b>/conf/obs-basic/Observation</b>
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

<b>Conformance class</b>	<b>/conf/obs-basic/ObservationCharacteristics</b>
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

<b>Conformance class</b>	<b>/conf/obs-basic/ObservationCollection</b>
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

<b>Conformance class</b>	<b>/conf/obs-basic/Observer</b>
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

<b>Conformance class</b>	<b>/conf/obs-basic/ObservingCapability</b>
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

<b>Conformance class</b>	<b>/conf/obs-basic/ObservingProcedure</b>
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

<b>Conformance class</b>	<b>/conf/sam-cpt</b>
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

<b>Conformance class</b>	<b>/conf/sam-cpt/PreparationProcedure</b>
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

<b>Conformance class</b>	<b>/conf/sam-cpt/PreparationStep</b>
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

<b>Conformance class</b>	<b>/conf/sam-cpt/Sample</b>
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

<b>Conformance class</b>	<b>/conf/sam-cpt/Sampler</b>
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

<b>Conformance class</b>	<b>/conf/sam-cpt/Sampling</b>
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

<b>Conformance class</b>	<b>/conf/sam-cpt/SamplingProcedure</b>
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

<b>Conformance class</b>	<b>/conf/sam-core</b>
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

<b>Conformance class</b>	<b>/conf/sam-core/AbstractPreparationProcedure</b>
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

<b>Conformance class</b>	<b>/conf/sam-core/AbstractPreparationStep</b>
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

<b>Conformance class</b>	<b>/conf/sam-core/AbstractSample</b>
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

<b>Conformance class</b>	<b>/conf/sam-core/AbstractSampler</b>
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**

<b>Conformance class</b>	<b>/conf/sam-core/AbstractSampling</b>
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**

<b>Conformance class</b>	<b>/conf/sam-core/AbstractSamplingProcedure</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/MaterialSample</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/NamedLocation</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/PhysicalDimension</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/Sample</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/SampleCollection</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/Sampler</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/Sampling</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/SpatialSample</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/StatisticalClassification</b>
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**

<b>Conformance class</b>	<b>/conf/sam-basic/StatisticalSample</b>
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

STANDARDSISO.COM : Click to view the full PDF of ISO 19156:2023

## Annex B (informative)

### Common usage of OMS concepts

#### B.1 Introduction

This document defines concepts in support of a generic, cross-domain model for observations, measurements and samples. Concepts are taken from a variety of disciplines. The concepts are used within the model described in this document in a consistent manner, but in order to achieve internal consistency, this varies from how the same concepts 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 OMS concepts to those used within some domains.

For the following domain, information has been provided in the tables listed:

- Earth Observations (EO): [Table B.1](#)
- Metrology: [Table B.2](#)
- Earth science simulations: [Table B.3](#)
- Assay/Chemistry: [Table B.4](#)
- Geology field observations: [Table B.5](#)
- Geotechnics observations: [Table B.6](#)
- Water quality observations: [Table B.7](#)
- Soil quality observations: [Table B.8](#)

#### B.2 Earth observations (EO)

**Table B.1 — Earth observations (EO)**

OMS	EO	Example
Observation::result	Observation value, measurement value, observation	35 µg/m <sup>3</sup>
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	<a href="http://sweetontology.net/realm/PlanetarySurface">http://sweetontology.net/realm/PlanetarySurface</a>
Observation::ultimateFeatureOfInterest	Earth surface	<a href="http://sweetontology.net/realm/PlanetarySurface">http://sweetontology.net/realm/PlanetarySurface</a>
Observation::proximateFeatureOfInterest: SpatialSample	3-D sampling space	Sampling grid

**Table B.1 (continued)**

OMS	EO	Example
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**

OMS	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**

OMS	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**

OMS	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

Table B.4 (continued)

OMS	Geochemistry
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

OMS	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

OMS	Geotechnical in situ test
Observation::result	A log
Observation::observedProperty	A soil property (e.g. 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

OMS	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