*International*
*Virtual*
*Observatory*
*Alliance*

# IVOA Dataset Metadata Model

## Version 1.0

## IVOA Working Draft 20170928

**This version:**
WD-DatasetDM-1.0-20170928
**Previous version(s):**

**Editor(s):**
Mark Cresitello-Dittmar

**Authors:**
Mark Cresitello-Dittmar, Francois Bonnarel, Omar Laurino, Gerard Lemson, Mireille Louys, Arnold Rots, Doug Tody, and the IVOA Data Model Working Group.

# Abstract

This document provides a data model describing the structure and content of generic Dataset metadata for the IVOA. This is a high-level model which is to be referenced and extended by other models describing specific types of Datasets and Data products. In this document, we specify the generic Dataset, as well as an ObservationDataset model which covers the class of Datasets which are derived from an Observation. At the time of this writing, there is no formal Observation-Experiment model for the IVOA, so we include a hypothetical Observation-Experiment model to serve as a placeholder.

# Status of This Document

This is an IVOA Working Draft for review by IVOA members and other interested parties. It is a draft document and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use IVOA Working Drafts as reference materials or to cite them as other than "work in progress".

# Acknowledgements

**Change Log:**

2014 Jun 13: Initial draft of version 1.0
2014 Sep 30: Draft revised with STC2 prototype, initial draft feedback, and updates due to Cube
model development.
2015 Mar 02:
    - Incorporate comments from Spectral2.0 feedback related to Dataset metadata.
    - Format change to better illustrate data type and multiplicity of elements.
    - Update STC2 prototype to current state of development.
    - General review of element descriptions for clarity.
2015 Oct 07:
    - Update Acknowledgments for European contributors
    - Generalize Derived object using ObsConfig pattern
    - Remove Redshift type (used in Derived)
    - Update STC2 prototype to 2015-05-04 model descriptions
     (except for Transform model)
    - Update reference to VO-DML specification (2015-02-06 version)
    - Make section references links.
2015 Oct 18:
    - Update STC2 prototype to 2015-10-14 model diagrams
     o Frame - Transform relation.
2016 Mar 09:
    - Update elements for vo-dml compatibility
     o Curation.reference, DataID.collection, DataID.contributor
     o Enumerations (SpectralBandType)
    - Added simple Party model for Entity types
     o Curation.publisher, Curation.contact, DataID.creator, DataID.contributor
    - Restructured ObsConfig, folding in content concepts, not just content.
    - Normalized Curation.rights from string to AccessRights type.
    - Update STC2 prototype
     o duplicate packaging
     o mods for vo-dml compliance, PixelSpace, PixelCoordinate, Transforms.
2016 Apr 12:
    - restore Curation.rights to attribute of type RightsType, delete AccessRights type.
    - reduce Party types to just Organization and Individual.
    - enhanced language in Publication.refCode description.
    - moved ObsDataset and associated elements to Observation-Experiment package
    - added list of figures
2016 Oct 31:
    - remove STC2 prototype model
2017 Sep 27:
    - removed Characterisation element from ObsDataset; belongs on specific dataset types.
    - removed CoordSys element from ObsDataset; convenience element for content defined
elsewhere.
    - update to current STC2 Measurement model interface

# Contents

5

7

# List of Figures

# 1  Introduction

## 1.1  Motivation

All IVOA datasets must contain a common set of metadata elements to facilitate the registration, discovery, and interoperability of these datasets.  To date, individual IVOA data models have independently defined this metadata within the separate documents.  This has resulted in some level of inconsistency between models, as well as document bloat, and some ambiguity as to the hierarchy and relation of models to each other.  For example, the ObsCore-1.0 model describes itself as defining "the core components of the Observation data model ", but there is no formal definition of an Observation data model in the IVOA.  Without this higher-level document, it is difficult for detailed models to properly reference and/or extend this content consistently.

With the development of the Cube model, significant effort has been made to properly model this high-level metadata, and separate the components related to the generic dataset, a dataset derived from an observation, and the observation itself.  This document represents the results of that effort.  Here, we define the generic dataset metadata, and provide an example for extending this with metadata related to datasets resulting from a specific process (Observation).  As such, the ObsCore model should be considered a 'view' of this model, highlighting the core components required for supporting TAP services.

The descriptions of many elements of this model are a result of reviewing and combining those contained in the ObsCore (1.0), Spectral (2.0), and Characterisation (1.13) models.  As such, it represents a uniform, consistent description set.  Future revisions of those documents should be defined with respect to this model.

## 1.2  Requirements

The primary goals of this document are:

 • to provide a specification of generic dataset metadata sufficient to support current IVOA data product models and access protocols.

 • to specify metadata associated with an Observation (experiment) which are to be included in datasets derived from observations (ObsDataset).

## 1.3  Role in the IVOA Architecture



**USERS**

**Dataset**

**COMPUTERS**

**USER LAYER**

| Browser Based Apps | Desktop Apps | Script Based Apps |

**USING**

R E G I S T R Y

STC

**VO Query Languages**

Dataset

UCD  VOUnits  CharDM

**Semantics**  VO CORE  **Data Models**

**Formats**

D A T A   A C C E S S

P R O T O C O L S

**SHARING**

**Data and Metadata Collection**

**Storage**  **RESOURCE LAYER**  **Computation**

20120521
IVOA Architecture
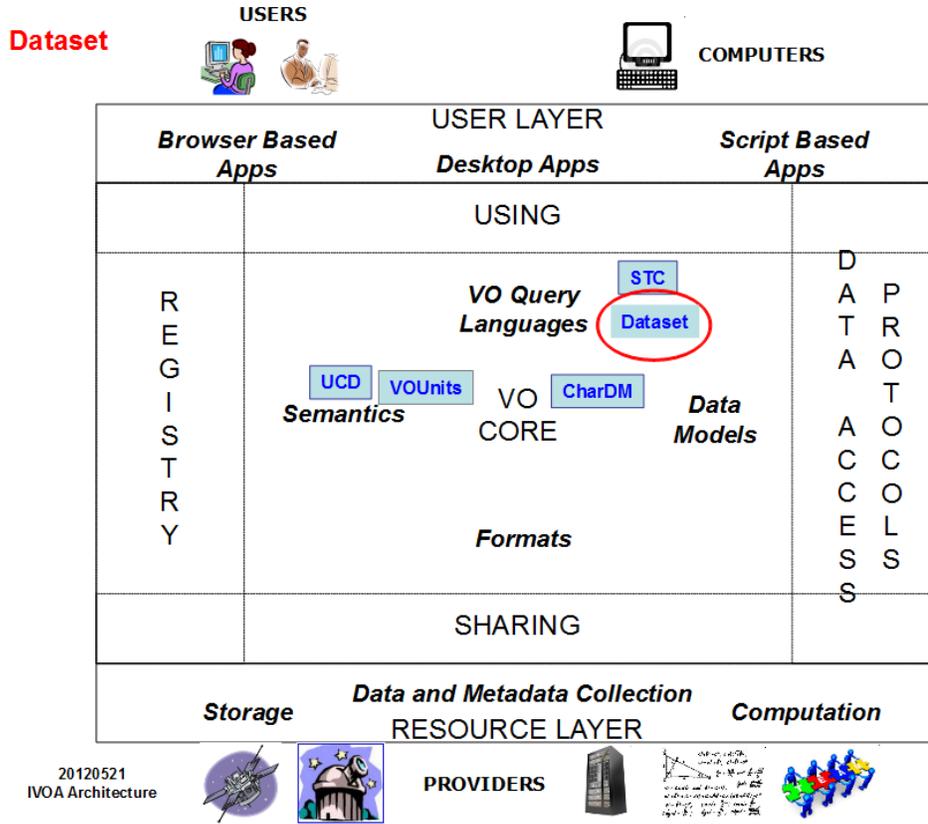
**PROVIDERS**

## 1.4  Model Dependencies
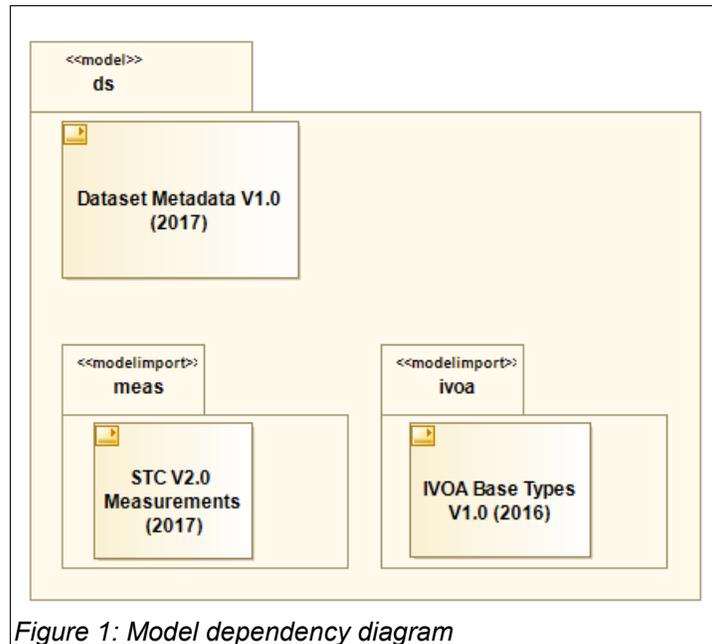


*Figure 1: Model dependency diagram*

The Dataset model is built on other data models as indicated in Figure 1.  The <<model>> and <<modelimport>> stereotypes provide information identifying the model, it's version, any dependencies, and URLs to find more information about the model definitions including HTML and schema documentation.  See Appendix B for more information about the content of these stereotypes and how they are used in serializations.

## 1.5  Structure of this Documentation

+ Major sections for each model area (Dataset, Observation, etc. ).

+ First subsection in each section is the primary element within that model

+ Subsequent subsections for secondary elements, generally in alphabetical order,

   but occasionally a logical grouping of related objects makes more sense.

+ Each subsection has sub-subsections for each attribute/relation

     - attributes show the full definition including datatype and usage.

     - relations describe the usage of the object in that context,

      the type of the target of the relation, and a reference to the full

      definition of that type.

+ Capitalization convention

     - Objects and complex data types are expressed in PascalCase

     - Attributes are camelCase

     - Primitive data types (string, double, etc.) are lower case

# 2 Dataset Model

This section describes the generic, high-level metadata associated with an IVOA Dataset. Since serialization format choices may effect the number of files or components which comprise a dataset, we define an IVOA Dataset as "a file or files which are considered to be a single deliverable". Examples of viable datasets include:

  + An individual data product, such as a Spectrum, or Image.

  + A 'tar' file or directory of processed observational data files.

This metadata identifies the dataset, and provides information regarding the ownership, rights and associations with other datasets. The primary purpose of this metadata is to facilitate the registry and discovery of datasets within the IVOA community.

Several of the objects modeled here are based on descriptions given in the IVOA document, "Resource Metadata for the Virtual Observatory; Version 1.12" [1] (Resource Metadata). Where applicable, we provide the appropriate citation in the text below.
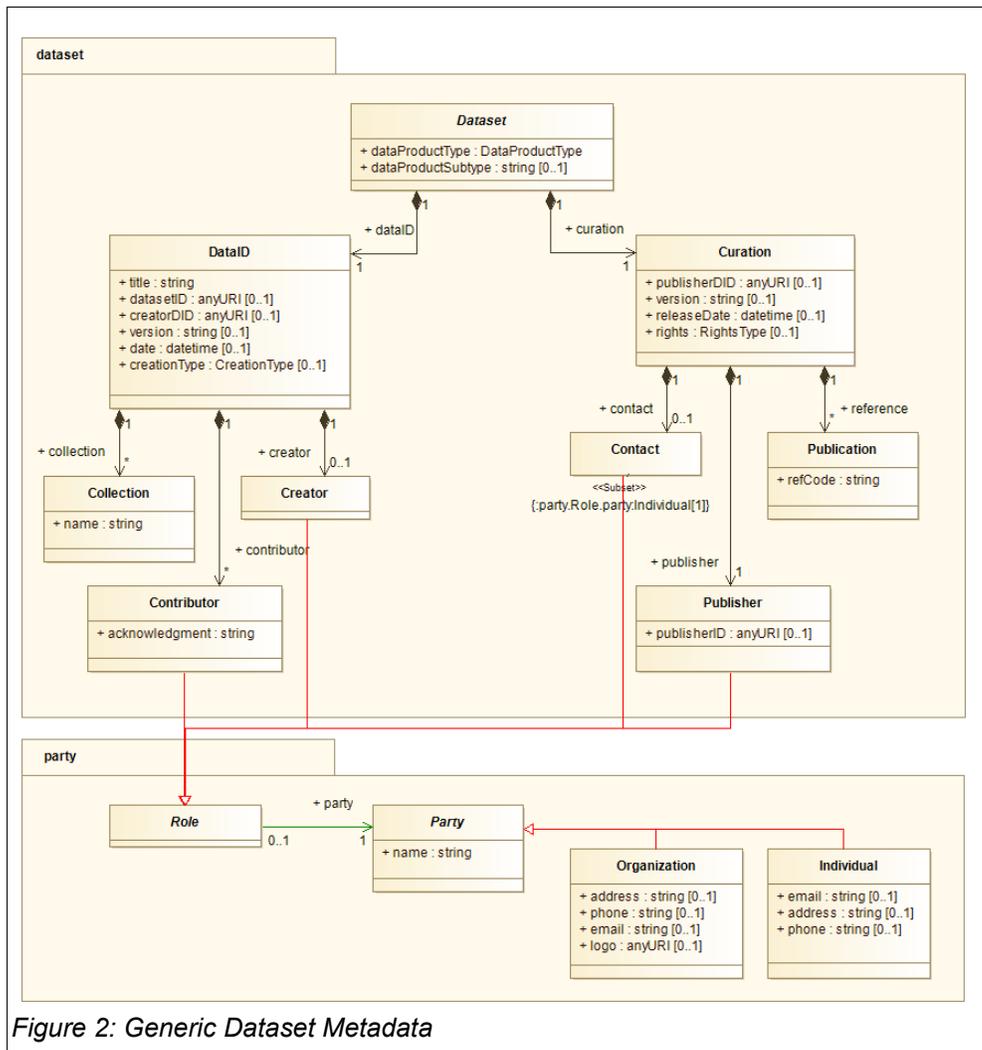


*Figure 2: Generic Dataset Metadata*

## 2.1 Dataset



Figure 3: Dataset Metadata detail
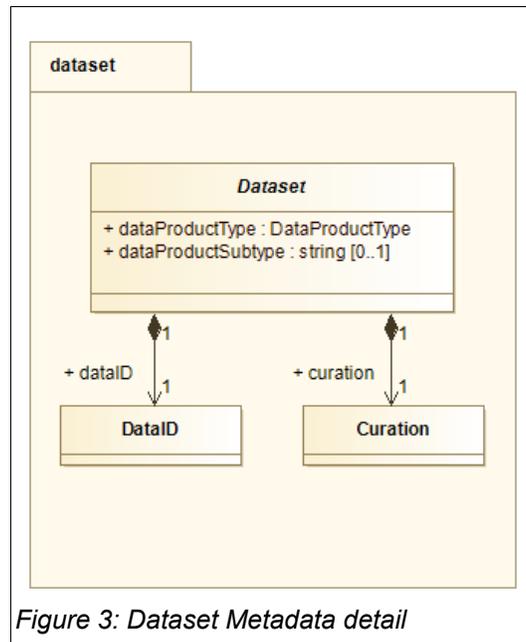
Abstract object for the generic IVOA Dataset.  It is intended to be useful for any type of data.  Specific dataset models should extend this object, providing detailed definitions and additional content as appropriate for that type of dataset.

### 2.1.1 Dataset.dataProductType

**type: DataProductType**　　　　　　　**type-detail: Section 5.2.2**
**multiplicity: 1**

Describes the high level scientific classification of the data content.  Values are restricted to the DataProductType enumeration set and convey the general idea of the content and organization of a dataset.

### 2.1.2 Dataset.dataProductSubtype

**type: string**　　　　　　　**type-detail: Section 5.1**
**multiplicity: 0..1**

Secondary type classification for the dataset.  This field is intended to precisely specify the scientific nature of the data product, possibly in terms relevant only to a specific archive or data collection.  For example, dataProductType='image' could have associated dataProductSubtype="src.image", "bkg.image", "PixelMask", etc.  Values are unrestricted strings.

13

### 2.1.3 Dataset.curation

**type: Curation**      **type-detail: Section 2.6**
**multiplicity: 1**

Provides metadata related to the entity responsible for the curation of the dataset.

### 2.1.4 Dataset.dataID

**type: DataID**      **type-detail: Section 2.7**
**multiplicity: 1**

DataID provides high level identification metadata for the dataset itself, and any associations with various collections.

## 2.2  Collection



dataset

Collection

+ name : string
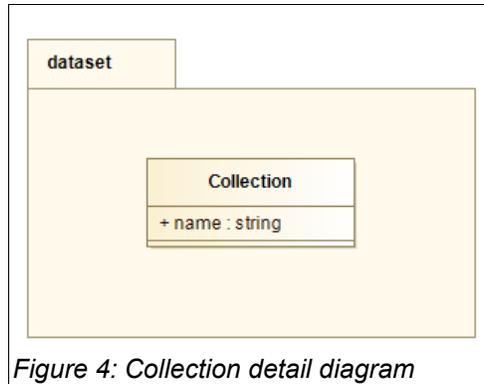
*Figure 4: Collection detail diagram*

A generic organizational construct which allows Datasets to be associated with each other by a set of Collection properties.  Datasets tagged with the same Collection properies can be assumed to have some degree of compatibility.

### 2.2.1  Collection.name

**type: string**                                    **type-detail: Section 5.1**
**multiplicity: 1**

The values are generally defined by the creating entity.  Examples: "WFC", "Sloan", "BFS Spectrograph", "MSX Galactic Plane Survey".

## 2.3  Contact



*Figure 5: Contact detail diagram*
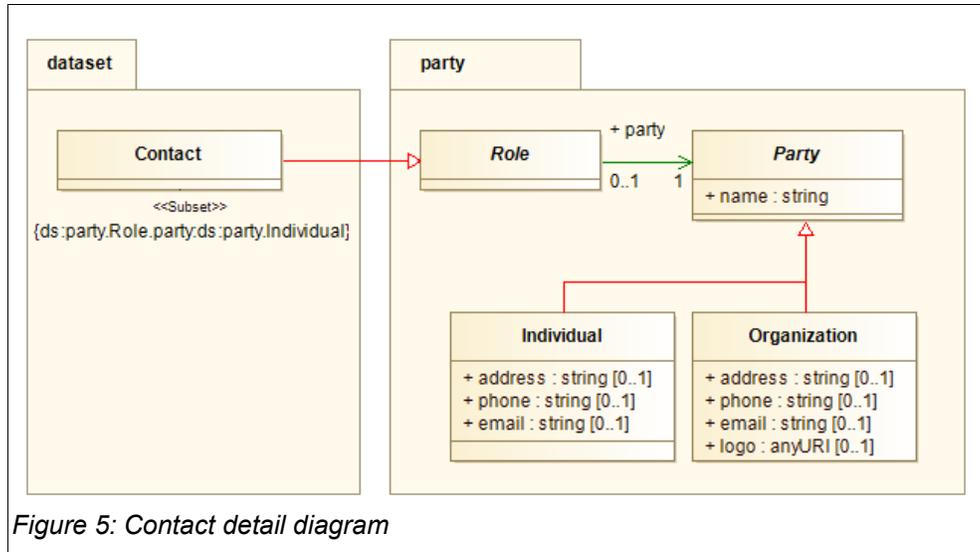
Contact information for a person or entity.

Contact is modeled as a Role played by a particular person or entity (Party).  We subset the type of Party to include only Individuals.  This includes both a physical person, or proxy service such as a helpdesk.

Party package is described in Section **4**

## 2.4  Contributor
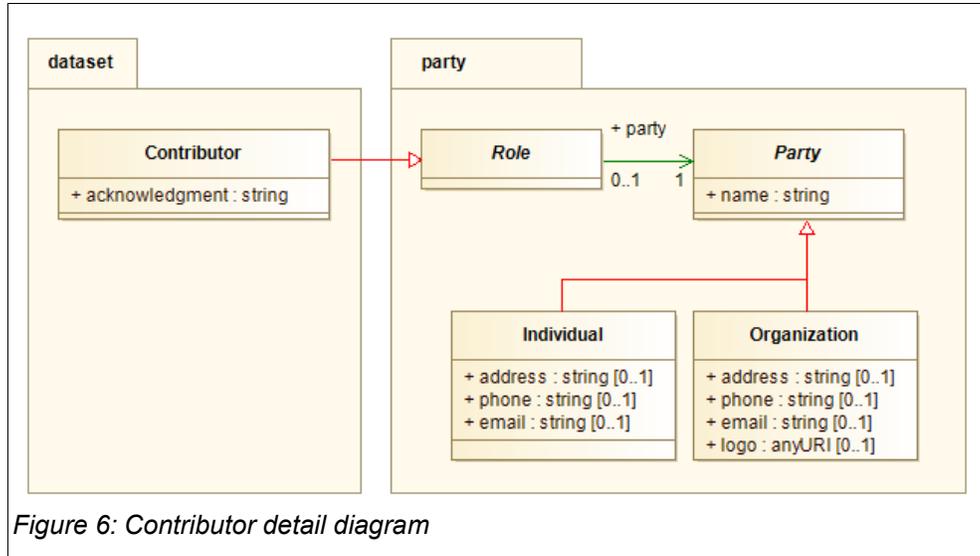


Figure 6: Contributor detail diagram

Contributor is modeled as a Role played by a Party or entity who participated in the generation of the Dataset.

Party package is described in Section **4**

### 2.4.1  Contributor.acknowledgment

**type: string**                    **type-detail: Section 5.1**
**multiplicity: 1**

Acknowledgment expression for the contributor.  Users of the dataset should include these in subsequent credits and acknowledgements.  The expression should be formatted as desired by the contributor.

17
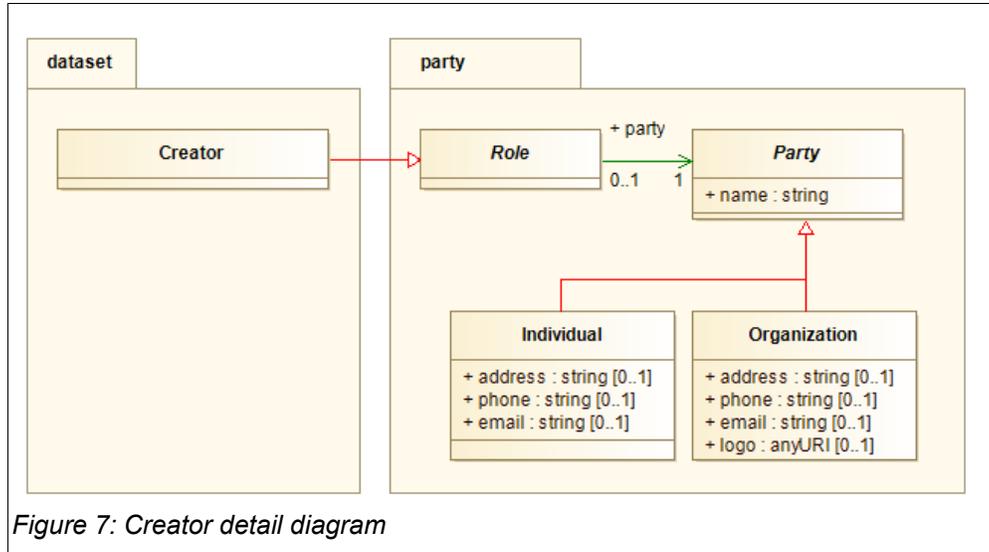
## 2.5  Creator



*Figure 7: Creator detail diagram*

Creator is modeled as a Role played by the organization or entity which created the Dataset.

Party package is described in Section **4**

18
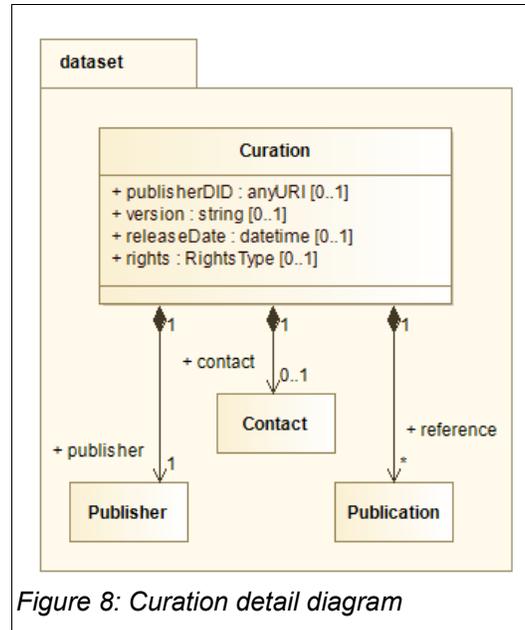
## 2.6  Curation



*Figure 8: Curation detail diagram*

The Curation object provides metadata assigned by the entity responsible for the support of the dataset content as well as identifying metadata about that entity.  It is assembled from definitions provided by the IVOA Resource Metadata document.  Here, we provide a brief description of each field for easy reference, along with a notation of its mapping to the Resource Metadata document (RM:field), where the reader may find more detailed information.

### 2.6.1  Curation.contact

**type: Contact**                                    **type-detail: Section 2.3**
**multiplicity: 0..1**

Contact information of the person/entity responsible for the content of the dataset.  We recommend using a generic 'helpdesk' type contact rather than individuals whose information may more easily become obsolete. (RM:Curation.Contact)

### 2.6.2  Curation.publisher

**type: Publisher**                                  **type-detail: Section 2.9**
**multiplicity: 1**

The entity making the data available.  (RM:Curation.Publisher)

19

### 2.6.3  Curation.publisherDID

**type: anyURI**          **type-detail: Section 5.1**
**multiplicity: 0..1**

IVOA dataset identifier assigned by the publisher to uniquely identify the dataset within its holdings.  Typically, the basis of this identifier will be the publisher ID.  However, if the publisher chooses to use a 'global index service' such as ADS to obtain persistent identifiers for their datasets, rather than generate their own, that identifier should be used both here and for DataID.datasetID.  Note: this model also defines a creator dataset ID (DataID.creatorDID), these will differ if the publishing entity is not the creator of the dataset.  Values are to be expressed as dataset identifiers using the syntax described in "IVOA Identifiers"[2].

### 2.6.4  Curation.reference

**type: Publication**          **type-detail: Section 2.8**
**multiplicity: 0..\***

Zero or more bibliographic or documentation references associated with the dataset.  Each instance provides a single forward link to a major publication which references the dataset. (RM:General.Source)

### 2.6.5  Curation.releaseDate

**type: datetime**          **type-detail: Section 5.1.3**
**multiplicity: 0..1**

Date the curated dataset was last modified.  (RM:Curation.Date)

### 2.6.6  Curation.rights

**type: RightsType**          **type-detail: Section 5.2.3**
**multiplicity: 0..1**

Indicates the level of access being granted.  Values are restricted to the RightsType enumeration set. (RM:Collection.Rights)

### 2.6.7  Curation.version

**type: string**          **type-detail: Section 5.1**
**multiplicity: 0..1**

Version of the curated dataset, assigned by the publisher.  This is an independent versioning from DataID.version that allows the publisher to track changes to the high level dataset metadata (e.g. curation metadata, identifiers, etc.) without effecting the creator defined dataset version.  The value may be based on the DataID.version (e.g. by adding a sub-version extension), or an independent versioning.  There are no format restrictions on the value.  (RM:Curation.Version)

20
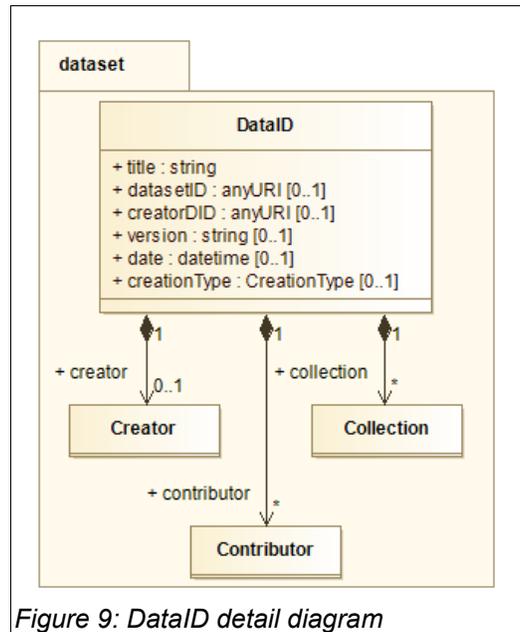
## 2.7  DataID



*Figure 9: DataID detail diagram*

The Data Identification object (DataID) stores the dataset identifiers and other metadata typically assigned by the dataset creator.

The Dataset IDs in this object must comply with the syntax for dataset identifiers defined in the "IVOA Identifiers" [2] document, including the use of 'stop' characters to identify specific datasets that are not individually in the registry.  e.g., ivo://example.net/aservice?2013/5/2342.

Much of the content of this object is assembled from various definitions in the IVOA Resource Metadata document.  Here, we provide a brief description of each field for easy reference, along with a notation of its mapping to the Resource Metadata document (RM:field), where the reader may find more detailed information.

### 2.7.1  DataID.collection
**type: Collection**                    **type-detail: Section 2.2**
**multiplicity: 0..\***

The dataset is associated with zero or more Collections (instrument name, survey name, etc.) . Each instance identifies a tag indicating some degree of compatibility with other data sharing the same Collection properties.

### 2.7.2  DataID.contributor
**type: Contributor**                    **type-detail: Section 2.4**
**multiplicity: 0..\***

Persons or entities who contributed to the generation of the scientific content of the dataset. Users of the dataset should include these in subsequent credits and acknowledgements. (RM:Curation.Contributor)

21

### 2.7.3 DataID.creationType

**type: CreationType**          **type-detail: Section 5.2.1**
**multiplicity: 0..1**

The dataset creation type describes the nature or genre of the content. Values are restricted to the CreationType enumeration set. (RM:General.Type)

Note: This field provides information about the process by which the dataset was created. As the Observation/Experiment model matures, this may evolve into a provenance element on the Experiment type.

### 2.7.4 DataID.creator

**type: Creator**          **type-detail: Section  2.5**
**multiplicity: 0..1**

The institution or entity which created the dataset. (RM:Curation.Creator)

### 2.7.5 DataID.creatorDID

**type: anyURI**          **type-detail: Section 5.1**
**multiplicity: 0..1**

The dataset identifier assigned by the creator. Here, the authority-id of the identifier must be that of the creator. It is used to identify the original exposure of the dataset in an archive, and will remain static regardless of where the dataset is published. The creator ID will not necessarily change even if the VO object in question is a cutout or is otherwise further processed.

### 2.7.6 DataID.datasetID

**type: anyURI**          **type-detail: Section 5.1**
**multiplicity: 0..1**

If the dataset is registered with an external 'global index service' such as ADS, the publisher may include that identifier here. This provides a common, persistent identifier for the dataset, and possible access point to follow for information on publications and other related datasets. Note: the same dataset published at more than one location would have different Curation.publisherDID values, but the same DataID.datasetID. eg: "ivo://ADS/Sa.CXO?obsid=1234", "ivo://ADS/sh.hut#ngc4151_141"

### 2.7.7 DataID.date

**type: datetime**          **type-detail: Section 5.1.3**
**multiplicity: 0..1**

Data processing or creation date (RM:Curation.Date).

### 2.7.8 DataID.title

**type: string**          **type-detail: Section 5.1**
**multiplicity: 1**

A free form string giving a title for the dataset. (RM:Identity.Title)

## 2.7.9  DataID.version

**type: string**                                    **type-detail: Section 5.1**
**multiplicity: 0..1**

Version assigned by the creator, reflecting the production version of the dataset.  This value should only be changed by the creator, upon the new release of a dataset.  There are no format restrictions or specifications on the versioning scheme.

## 2.8 Publication
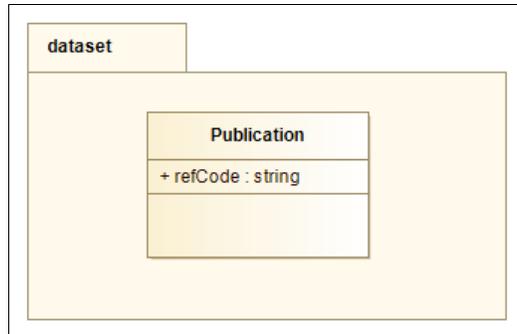


Any referenceable publication associated with a Dataset.

### 2.8.1 Publication.refCode

**type: string**                    **type-detail: Section 5.1**
**multiplicity: 1**

Reference code of the publication.  Values should be expressed as a URI formatted in accordance to an accepted schema.  For example: a 'doi' should use the form described by the doi schema (http://doi.org);  bibcode according to the bibcode pattern, namely a 19 character string beginning with 4 digits.  Free text references are allowed, but discouraged.  If used, they must not start with the pattern "[a-zA-Z]+:" to ensure they are not interpreted as URIs.

## 2.9  Publisher
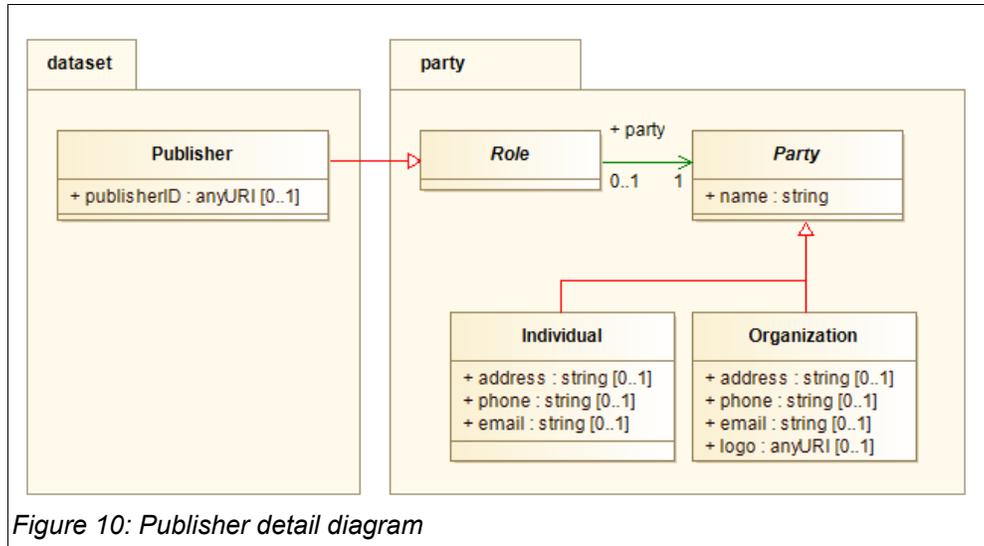


*Figure 10: Publisher detail diagram*

Publisher is modeled as a Role played by the organization or entity making the Dataset available.

Party package is described in Section 4

### 2.9.1  Publisher.publisherID

**type: anyURI**             **type-detail: Section 5.1**
**multiplicity: 0..1**
IVOA resource identifier associated with the publisher and registered with an IVOA compliant registry  (eg: ivo://mast.stsci.edu).  Values are to be expressed using the syntax described in "IVOA Identifiers"[2].
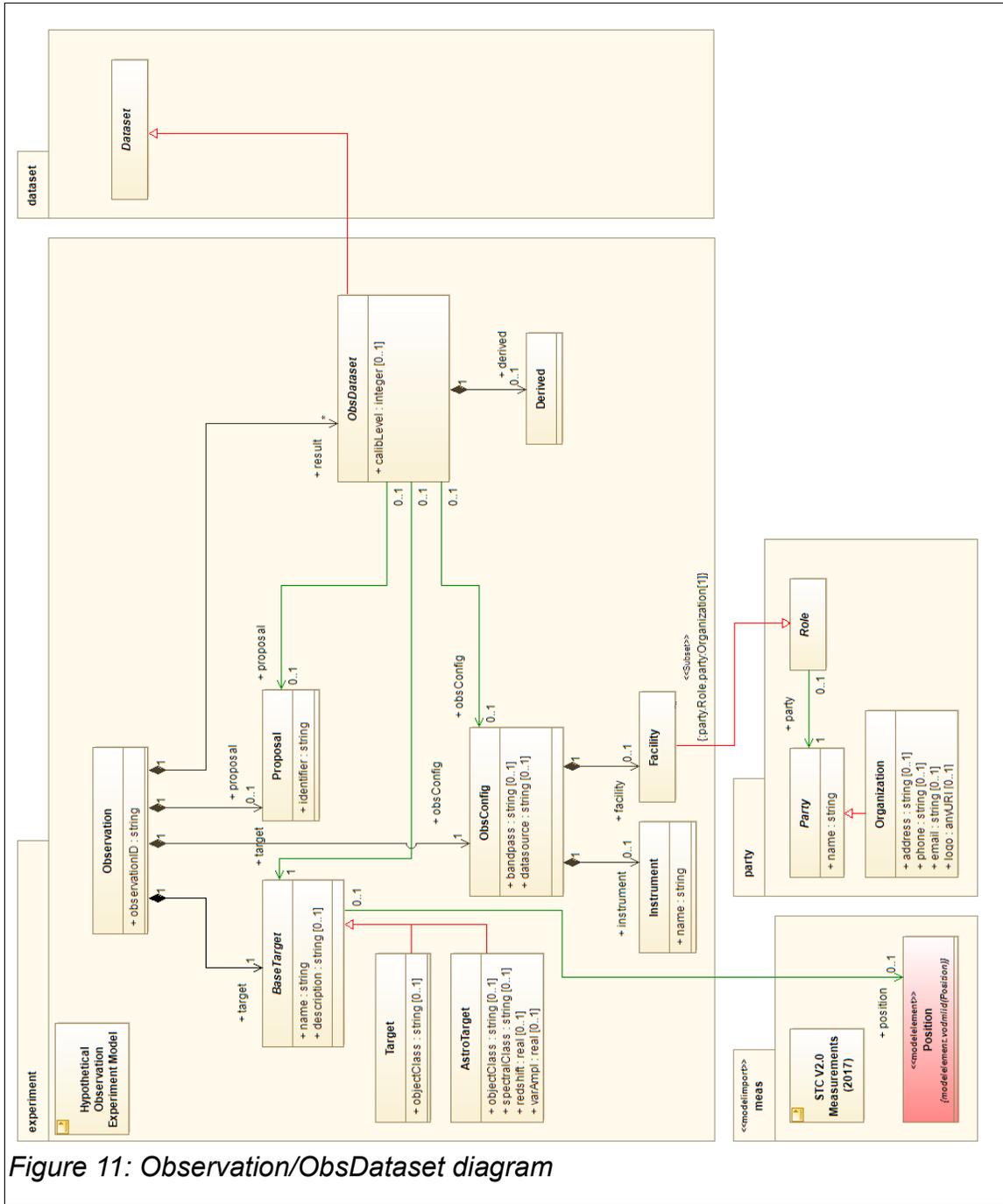
*Figure 11: Observation/ObsDataset diagram*

26

# 3 Observation-Experiment

An Observation Dataset (ObsDataset) is modeled as the result of an Observation, and refers to several elements related to the Observation and its configuration. As of the time of this writing, there is no IVOA recommendation for a general Observation data model. The Provenance data model, in progress, will define the pattern for describing the relation between actions and results, and how to record these in datasets. In lieu of these standards, this document defines a straw-man Observation model.

## 3.1 Observation

Head class for an Observation Experiment.

The Observation is modeled as a type of 'Experiment', with some basic structure defined to provide metadata about the observation target and configuration. The product, or 'result' of the Observation is zero or more ObsDataset objects. This pattern is inspired by, and compatible with the Simulation Data Model [6], where a 'Simulation' can be considered another form of 'Experiment' or perhaps even another form of 'Observation'.

### 3.1.1 Observation.observationID

**type: string**                    **type-detail: Section 5.1**
**multiplicity: 1**

Internal ID determined by the data provider to uniquely identify the observation within the institution or entity performing the observation.

### 3.1.2 Observation.target

**type: BaseTarget**                 **type-detail: Section 3.4**
**multiplicity: 1**

The target of the observation. The content of this object may vary greatly depending on the goals and nature of the observation. For example, the 'target' could be a galaxy, stellar object, planet, or calibration source. As such, we allow the BaseTarget class here, and permit users to define and use more content rich flavors according to their needs.

### 3.1.3 Observation.obsConfig

**type: ObsConfig**                  **type-detail: Section 3.11**
**multiplicity: 1**

Observation configuration metadata, provides information about who, where, and how the observation was conducted.

### 3.1.4 Observation.proposal

**type: Proposal**                   **type-detail: Section 3.12**
**multiplicity: 0..1**

Identifies any proposal related to the observation. This field may be used to gather all observations and products related to a particular proposal.

### 3.1.5 Observation.result

**type: ObsDataset**          **type-detail: Section 3.2**
**multiplicity: 0..\***

The result of an observation is zero or more Observation Datasets

## 3.2 ObsDataset

ObsDataset is an abstract extension of Dataset defining additional metadata relevant to Datasets which are derived from Observations.  This metadata gives a high-level summary of the coverage of the dataset in coordinate space, as well as the coordinate systems used, and general information about the observation itself.

### 3.2.1 ObsDataset.calibLevel

**type: integer**          **type-detail: Section 5.1**
**multiplicity: 0..1**

High level classification for the calibration level of a particular dataset as a whole.  The calibration level concept conveys to the user information on how much data reduction/processing has been applied to the data.  It is up to the data providers to consider how to map their own internal classification to the scale defined here.

Scale:

- 0 - Raw instrumental data, in a proprietary or internal data-provider defined format.

- 1 - Instrumental data in a standard format (FITS, VOTable, etc )

- 2 - Calibrated, science ready data with the instrument signature removed.

- 3 - Enhanced data products like mosaics, resampled or drizzled images, or heavily processed survey fields.  Level 3 data products may represent the combination of data from multiple primary observations.

### 3.2.2 ObsDataset.derived

**type: Derived**          **type-detail: Section 3.5**
**multiplicity: 0..1**

Provides a high level summary of certain properties of the dataset.  Its primary purpose is to support high level filtering of datasets during data discovery.

### 3.2.3 ObsDataset.obsConfig

**type: ObsConfig**          **type-detail: Section 3.10**
**multiplicity: 0..1**

Reference to ObsConfig object from Observation.  This object provides some high-level metadata related to the observation configuration.

28

### 3.2.4  ObsDataset.observationID

**type: id**                                **type-detail: n/a**
**multiplicity: n/a**

Implicit element of the ObsDataset object due to the composition relation to Observation.  This is an internal ID determined by the data provider to identify the Observation from which the dataset was produced.

### 3.2.5  ObsDataset.proposal

**type: Proposal**                         **type-detail: Section 3.11**
**multiplicity: 0..1**

Reference to Proposal object from Observation.  This object provides metadata identifying any proposal related to the observation which produced the dataset.

### 3.2.6  ObsDataset.target

**type: BaseTarget**                       **type-detail: Section 3.4**
**multiplicity: 1**

Reference to a BaseTarget object from Observation.  Provides metadata describing the target of the observation.
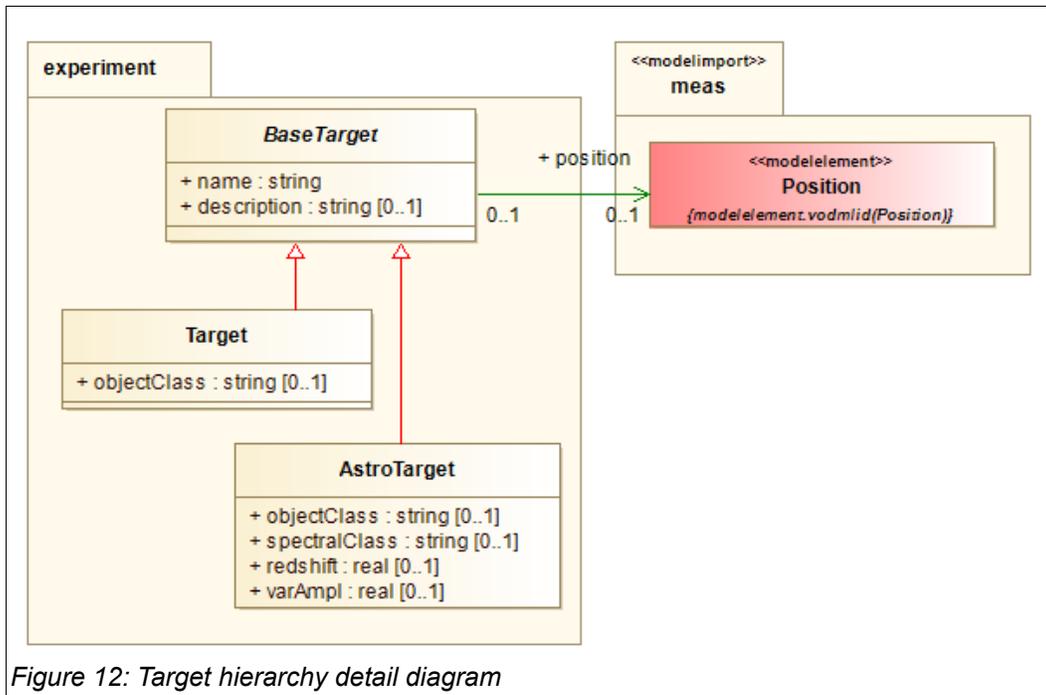
*Figure 12: Target hierarchy detail diagram*

## 3.3  AstroTarget

Extension of BaseTarget specialized for astronomical objects.  The AstroTarget defines additional astronomical properties of the target.

### 3.3.1  AstroTarget.name

**type: string**                                **type-detail: Section 5.1**
**multiplicity: 1**

When referring to an astronomical target, one may specify a particular object, or a more general target such as the name of a survey field.  When specifying a particular object, it is highly recommended to use a name suitable for input to a name resolver.

### 3.3.2  AstroTarget.position

**type: Position**                                **type-detail: Section 6.5.2**
**multiplicity: 0..1**

In the context of the astronomical target, this field gives the nominal RA and Dec location for the target.  For example, the catalog position of the source.

### 3.3.3  AstroTarget.objectClass

**type: string**                                **type-detail: Section 5.1**
**multiplicity: 0..1**

General classification or type of the target.  This field supports the discovery of data pertaining to a common class of object, e.g. "Star", "Galaxy", "AGN".  At the time of this writing, there is no

IVOA recommended vocabulary for this field. The SIMBAD and NED databases use defined vocabularies for astronomical object classifications which may serve as the basis for such.

### 3.3.4 AstroTarget.spectralClass

**type: string**     **type-detail: Section 5.1**
**multiplicity: 0..1**

Spectral class of the object. As with objectClass, there is no IVOA recommended vocabulary for specifying the spectral class of an object. There is an IVOA Note on the subject entitled "An encoding system to represent stellar spectral classes in archival databases and catalogs"[7], describing an encoding system which has been adopted by the MAST archive.

### 3.3.5 AstroTarget.redshift

**type: real**     **type-detail: Section 5.1**
**multiplicity: 0..1**

This field gives the canonical redshift of the astronomical object. It is normally used to store the cosmological redshift of extragalactic objects, although it may also be used to store the observed redshift of Galactic sources if that information is felt by the data provider to be useful.

### 3.3.6 AstroTarget.varAmpl

**type: real**     **type-detail: Section 5.1**
**multiplicity: 0..1**

Canonical variability amplitude attributed to the target.

## 3.4  BaseTarget

Abstract base class for the Target object tree.  The target object provides identifying metadata related to the subject or goal of the experiment.  For an Observational experiment, this would typically be an astronomical object.  The BaseTarget class defines high-level identifying information, and must be extended for particular classes of Target which may define additional characteristics.

### 3.4.1  BaseTarget.name

**type: string**                              **type-detail: Section 5.1**
**multiplicity: 1**

The target name.  The primary purpose of this field is to provide the user with a recognizable identity of the particular subject or goal.  However, since this may be a query-able field in data discovery protocols, care should be taken to use values which follow conventions for the domain appropriate for the data.  For an astronomical object, this may be a name suitable for use within a domain-specific resolution service.  Simulated data might also use this sort of name (if simulating a particular object), or a more generic term such as "G2V star".

### 3.4.2  BaseTarget.description

**type: string**                              **type-detail: Section 5.1**
**multiplicity: 0..1**

Free form description of target.

### 3.4.3  BaseTarget.position

**type: Position**                            **type-detail: Section 6.1.1**
**multiplicity: 0..1**

This field provides the spatial location of the target.  The value is a STC Position object which supports all required dimensionality and coordinate frame specification needs.

## 3.5  Derived
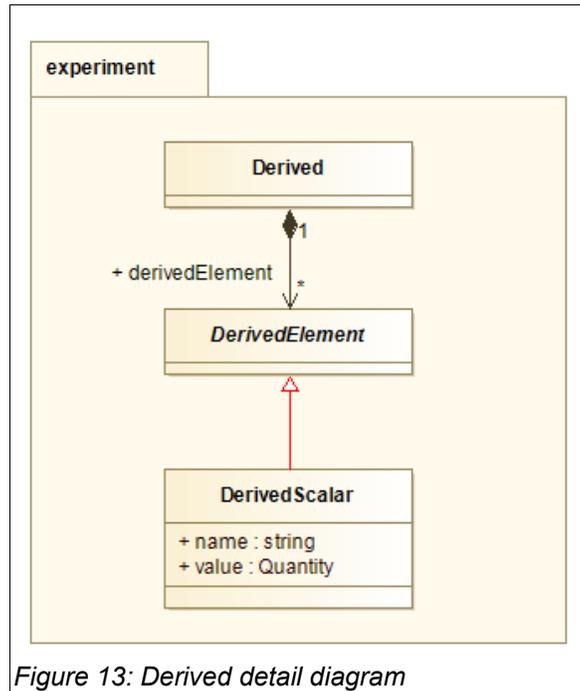


Figure 13: Derived detail diagram

The Derived (short for Derived Data) object holds derived information obtained by evaluating or analyzing the contents of the dataset.  The specific content of this object is strongly dependent on the specific type of dataset, so we provide a generic model which may be specialized in other models to define elements appropriate for that type of dataset.

The primary purpose of this object is to provide a common framework in which specific information may be placed to aid in discovery and filtering of datasets in various access protocols.

### 3.5.1  Derived.derivedElement

**type: DerivedElement**                **type-detail: Section 3.6**
**multiplicity: 0..*** 

Collection of zero or more DerivedElement objects, each of which provides a specific quantity obtained by analyzing the dataset content.

## 3.6  DerivedElement

Abstract base for defining derived data elements.  Typically, models for specific data products would extend this object to define various elements appropriate for that model.  For example, the Spectrum model could define signal-to-noise ratio (SNR), or TimeSeries could define period, or variability.  We put no restriction on the DerivedElement content since the result could be a simple value or a complex object.  However, it is recommended that extensions be simple and compact in keeping with the primary intent of use in data discovery.

33

## 3.7  DerivedScalar

Simple extension of DerivedElement class which can serve many use cases.  Usages of this object in other models to define specific elements should explicitly define the element name, and the process by which the value is determined.

### 3.7.1  DerivedScalar.name

**type: string**         **type-detail: Section 5.1**
**multiplicity: 1**

Name identifying the derived element.

### 3.7.2  DerivedScalar.value

**type: Quantity**         **type-detail: Section 5.1**
**multiplicity: 1**

Value of the derived element.
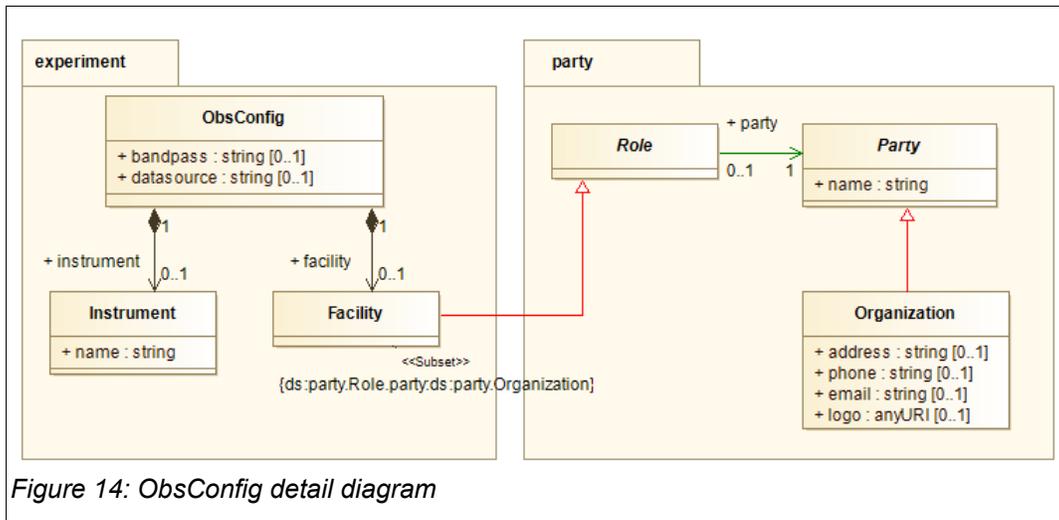
*Figure 14: ObsConfig detail diagram*

## 3.8  Facility

The Facility performing the observation.  This is modeled as the Role played by a particular Organization (Party entity).

## 3.9  Instrument

The instrument used to create the data. This can be a specific instrument, general type or something else, such as a program in the case of theoretical data. (RM:Collection.Instrument)

### 3.9.1  Instrument.name

**type: string**                                            **type-detail: Section 5.1**
**multiplicity: 1**

Name of the instrument.

## 3.10 ObsConfig

ObsConfig describes all Observation Configuration metadata.  We define a small set of configuration elements which are required as Provenance in the observation dataset.

### 3.10.1          ObsConfig.bandpass

**type: string**                                            **type-detail: Section 5.1**
**multiplicity: 0..1**

Describes the Spectral domain of the Observation in very general sense.

The value may be expressed in terms of general spectral bands, or specific bandpass names.  If multiple bands are covered, the value may be a comma delimited combination of appropriate bands.  If expressed as general bands, the value(s) must be selected from the enumerated set given by the **SpectralBand type** (Section **5.2.4)** .  There is no controlled vocabulary for specific

bandpass names as the list is too long to enumerate.  Effort should be made to use highly recognized bandpass names (eg: "U","V","B","R","I", "H-alpha").

This field corresponds to both the Coverage.Spectral and Coverage.Spectral.Bandpass fields of the Resource Metadata document.

### 3.10.2 ObsConfig.dataSource

**type: string**               **type-detail: Section 5.1**
**multiplicity: 0..1**

Describes the original source of the data in a very general fashion.  In other words, "What sort of observation originally generated the data?"  Suggested values include:

- **survey**: Survey data typically covers some region of observational parameter space with as complete as possible coverage within that region.

- **pointed**:  Pointed data of a particular object or field.

- **theory**:  Theory data, generated based on a theoretical model.

- **artificial**:  Artificial, or simulated data.  Similar to 'theory', but not necessarily based on a theoretical model.

- **custom**:  Custom data, as part of a specific research project.

### 3.10.3 ObsConfig.facility

**type: Facility**               **type-detail: Section 3.8**
**multiplicity: 0..1**

Metadata pertaining to the facility performing the observation.

### 3.10.4 ObsConfig.instrument

**type: Instrument**               **type-detail: Section 3.9**
**multiplicity: 0..1**

Metadata pertaining to the instrument used to create the data.

## 3.11 Proposal
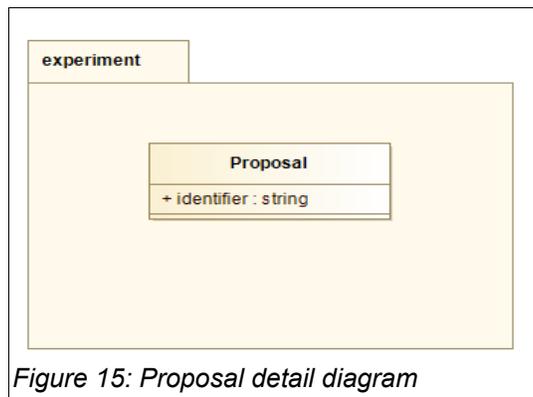


*Figure 15: Proposal detail diagram*

Metadata related to the proposal or document which spawned the observation.

### 3.11.1 Proposal.identifier

**type: string**       **type-detail: Section 5.1**
**multiplicity: 0..1**

Tag used to uniquely identify a particular proposal within the institution or entity.

## 3.12 Target

Extension of BaseTarget, this is a general purpose Target object.

### 3.12.1    Target.objectClass

**type: string**                    **type-detail: Section 5.1**
**multiplicity: 0..1**

General classification or type of the target.  This field supports the discovery of data pertaining to a common class of object, e.g. "Star", "Galaxy", "AGN".  At the time of this writing, there is no IVOA recommended vocabulary for this field.  The SIMBAD and NED databases use defined vocabularies for astronomical object classifications which may serve as the basis for such.

# 4   Party Package
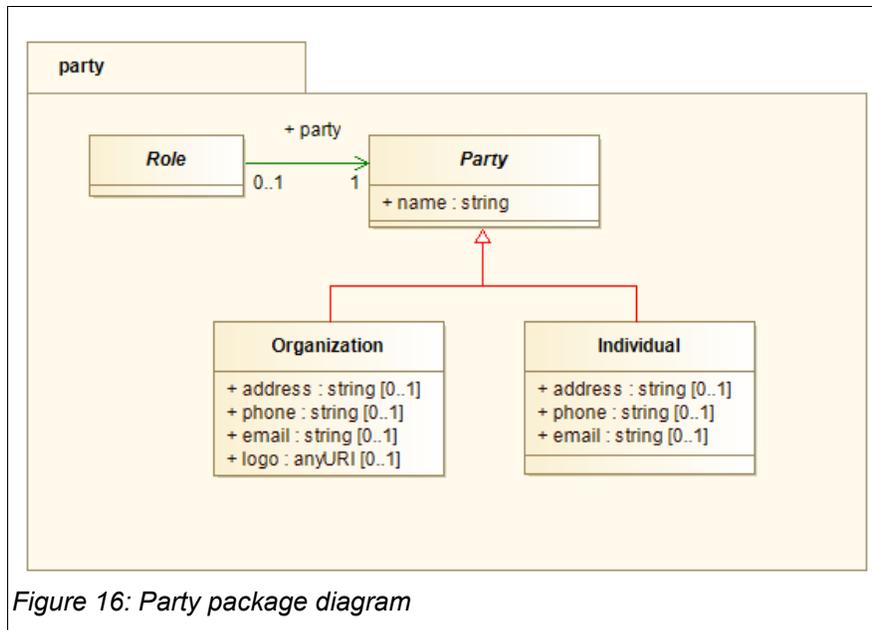


*Figure 16: Party package diagram*

We include a simple Party model for associating an Entity with a Role that Entity is playing.  For example, a particular Individual can be both a Contact and Publisher of a dataset.

## 4.1   Role

Abstract class for the entity role.  Models should extend this class to define local roles which are played by various entities/parties.

### 4.1.1   Role.party

**type: Party**                                  **type-detail: Section 4.2**
**multiplicity: 1**

Reference to the Party or Entity which is associated with this role.

## 4.2   Party

Abstract head of the set of classes describing various entities.

### 4.2.1   Party.name

**type: string**                                  **type-detail: Section 5.1**
**multiplicity: 1**

Name of the Party or entity.  All entities are assumed to have a name.

## 4.3  Individual

Party which describes an individual or representative.

### 4.3.1  Individual.address

**type: string**                              **type-detail: Section 5.1**
**multiplicity: 0..1**

Mailing address for the Individual.  The value is expressed as a single string containing all components of the address.

### 4.3.2  Individual.email

**type: string**                              **type-detail: Section 5.1**
**multiplicity: 0..1**

E-mail address of the Individual.

### 4.3.3  Individual.phone

**type: string**                              **type-detail: Section 5.1**
**multiplicity: 0..1**

Phone number associated with the Individual.  The value is expressed as a string according to the format appropriate for the locale.

## 4.4  Organization

Extension of Party for any Organization or Institution.

### 4.4.1  Organization.address

**type: string**                              **type-detail: Section 5.1**
**multiplicity: 0..1**

Mailing address.  The value is expressed as a single string containing all components of the address.

### 4.4.2  Organization.email

**type: string**                              **type-detail: Section 5.1**
**multiplicity: 0..1**

E-mail address of the Organization.

### 4.4.3  Organization.logo

**type: anyURI**                              **type-detail: Section 5.1**
**multiplicity: 0..1**

URL pointer to a graphical logo associated with the Organization.

### 4.4.4  Organization.phone

**type: string**                              **type-detail: Section 5.1**
**multiplicity: 0..1**

Phone number.  The value is expressed as a string according to the format appropriate for the locale.

# 5  Data Types

## 5.1  Base Data Types

Provides a set of standardized primitive data types as well as types for representing quantities ( values with associated units ).  We provide a diagram of the model here, and refer the reader to Section 5 of the VO-DML modeling specification document[8] for more information.



*Figure 17: Base Data Types*
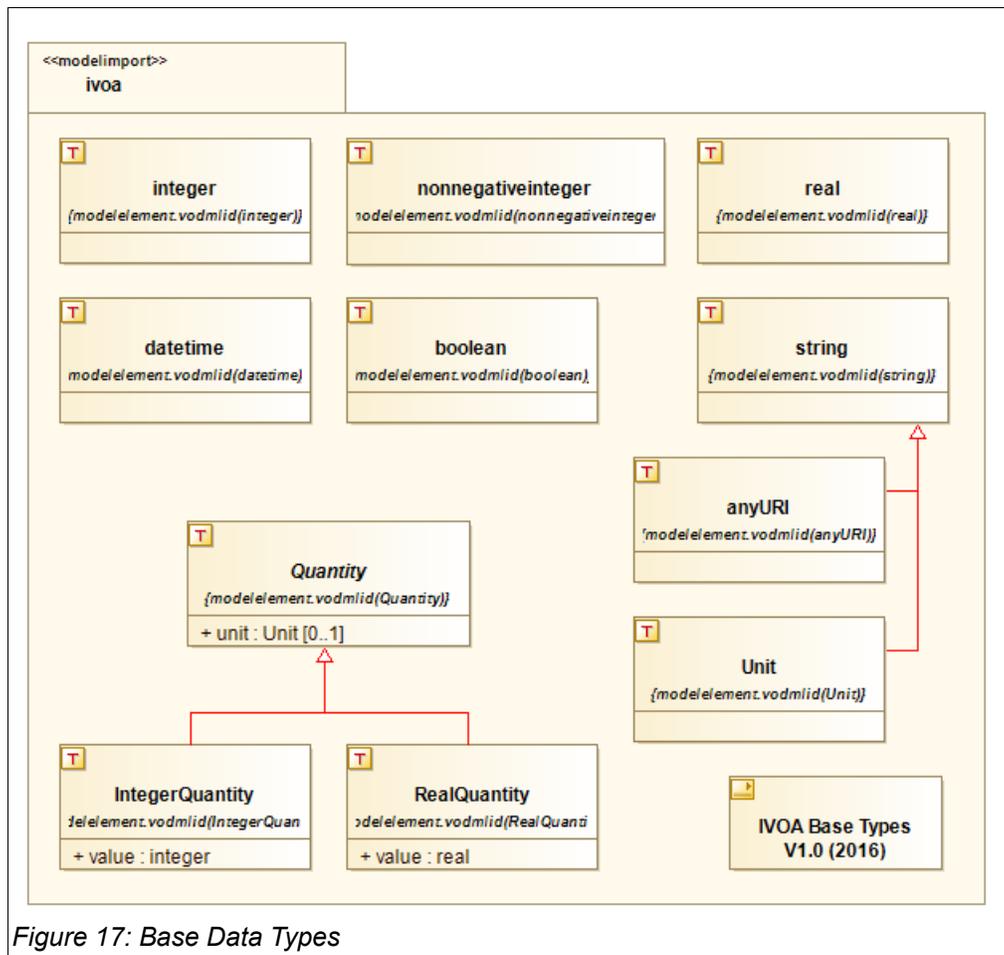
## 5.1.1  Units

This model requires the use of the IVOA VOUnits Standard[9] for representing units of physical quantities. This standard reconciles common practices and current standards for use within the IVOA community.

### 5.1.2 UCDs

This model requires any ucd field to comply with syntax defined in "An IVOA Standard for Unified Content Descriptors"[11].

### 5.1.3 Dates

The 'datetime' datatype is for expressing date-time values. The string representation of a datetime value should follow the FITS convention for representing dates. The FITS standard is effectively ISO8601 format without the "Z" tag to indicate UTC (YYYY-MM-DDThh:mm:ss). Values are nominally expressed in UTC.

## 5.2 Dataset Model DataTypes

The Dataset model has gathered and homogenized data type definitions from previous specifications like ObsCore (DataProductType), VOResource (RightsType), and VODataservice (SpectralBandType).
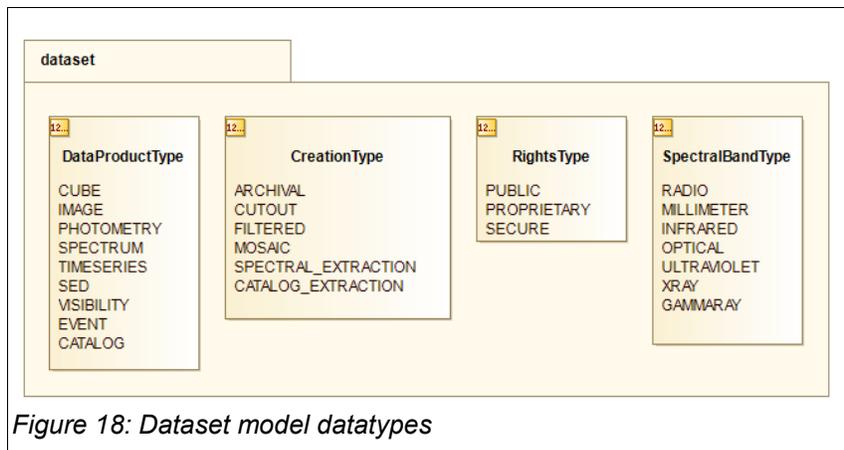


*Figure 18: Dataset model datatypes*

### 5.2.1 CreationType

Enumeration of dataset creation types. Allowed values are:

| Token | Label | Meaning |
|---|---|---|
| ARCHIVAL | archival | Indicates that it is one of a collection of datasets generated in a systematic, homogeneous way and is stored statically (or at least versioned). It will be possible to regenerate this dataset at a later date. The remaining types imply on-the-fly manipulation. |
| CUTOUT | cutout | Indicates that the dataset was created "on-the-fly", by subsetting, but not by modifying values. |
| FILTERED | filtered | May involve excluding data prior to binning into samples, also "on-the-fly" |
| MOSAIC | mosaic | Combines multiple original datasets "on-the-fly" |
| SPECTRAL_EXTRACTION | spectral extraction | Has been extracted, for example, from a spectral data cube. |
| CATALOG_EXTRACTION | catalog extraction | Has been extracted from a catalog. |

## 5.2.2  DataProductType

Enumeration identifying the high level classification of a data product.  Allowed values are:

| Token | Labael | Meaning |
|---|---|---|
| CUBE | cube | A multidimensional astronomical image of three (3) or more axes. |
| IMAGE | image | A two (2) dimensional astronomical image. |
| PHOTOMETRY | photometry | Dataset with spectral coverage with irregular gaps. |
| SPECTRUM | spectrum | Dataset where spectral coverage is the primary attribute, in contiguous bins. e.g. a 1D spectrum or a long slit spectrum. |
| TIMESERIES | timeseries | Dataset presenting some quantity varying as a function of time.  A light curve is a typical example of a timeseries dataset. |
| SED | sed | A spectral energy distribution, an advanced data product often produced by combining data from multiple observations. |
| VISIBILITY | visibility | A visibility (radio) dataset. Typically this is instrumental data, and is often a complex object containing multiple files or other substructures.  A visibility dataset may contain data with spatial, spectral, time, and polarization information for each measured visibility. |
| EVENT | event | An event counting dataset (e.g. X-ray). An event dataset may contain data with spatial, spectral, and time information for each measured event. |
| CATALOG | catalog | A catalog. |

## 5.2.3  RightsType

Enumeration indicating access rights levels.  Allowed values are:

| Token | Label | Meaning |
|---|---|---|
| PUBLIC | public | unrestricted, public access is allowed, without authentication. |
| SECURE | secure | authenticated, public access is allowed. |
| PROPRIETARY | proprietary | only proprietary access is allowed with authentication. |

## 5.2.4 SpectralBandType

Enumeration of generic spectral bands:

| Token | Label | Meaning ( λ=wavelength,  ν=frequency, E=energy) |
|---|---|---|
| RADIO | Radio | $\lambda \geq 10$ mm; $\nu \leq 30$ GHz |
| MILLIMETER | Millimeter | $0.1$ mm $\leq \lambda \leq 10$ mm; $3000$ GHz $\geq \nu \geq 30$ GHz |
| INFRARED | Infrared | $1\ \mu \leq \lambda \leq 100\ \mu$ |
| OPTICAL | Optical | $0.3\ \mu \leq \lambda \leq 1\ \mu$ |
| ULTRAVIOLET | Ultraviolet | $100$ Å $\leq \lambda \leq 3000$ Å; $1.2$ eV $\leq E \leq 120$ eV |
| XRAY | X-ray | $0.1$ Å $\leq \lambda \leq 100$ Å; $0.12$ keV $\leq E \leq 120$ keV |
| GAMMARAY | Gamma-ray | $E \geq 120$keV |

# 6 STC 2.0 Data Model

This model uses (imports) elements from the STC 2.0 data model. This section gives a brief description of the elements used. This content is for reference only, the model document is considered normative and supercedes any descriptions provided here.

## 6.1 Physical Measurements (Meas)



Figure 19: Spatial Domain Coordinates diagram
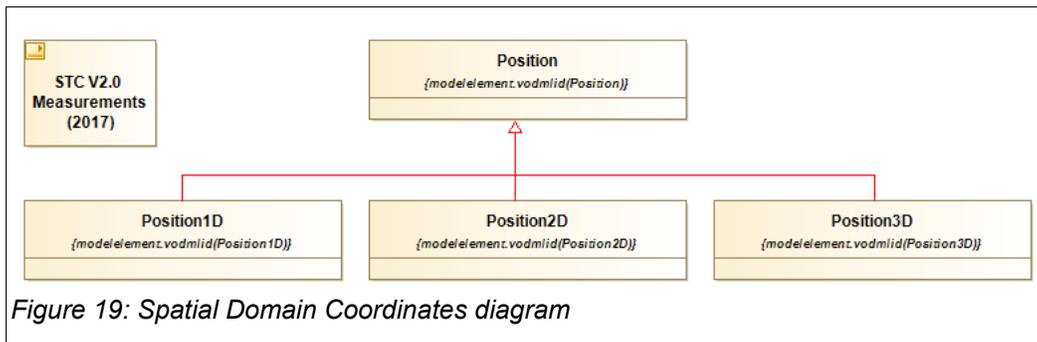
## 6.1.1 Position

Provides a physical location in a spatial coordinate space.

# Appendix A:  Modeling Conventions

## 1   Diagram notation

This model follows the VO-DML modeling practices, however, UML representations may vary depending on the tool used.  Below, we describe the graphical representation of the modeling concepts and relations.
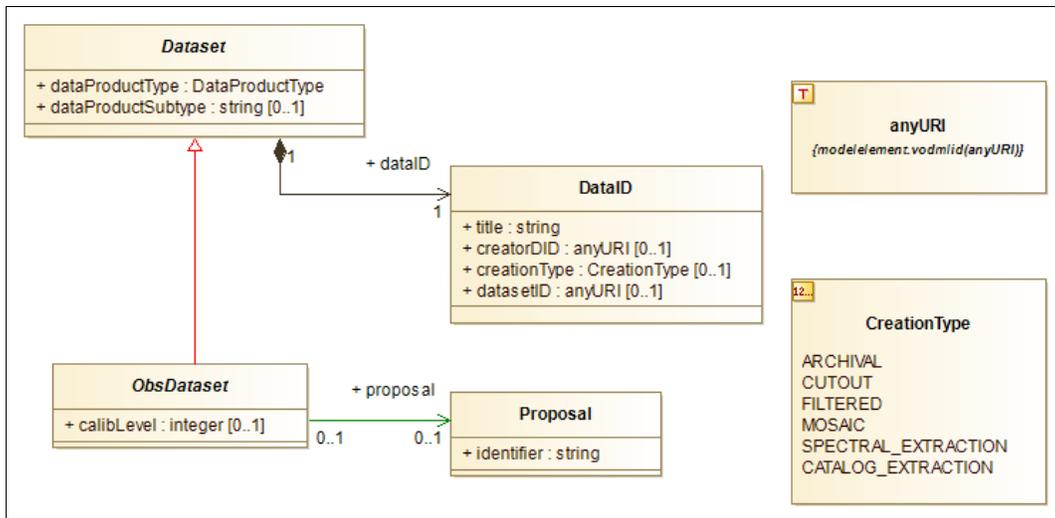


## 1.1  Class

Classes are represented by a plain box.  The class name is annotated in the top window, abstract classes use italic typeface.  Attributes, if any, are listed in the lower panel.  Attributes may only be of primitive type (real, string, etc), a defined DataType, or an Enumeration type.  Relationships to other objects are defined via the composition and reference relation arrows.

## 1.2  DataType

DataTypes are represented by a box shape similar to Class, but annotated with a "T" symbol in the top left corner.

## 1.3  Enumerations

Enumerations are represented by a box shape similar to Class, but annotated with a "1,2.." symbol in the top left corner.  Enumeration Literals (possible values) are listed below the enumeration class name.

## 1.4  Generalization

Generalizations are represented by a red line, with open triangle at the end of the source, or more general, object.

## 1.5  Composition

The composition relation is indicated by a black line with a solid diamond attached to the containing object, and an arrow pointing to the object being contained. The composition relation is very tight, where the container is responsible for the creation and existence of the target.  Any object may be in no more than one composition relation with any container.  The attribute name for the composition relation is annotated at the destination of the relation (e.g. "+ dataID").  This is typically a lower-cased version of the destination class name, but this is not required.

## 1.6  Reference

The reference relation is indicated by a green line, with an arrow pointing to the object being referenced.  The reference relation is much looser than composition, the container has no ownership of the target, but merely holds a pointer, or other indirect connection to it.   The attribute name is annotated at the destination of the relation ( e.g. "+ proposal").  This is typically a lower-cased version of the destination class name, but may be another name indicating the role that the class is playing in this context.

## 1.7  Multiplicity

All attributes and relations have a multiplicity associated with them.  For attributes, the multiplicity is contained within brackets just after the attribute name.  If no bracket is displayed, this is equivalent to '[1]'.

+ 1 = one and only one value must be provided.

+ 0..1 = zero or one value may be provided.

+ * =  zero or more values may be provided (open ended).

# 2   Model Identification metadata

Interoperability of datasets requires that there be a standardized method for identifying the specific type of dataset, and which model(s) and versions thereof it conforms to.  These elements are not properties of the dataset, but rather, of the Model itself.  We provide this information via stereotypes assigned to the model packages (e.g. Dataset, Char, STC, IVOA ).

## 2.1  Model stereotype

The Model stereotype (<<model>>) consists of a set of Model properties which identify a particular model and its dependencies.  Each model should specifically state the appropriate values for these properties.

### 2.1.1  title:string[1]

The model title or long name.

### 2.1.2  version:string[1]

The version of this model.  To be represented as a string with format "<version>.<subversion>"

### 2.1.3  name:string[1]

Sometimes referred to as 'namespace' or 'prefix', this is a tag which is used to label elements of a particular model.  The value must match the name of the model package itself.  This string identifies the particular model type (eg. ds, char, stc ).  Each model must declare a prefix string

47

which is unique within the IVOA to tag elements from those models. A typical use of the prefix is in the construction of element UTYPE strings.

### 2.1.4  url:anyURI[1]

A URL from which the full model description may be obtained (e.g. XML schema).

### 2.1.5  imports:Import[*]

Here, we specify which the models on which this model is dependent. This model uses and/or extends elements from the Characterisation and STC Data models. In this document, we provide descriptions and supporting information about usage of these objects in a particular context. The originating documents, however, remain the definitive source for element definitions.

## 2.2  Import Stereotype

The <<modelimport>> stereotype is attached to Packages representing imported models. It identifies the model by name, and provides URLs from which the full description may be obtained.

### 2.2.1  name:string[1]

The name of the imported model. This name MUST match the 'name' property of imported model's Model metadata.

### 2.2.2  version:string[1]

The version of this model. To be represented as a string with format "<version>.<subversion>"

### 2.2.3  url:anyURI[1]

A URL from which the full model description may be obtained (e.g. XML schema).

# 3  Extensibility

There is no formal mechanism in the IVOA defining how users may extend models with their own content. However, the above Model identification metadata provides a simple means to do so. Using this process, a user would model their content as an extension of the IVOA standard.

## 3.1  Model

### 3.1.1  name

The user-defined model would need a name unique from that of the standard.

### 3.1.2  prefix

A unique prefix must be defined for the user-defined model elements. Users must take care not to make use of prefix tags which are associated with current IVOA standards, (e.g. 'cha', 'spec', 'ssa', 'stc'). At the time of this writing, there is no central repository of reserved namespace strings.

### 3.1.3  imports

The user defined model should declare the IVOA standard being extended as an imported model. Fields for the imported model name and url may be obtained from that standard's documentation.

## 3.2  Scope

We permit any object modeled in this document to be extended with user-defined content, with the following restrictions:

• Follow VO-DML modeling practices.

• Values of extended content must be consistent with the content of modeled data. That is, using the IVOA base primitive types, Quantity, and STC Coordinates as appropriate.

• Since extended content, by definition, does not follow the corresponding model, it is not possible for general applications to interpret complex structures within that content.  It is, therefore, recommended that users define extended content in such a way as to avoid ambiguity between its components.

## 3.3  Support

Applications should, but are not required to, provide the following support for extended content:

• Retain existence of extended content, including namespace and UTypes.

• Retain association with modeled component.

• Provide access to extended content by users.

# Appendix C: Dataset Metadata Model Summary

| Dataset Model Identification | | | | |
|---|---|---|---|---|
| **Model Element** | **Datatype** | **Mult.** | **Meaning** | **value** |
| | | | | |
| **Model identification** | | | | |
| Model | | | | |
| model.title | string | 1 | Data model name | "Dataset Metadata" |
| model.version | string | 1 | Data model version | "1.0" |
| model.name | string | 1 | Data model short name, prefix tag | "ds" |
| model.url | anyURI | 1 | Reference URL for model | <TBD> |
| **Imported Model** | | | | |
| modelimport.name | string | 1 | Imported model name | "meas" |
| modelimport.version | string | 1 | Imported model version | "2.0" |
| modelimport.url | anyURI | 1 | Reference URL for imported model | "https://volute.g-vo/svn/trunk/projects/dm/STC/vo-dml/STC_meas_alt-v2.0.vo-dml.xml" |
| modelimport.documentationURL | anyURI | 1 | Reference URL for imported model documentation | "https://volute.g-vo/svn/trunk/projects/dm/STC/vo-dml/STC_meas_alt-v2.0.html" |
| **Imported Model** | | | | |
| modelimport.name | string | 1 | Imported model name | "ivoa" |
| modelimport.version | string | 1 | Imported model version | "1.0" |
| modelimport.url | anyURI | 1 | Reference URL for imported model | "https://volute.g-vo/svn/trunk/projects/dm/vo-dml/models/ivoa/vo-dml/IVOA-v1.0.vo-dml.xml" |
| modelimport.documentationURL | anyURI | 1 | Reference URL for imported model documentation | "https://volute.g-vo/svn/trunk/projects/dm/vo-dml/models/ivoa/vo-dml/IVOA-v1.0.html" |

| Dataset Model Summary | | | | |
|---|---|---|---|---|
| **Model Element** | **Datatype** | **Mult.** | **Meaning** | **UCD1+** |
| **Dataset Model Elements** | | | | |
| Collection | Collection | | Descriptive metadata of a compatible grouping | |
| Collection.name | string | 1 | Collection name | |
| Contact | Contact | | Extension of party Role. Entity to contact for assistance. | |
| Contact.party | *Individual* | 0..1 | Entity associated with Contact role. | |
| Contributor | Contributor | | Extension of Role.  Entity participated in dataset generation | |
| Contributor.acknowledgement | string | 1 | Acknowledgment expression for the contributor | |
| Creator | Creator | | Extension of Role. Entity which created the dataset | |
| Curation | Curation | | | |
| Curation.contact | Contact | 0..1 | Contact entity for curation service | |

| Dataset Model Summary | | | | |
|---|---|---|---|---|
| **Model Element** | **Datatype** | **Mult.** | **Meaning** | **UCD1+** |
| Curation.publisher | Publisher | 1 | Publisher | meta.curation |
| Curation.publisherDID | anyURI | 0..1 | Publisher specified dataset ID | meta.ref.url;meta.curation |
| Curation.reference | Publication | 0..* | Reference to publication associated with the dataset | |
| Curation.releaseDate | datetime | 0..1 | Date curated dataset last modified | time.release |
| Curation.rights | RightsType | 0..1 | Level of access granted | meta.code |
| Curation.version | string | 0..1 | Publisher version of the dataset | meta.version;meta.curation |
| DataID | DataID | | | |
| DataID.collection | Collection | 0..* | Collection(s) to which the dataset is a member | meta.id |
| DataID.contributor | Contributor | 0..* | Contributor(s) | |
| DataID.creationType | CreationType | 0..1 | Dataset creation type | |
| DataID.creator | Creator | 0..1 | Institution or entity which created the dataset | meta.curation |
| DataID.creatorDID | anyURI | 0..1 | Creator defined Dataset Identifier | meta.id |
| DataID.datasetID | anyURI | 0..1 | IVOA Dataset Identifier | meta.id;meta.dataset |
| DataID.date | datetime | 0..1 | Data processing/creation date | time.epoch;meta.dataset |
| DataID.title | string | 1 | Dataset title | meta.title;meta.dataset |
| DataID.version | string | 0..1 | Version of dataset | meta.version;meta.dataset |
| Dataset | *Dataset* | | | |
| Dataset.curation | Curation | 1 | Dataset curation metadata | |
| Dataset.dataID | DataID | 1 | Dataset identification metadata | |
| Dataset.dataProductSubType | string | 0..1 | Dataset subtype | meta.id |
| Dataset.dataProductType | DataProductType | 1 | Dataset or segment type | meta.id |
| Publication | Publication | | Referenceable publication | |
| Publication.refCode | string | 1 | URL or Bibcode of the publication | meta.bib.bibcode |
| Publisher | Publisher | | | |
| Publisher.publisherID | anyURI | 0..1 | URI for VO Publisher | meta.ref.url;meta.curation |

| Dataset Model Summary | | | | |
|---|---|---|---|---|
| **Model Element** | **Datatype** | **Mult.** | **Meaning** | **UCD1+** |
| **Observation Model Elements** | | | | |
| AstroTarget | AstroTarget | | Astronomical target | |
| AstroTarget.objectClass | string | 0..1 | Target or object class | src.class |
| AstroTarget.redshift | real | 0..1 | Target redshift | src.redshift |
| AstroTarget.spectralClass | string | 0..1 | Object spectral class | src.spType |
| AstroTarget.VarAmpl | real | 0..1 | Target variability amplitude - typical | src.var.amplitude |
| BaseTarget | BaseTarget | | | |
| BaseTarget.description | string | 0..1 | Target descriptive text | meta.note;src |
| BaseTarget.name | string | 1 | Target name | meta.id;src |
| BaseTarget.position | Position | 0..1 | Target location (eg: RA, DEC) | pos[.eq];src |
| Derived | Derived | | | |
| Derived.derivedElement | DerivedElement | 0..* | property derived from analysis of data content | |
| DerivedScalar | DerivedScalar | | | |
| DerivedScalar.name | string | 1 | name of derived property | |
| DerivedScalar.value | Quantity | 1 | value of derived property | |
| Facility | Facility | | Extension of Role.  Associates Organization as Facility | |
| Facility.party | Organization | 1 | Organization responsible for performing the observation | meta.id;instr.tel |
| Instrument | Instrument | | Instrument metadata | |
| Instrument.name | string | 1 | Instrument ID | meta.id;instr |
| ObsConfig | ObsConfig | | | |
| ObsConfig.bandpass | string | 0..1 | General regime in spectral domain | instr.bandpass |
| ObsConfig.dataSource | string | 0..1 | Original source/type of data | |
| ObsConfig.facility | Facility | 0..1 | Metadata for the facility performing the observation | |
| ObsConfig.instrument | Instrument | 0..1 | Metadata for the Instrument which created the data | |
| ObsDataset | ObsDataset | | | |
| ObsDataset.calibLevel | integer | 0..1 | Calibration level | meta.code;obs.calib |
| ObsDataset.derived | Derived | 0..1 | Derived metadata | |
| ObsDataset.obsConfig | ObsConfig | 0..1 | Observatino configuration | |
| ObsDataset.proposal | Proposal | 0..1 | Proposal information | |
| ObsDataset.target | *BaseTarget* | 1 | | |
| Observation | Observation | | | |
| Observation.obsConfig | ObsConfig | 1 | Observation configuration metadata | |
| Observation.observationID | string | 1 | Observation ID | |
| Observation.proposal | Proposal | 0..1 | Proposal which spawned the observation | |
| Observation.result | ObsDataset | 0..* | Dataset(s) resulting from the observation | |
| Observation.target | BaseTarget | 1 | Target or goal of the observation | |
| Proposal | Proposal | | | |
| Proposal.Identifier | string | 1 | Proposal ID | meta.id;obs.proposal |

| Dataset Model Summary | | | | |
|---|---|---|---|---|
| **Model Element** | **Datatype** | **Mult.** | **Meaning** | **UCD1+** |
| Target | Target | | Generic Target | |
| Target.objectClass | string | 0..1 | Target or object class | src.class |
| | | | | |
| **Party Package Elements** | | | | |
| Individual | *Individual* | | Entity described as a single individual or representative | |
| Individual.address | string | 0..1 | address of the Individual | |
| Individual.email | string | 0..1 | email address of individual | meta.ref.url;meta.email |
| Individual.phone | string | 0..1 | phone number for the Individual | |
| Organizaton | Organization | | Entity represented by an organization | |
| Organization.address | string | 0..1 | address of the organization | |
| Organization.email | string | 0..1 | email address for the organization | |
| Organization.logo | anyURI | 0..1 | URL for organization logo | meta.ref.url |
| Organization.phone | string | 0..1 | phone number of the organization | |
| Party | *Party* | | Head of the Entity object tree | |
| Party.name | string | 1 | Entity name | |
| Role | *Role* | | Role being associated with a particular Party | |
| Role.party | *Party* | 1 | Entity associated with the role. | |

# References

[1] *Resource Metadata for the Virtual Observatory*: Version 1.12, 02 March 2007
http://www.ivoa.net/Documents/latest/RM.html
[2] "IVOA Identifiers": Version 1.12, 14 March 2007
http://www.ivoa.net/Documents/latest/IDs.html
[3] "Data Model for Astronomical DataSet Characterisation": Version 1.13
http://www.ivoa.net/Documents/REC/DM/CharacterisationDM-20080325.pdf
[4] "ST-ECF newsletter, issue #42", June 2007
http://www.spacetelescope.org/about/further_information/newsletters/html/newsletter42.html
[5] "DER SNR: A simple and general spectroscopic signal-to-noise measurement algorithm";
http://www.stecf.org/software/ASTROsoft/DER_SNR
[6] "Simulation Data Model": Version 1.0, 03 May 2012;
http://www.ivoa.net/documents/SimDM/20120503/REC-SimulationDataModel-1.00-20120503.pdf
[7] "An encoding system to represent stellar spectral classes in archival databases and catalogs":
Version 1.04, 2011 Dec 15
http://www.ivoa.net/documents/Notes/SpectClasses/20111215/SpectClasses-20111215.pdf
[8] "VO-DML a consistent modeling language for IVOA data models": Version 1.00-20170925
http://volute.googlecode.com/svn/trunk/projects/dm/vo-dml/doc/VO-DML-PR-v1.0-20170925.pdf
[9] "Units in the VO": Version 1.0
http://www.ivoa.net/Documents/VOUnits/20120820/PR-VOUnits-1.0-20120820.pdf
[10] "The UCD1+ controlled vocabulary": Version 1.23
http://www.ivoa.net/Documents/REC/UCD/UCDlist-20070402.pdf
http://cdsweb.u-strasbg.fr/UCD/ucd1p-words.txt
[11] "An IVOA Standard for Unified Content Descriptors": Version 1.10
http://www.ivoa.net/Documents/REC/UCD/UCD-20050812.pdf
http://www.ivoa.net/Documents/latest/UCD.html