International
Virtual
Observatory
Alliance

# MANGO: A Component and Association Based Model for representing data for astronomical sources

# Version 1.0

## IVOA Working Draft 2020-07-15

Working group
DM
This version
http://www.ivoa.net/documents/MANGO/20200715
Latest version
http://www.ivoa.net/documents/MANGO
Previous versions
This is the first public release

Author(s)
François Bonnarel, Gilles Landais, Laurent Michel, Jesus Salgado, Mireille Louys, Marco Molinaro
Editor(s)
Laurent Michel

## Abstract

The MANGO model proposes a flexible way to expose data related to astronomical source objects in an interoperable way. It takes into account the huge diversity of source data in terms of feature description, format and usage. The MANGO model attaches identifiers on astronomical sources and associates to each a flexible set of parameters (e.g. observed physical quantities) and other information like e.g. spectra, time series or preview images. Parameters usually appear in the columns of a source catalogue. Additional data products are bound to the source to contribute to the science analysis and enhance data understanding. Mango object parameters are built upon classes or extended classes of the IVOA Measure and Coordinates data model. Associated data can be simple URLs, VO service endpoints or VO data model instances. The roles of both parameters and associated data are qualified by semantic tags

## 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".

A list of current IVOA Recommendations and other technical documents can be found at http://www.ivoa.net/documents/.

## Contents

**1 Introduction** **3**
1.1 Role within the VO Architecture . . . . . . . . . . . . . . . . 4

**2 Representing observed astronomical objects : Use Cases and Requirements** **4**
2.1 Use Cases . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
2.1.1 Gaia . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
2.1.2 Euclid . . . . . . . . . . . . . . . . . . . . . . . . . . 5
2.1.3 Exoplanets . . . . . . . . . . . . . . . . . . . . . . . . 5
2.1.4 Morphologically Complex Structures . . . . . . . . . . 7
2.1.5 Chandra Archive . . . . . . . . . . . . . . . . . . . . . 8
2.1.6 Vizier catalog archive . . . . . . . . . . . . . . . . . . 8
2.1.7 Client on (MT behalf) . . . . . . . . . . . . . . . . . . 10
2.1.8 Xmatch tool . . . . . . . . . . . . . . . . . . . . . . . 11
2.2 Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . 13

2.2.1 Parameters and Associated Data . . . . . . . . . . . . 13
2.2.2 R01: Supported Quantities . . . . . . . . . . . . . . . 13
2.2.3 R02: Parameters . . . . . . . . . . . . . . . . . . . . . . 13
2.2.4 R03: Associated Data . . . . . . . . . . . . . . . . . . . 14

**3 Model Overview 15**
3.1 Parameters . . . . . . . . . . . . . . . . . . . . . . . . . . . 15
3.1.1 Parameters Identification . . . . . . . . . . . . . . . . 16
3.1.2 Measure Extension . . . . . . . . . . . . . . . . . . . . 16
3.2 Associated Data . . . . . . . . . . . . . . . . . . . . . . . . 17
3.2.1 AssociatedData Identification . . . . . . . . . . . . . . 18
3.2.2 Associated Model Instances . . . . . . . . . . . . . . . 20

**4 Model: mango 21**
4.1 AssociatedData (Abstract) . . . . . . . . . . . . . . . . . . 21
4.1.1 AssociatedData.semantic . . . . . . . . . . . . . . . . . 21
4.1.2 AssociatedData.dataType . . . . . . . . . . . . . . . . 21
4.1.3 AssociatedData.description . . . . . . . . . . . . . . . 21
4.2 AssociatedMangoInstance . . . . . . . . . . . . . . . . . . . 21
4.2.1 AssociatedMangoInstance.mangoInstance . . . . . . 21
4.3 ModelInstance . . . . . . . . . . . . . . . . . . . . . . . . . 22
4.4 Parameter . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
4.4.1 Parameter.semantic . . . . . . . . . . . . . . . . . . . 22
4.4.2 Parameter.ucd . . . . . . . . . . . . . . . . . . . . . . . 22
4.4.3 Parameter.description . . . . . . . . . . . . . . . . . . 22
4.4.4 Parameter.measure . . . . . . . . . . . . . . . . . . . . 22
4.4.5 Parameter.associatedParameters . . . . . . . . . . . . 23
4.5 Source . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23
4.5.1 Source.identifier . . . . . . . . . . . . . . . . . . . . . . 23
4.5.2 Source.associatedDataDock . . . . . . . . . . . . . . . 23
4.5.3 Source.parameterDock . . . . . . . . . . . . . . . . . . 23
4.6 VOModelInstance . . . . . . . . . . . . . . . . . . . . . . . . 23
4.6.1 VOModelInstance.ivoid . . . . . . . . . . . . . . . . . 23
4.6.2 VOModelInstance.modelUrl . . . . . . . . . . . . . . . 24
4.6.3 VOModelInstance.modelName . . . . . . . . . . . . . 24
4.6.4 VOModelInstance.modelDoc . . . . . . . . . . . . . . 24
4.6.5 VOModelInstance.modelInstance . . . . . . . . . . . . 24
4.7 VOService . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24
4.7.1 VOService.ivoid . . . . . . . . . . . . . . . . . . . . . . 24
4.8 VocabularyTerm . . . . . . . . . . . . . . . . . . . . . . . . 24
4.8.1 VocabularyTerm.uri . . . . . . . . . . . . . . . . . . . 25

4.8.2 VocabularyTerm.label . . . . . . . . . . . . . . . . . . 25
4.9 WebEndpoint . . . . . . . . . . . . . . . . . . . . . . . . . . 25
4.9.1 WebEndpoint.ContentType . . . . . . . . . . . . . . . . 25
4.9.2 WebEndpoint.url . . . . . . . . . . . . . . . . . . . . . 25
4.10 LinkSemantic . . . . . . . . . . . . . . . . . . . . . . . . . . 25
4.11 ParamSemantic . . . . . . . . . . . . . . . . . . . . . . . . . 26
4.12 ShapeFrame . . . . . . . . . . . . . . . . . . . . . . . . . . . 26

**5 Package: errors** **27**
5.1 CorrMatrix2x2 . . . . . . . . . . . . . . . . . . . . . . . . . 27
5.1.1 CorrMatrix2x2.correlation . . . . . . . . . . . . . . . . 27
5.1.2 CorrMatrix2x2.diagMatrix . . . . . . . . . . . . . . . . 27
5.2 Correlated2D1D . . . . . . . . . . . . . . . . . . . . . . . . . 27
5.2.1 Correlated2D1D.correlation2_1 . . . . . . . . . . . . . 27
5.3 CorrelatedError . . . . . . . . . . . . . . . . . . . . . . . . . 27
5.3.1 CorrelatedError.ucd . . . . . . . . . . . . . . . . . . . 28
5.4 CorrelatedError1D1D . . . . . . . . . . . . . . . . . . . . . . 28
5.4.1 CorrelatedError1D1D.correlation1_1 . . . . . . . . . . 28
5.5 CorrelatedError1D2D . . . . . . . . . . . . . . . . . . . . . . 28
5.5.1 CorrelatedError1D2D.correlation1_1 . . . . . . . . . . 28
5.5.2 CorrelatedError1D2D.correlation1_2 . . . . . . . . . . 28
5.6 CorrelatedError2D2D . . . . . . . . . . . . . . . . . . . . . . 29
5.6.1 CorrelatedError2D2D.correlation2_1 . . . . . . . . . . 29
5.6.2 CorrelatedError2D2D.correlation2_2 . . . . . . . . . . 29
5.7 DiagMatrix2x2 . . . . . . . . . . . . . . . . . . . . . . . . . . 29
5.7.1 DiagMatrix2x2.diagMatrix . . . . . . . . . . . . . . . . 29
5.8 Ellipse . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29
5.8.1 Ellipse.semiMajorAxis . . . . . . . . . . . . . . . . . . 30
5.8.2 Ellipse.semiMinorAxis . . . . . . . . . . . . . . . . . . 30
5.8.3 Ellipse.positionAngle . . . . . . . . . . . . . . . . . . . 30
5.9 MultParamErro2D (Abstract) . . . . . . . . . . . . . . . . . . 30
5.10 MultiParamError (Abstract) . . . . . . . . . . . . . . . . . . . 30
5.10.1 MultiParamError.confidenceLevel . . . . . . . . . . . . 30
5.10.2 MultiParamError.unit . . . . . . . . . . . . . . . . . . 31
5.10.3 MultiParamError.correlatedErrors . . . . . . . . . . . . 31
5.11 MultiParamError1D . . . . . . . . . . . . . . . . . . . . . . . 31
5.11.1 MultiParamError1D.sigma . . . . . . . . . . . . . . . . 31
5.12 DiagElems2x2 . . . . . . . . . . . . . . . . . . . . . . . . . . 31

**6 Package: measures 32**
6.1 Flag . . . 32
6.1.1 Flag.coord . . . 32
6.2 HardnessRatio . . . 32
6.2.1 HardnessRatio.coord . . . 32
6.3 LonLatSkyPosition . . . 32
6.3.1 LonLatSkyPosition.coord . . . 32
6.4 Orbit . . . 32
6.4.1 Orbit.coord . . . 33
6.5 Photometry . . . 33
6.5.1 Photometry.coord . . . 33
6.6 Redshift . . . 33
6.6.1 Redshift.coord . . . 33
6.7 Shape . . . 33
6.7.1 Shape.coord . . . 33
6.8 StringGenericMeasure . . . 33
6.8.1 StringGenericMeasure.coord . . . 34
6.9 VocabGenericMeasure . . . 34
6.9.1 VocabGenericMeasure.coord . . . 34

**7 Package: coordinates 35**
7.1 FlagCoord . . . 35
7.1.1 FlagCoord.status . . . 35
7.2 FlagState . . . 35
7.2.1 FlagState.value . . . 35
7.2.2 FlagState.label . . . 35
7.3 FlagSys . . . 35
7.3.1 FlagSys.statusLabels . . . 36
7.4 HardnessRatioCoord . . . 36
7.4.1 HardnessRatioCoord.hardnessRatio . . . 36
7.5 HardnessRatioFrame . . . 36
7.5.1 HardnessRatioFrame.low . . . 36
7.5.2 HardnessRatioFrame.high . . . 36
7.6 HardnessRatioSys . . . 37
7.7 LonLatCoordSys . . . 37
7.8 LonLatPoint . . . 37
7.8.1 LonLatPoint.longitude . . . 37
7.8.2 LonLatPoint.latitude . . . 37
7.9 ObjectTypeCoord . . . 38
7.9.1 ObjectTypeCoord.objectType . . . 38
7.10 OrbitCoord . . . 38

7.11 PhotFilter . . . 38
7.11.1 PhotFilter.name . . . 38
7.11.2 PhotFilter.zeroPointFlux . . . 38
7.11.3 PhotFilter.magnitudeSystem . . . 39
7.11.4 PhotFilter.effectiveWavelength . . . 39
7.11.5 PhotFilter.unit . . . 39
7.11.6 PhotFilter.bandWidth . . . 39
7.12 PhotometryCoord . . . 39
7.12.1 PhotometryCoord.luminosity . . . 39
7.13 PhotometryCoordSys . . . 40
7.14 RedshiftCoord . . . 40
7.14.1 RedshiftCoord.redshift . . . 40
7.15 RedshiftSys . . . 40
7.16 ShapeCoord . . . 40
7.16.1 ShapeCoord.shape . . . 40
7.17 ShapeSys . . . 40
7.17.1 ShapeSys.frame . . . 41
7.18 StringCoordinate . . . 41
7.18.1 StringCoordinate.cval . . . 41
7.19 VocabCoordinate . . . 41
7.20 ShapeFrame . . . 41

**8 TAP and MANGO** **41**
8.1 Storing MANGO Catalogs in TAP . . . 41
8.2 *MANGOCore* Table . . . 42

**A Imported Models Instance** **42**

**B Changes from Previous Versions** **42**

## Acknowledgments

We would like to thank all those who took the time to present their own use cases (INAF, CDS, CFA, ESAC) on which the model has been built. We would also like to thank all the people having tested MANGO on their own data.

## Model Name

This model was initially named with a very explicit but hard to remember acronym, CAB-MSD standing for Component and Association Based Model for Source Data. We decided later to rename it MANGO with reference to the inside out MANGO picture used to introduce the model in Groningen. As the tradition requires that such unexpected names are acronyms, let's assume that MANGO stands for Metadata ANnotation for Generic Objects (in astronomy).

## Conformance-related definitions

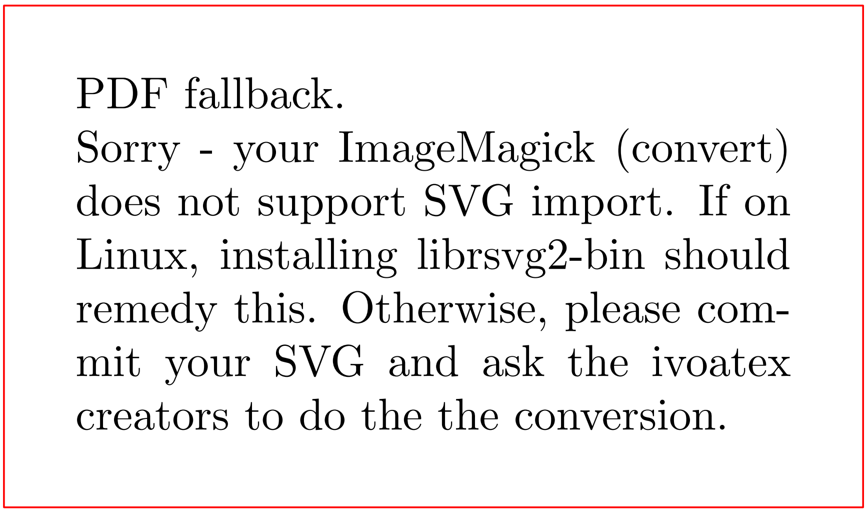
The words "MUST", "SHALL", "SHOULD", "MAY", "RECOMMENDED", and "OPTIONAL" (in upper or lower case) used in this document are to be interpreted as described in IETF standard RFC2119 (**?**).

The *Virtual Observatory (VO)* is a general term for a collection of federated resources that can be used to conduct astronomical research, education, and outreach. The International Virtual Observatory Alliance (IVOA) is a global collaboration of separately funded projects to develop standards and infrastructure that enable VO applications.

## 1 Introduction

Modeling data collected to study astronomical source objects has been a long term concern for the DM working group and more generally for the IVOA. In the past years, there were some proposals to design a global model for sources (**?**) as well as for catalogs (**?**). Other proposals, more model-agnostic, were focused on the data annotation in VOTables (**?**) (**?**). In this case the goal was no longer to design a source model but to provide a complete description of individual quantities (positions, velocity, fluxes, magnitudes. . . ). None of these proposals have come to completion.

The source DM issue resurfaced at the spring 2018 Interop in Victoria during an hands-on session focusing on the tools available to work with VO data models and especially with VO-DML. The goal of this session was to annotate data from different origins in order to make them interoperable with each other. One of the main concerns outside de tools necessary to workout this notation was the lack of models for source objects. This is a big paradox in the VO world: source data which represent the basic building blocks of astronomers' work, is not modelled. This paradox can be explained by the fact that the observation of source objects is multi-faceted. In a general way, the way features for source data are described and organised depends on the targeted science case. Principal investigators and archive designers set up the data profile and structure it according to this goal which varies from



PDF fallback.
Sorry - your ImageMagick (convert) does not support SVG import. If on Linux, installing librsvg2-bin should remedy this. Otherwise, please commit your SVG and ask the ivoatex creators to do the the conversion.

*Figure 1:* Architecture diagram for this document

one project to another. Therefore this diversity cannot be served by a single static data model describing a source item for all possible cases. Having a global source model would lead to a very complex solution not usable in practice.

This standard proposes to overcome this paradox and presents a template model gathering independent components from VO existing models together with VO data products and files embedded on demand in a container. The template supports fine grain association by composing classes as well as coarse grain relations to data products and files distributed within projects archive. Mango is not designed to describe what a source is but to help clients to discover and to understand the quantities available for a particular source instance.

### 1.1 Role within the VO Architecture

Fig. 1 shows the role this document plays within the IVOA architecture (?).

## 2 Representing observed astronomical objects : Use Cases and Requirements

### 2.1 Use Cases

The following uses-cases have been collected since 2019 from representatives of various astronomical missions, archive designers and tools developers. The contribution was totally open. This gave a good picture of the needs but we don not pretend that everything will be supported by this first version.

The physical parameters recorded for each source in the various projects are listed below:

### 2.1.1 Gaia

- identifier
- sky reference position
- proper motion
- parallax and distance
- source extension
- radial velocity
- redshift
- photometry
- date of observation ?
- correlation ( with other sources ?)
- multiple detection

### 2.1.2 Euclid

- identifier
- sky position
- correlation with Gaia counterpart?
- photometry (ground + satellite )
- morphology class?
- redshift
- photometric redshift

### 2.1.3 Exoplanets

Annotation of (exo-)planetary records in catalogues requires some specific metadata or model.

This was initially made clear when trying to allow discoverability of exoplanets time series (?) where (exercising the ObsCore model with specific use cases) information like stellar host characteristics together with stellar system details turned out to be required for a proper description of the catalogue records.

The use cases identified, besides some refinement needed in the ObsCore time axis annotation, the following metadata:

- the degree of confidence in the detection: exoplanets candidates w.r.t. confirmed ones, plus last update of the record content ;
- the method used in the discovery (since it affects the available stellar system description parameters);
- a set of stellar host characteristics (besides sky coordinates): activity, mass, type, metallicity, age, some systemic values, like the global RV (radial velocity) of the system, and so on;
- (exo-)planet parameters, like mass, orbital period, orbit's eccentricity, RV semi-amplitude, time at periastron (for RV detections) or central transit time (for transit method), longitude of periastron, and so on.

The scenario gained some further complexity, but also better stated the idea of modelling datasets and catalogue records in the (exo-)planetary systems subdomain, incorporating requirements from exoplanets atmosphere simulations and (first efforts) observations.

Specific metadata additions were not specified exactly, but a draft model to describe stellar systems was developed, specifically trying to solve the issue in adding metadata elements to describing orbiting celestial objects (?).

The model identified the main concepts and classes as:

- Celestial Object, typed/subclassed as: Star, Binary Star, Planet, Satellite, Brown Dwarf, Trojan, and so on;
- Orbit, as the class keeping the information needed to describe the orbit of a couple (or more) Celestial Objects.

Specific (exo-)planets metadata included: atmosphere with molecular composition, bulk details (mass, radius, . . . ).

A catalogue service that could make use of model metadata is the ExoMerCat (?) catalogue (or the other available exoplanets' catalogues), already available in a VO-aware solution[1], but using only simple UCD description of

[1] ivo://ia2.inaf.it/catalogues/exomercat

the exposed information.

**TODO:**
mention the involved projects : examples ? GAPS, TESS?

**TODO:**
For GAPS I can add specifics later, if implementation can (re)start

### 2.1.4 Morphologically Complex Structures

The ViaLactea Knowledge Base (VLKB, see **?**) is a set of data resources and services built up to study the star formation regions and processes in the Milky Way. Besides 2-D images and 3-D radial velocity cubes, the VLKB exposes a bunch of source catalogues. These catalogues can be categorised as:

- compact sources catalogues: where each source can be simply described by position and (usually) ellipsoidal extension plus some photometric flux(es);
- a band merged catalogue: where each source record combines the single band source of the previous type into a multi-center/multi-positional record;
- diffuse objects of two types, bubbles and filaments, where the positional description cannot be simplified by a point in the sky and some geometric error/shape value.

The first type of catalogue presents only a small issue, related to the ellipsoidal description of the extension of the single records/sources. For these catalogues the ellipsoidal (or circular) counterpart of the position is not to be considered as a positional uncertainty, but actually some level of confidence on the actual extension of the source at the observed frequency band, thus, besides a position on the celestial sphere, a minor axis, a major axis and a positional angle are needed to describe the source.

The band merged catalogue requires an extra step, that is the aggregagtion of multiple single band sources (potentially degenerate depending on the observational waveband) to be able to provide a sort of SED of the aggregated source.

A different type of challenge is presented by filaments and bubbles where (not approximating bubbles to simple circles) the source morphology requires or a complex geometrical shape, like single or multiple polygons, or a description based on tessellation (where the order of the tessels should fit the resolution/uncertainty of the source borders). Moreover, while bubbles can be fully described by a centroid and a single complex (multi-)polygon or set of tessels, filaments are themselves complex objects such that their description includes so-called branches, spines and nodes (areas, broken-lines, points in spherical geometry).

Thus, a model that supports description of such catalogues will need a way to describe sources with:

- non-point-like positions;
- extended complex area, possibly as multiple detached areas;
- aggregation of sub-parts (that can be heterogeneous).

The VLKB tried[2] to describe, both by (custom) contour polygons and MOCs, the bubbles and filaments having the main use case of cross-matching or distance measuring diffuse objects w.r.t. compact sources. However, work progressend slowly and anyway the proper model annotations were never investigated properly.

MANGO could be useful to serve nicely annotated and usable records for sources as the ones described above.

### 2.1.5 Chandra Archive

This X-ray mission has produced a very large catalog of sources. All quantities are time dependant, depend on calibration methods as well as on appropriate physical models that selects energy models for the origin of the recorded photons ...

**TODO:**
to be developed, explained ...

- name
- pos
- time
- extension
- PHA ???

[2] if you want to have a look at the plain content, there's a TAP service at http://vlkb.dev.neanias.eu:8080/vlkb/tap (http://saada.unistra.fr/taphandle?url=http%3A//vlkb.dev.neanias.eu%3A8080/vlkb/tap/) that can be accessed, even if not complete with descriptions nor registered

### 2.1.6 Vizier catalog archive

VizieR provides science ready catalogues coming from space agencies or articles and covering number of different science cases. Published data encompass a very large set of measures (position, photometry, redshift, source type, etc.) depending on their origin. They can result from observations, simulations, models or catalog compilations. Individual Vizier tables can contain data all related to one source (e.g. time series of positions or magnitudes) or to a set of sources (one row per source) or a mix of both. Data sets are ingested in Vizier on author request. Before to be put online they are processed and documented by documentalists so as to ensure a certain level of interoperability. This work relies on the analyse of both data content and scientific paper.

- Missing meta-data, e.g. space frames or filters, are added when available in the paper.
- Columns are renamed following the Vizier nomenclature in order make them compliant with the DBMS and to facilitate the grouping of all values related to one particular quantities (e.g. quality flag for a radial velocity)
- UCDs are checked or set for all columns.
- README files are generated. A README is a text file with a specific layout making it machine readable.
- Some values, not part of the original data but assigned by the CDS, are added to the tables (e.g. identifier, ICRS positions)
- Ancillary data pointing on associated data (e.g. spectra) or on linked services (e.g. visualisation), can also be added to enrich the table content.

All Vizier meta data are gathered in a specific resource in a way to facilitate the localisation of data of interest. The main specificity of the data is their heterogeneity

- Huge variety of data provenance and processing
- Meta data heterogeneity
- Table content heterogeneity
- Huge variety of possible measures
- Huge variety of patterns of measure groups
- Use of different coordinate systems

- Large variety of associated data

The Mango model must be able to provide a standard representation of most of the metadata contained in Vizier query responses, whether native or computed by the CDS, simple quantities or associated complex data. Mango is not meant to replace the current management of the meta-data, it is a way to make those meta-data understandable for a wide panel of VO-compliant clients.

Vizier gathers and delivers a curated version of published catalogs from various missions and experiments. It also distributes results of scientific papers, based on the computation , comparison and classification of sources extracted from archived data after science analysis. Vizier handles a very large set of measures in position, photometry, redshift, source type, etc. It adds value to it by recomputing additional quantities in various reference frames or equivalent spectral bands, units conversions , etc . It binds the resulting object description to other data sets representing the object, or its counterparts, or neighbourhood on sky (image), its spectral behaviour (spectrum, spectral energy distribution) or evolution through time (light curve, radial velocity curve, time series, etc.). Currently the binding and structure of the quantities is done by column grouping.

- pre-existing data
- grouping columns
- lots of available metadata
- column name formatting
- one column different frames

### 2.1.7 Client on (MT behalf)

Right now, the meta-data provided within the VOTable allow clients such Aladin or Topcat to run most of the functionalities expected by the user, either for data analysis of plotting. This information is often guess from UCDs, UTypes or columns name. It can also be given by the user. Clients have no expectations of working with full model instances but in some cases models can help to know how quantities in an input table relate to each other. In most cases this is for visualisation, e.g.:

- what is the sky position for this row (what columns contain latitude and longitude, and what sky system are they in)
- what +/-ERR error bars should I plot for these points (what column is a simple error for column A)

- what error ellipses should I plot for these sky positions (what columns provide ra_error, dec_error, ra_dec_corr, or how can I derive those from columns that do exist)
- where do I get the grid information for a column containing a vector of samples so I can label the X axis of a spectrogram (what column or parameter contains an axis vector matching the sample vectors)
- does this table contain sky positions, or HEALPix tiles, or both? What's the best way to represent it on the sky?
- What is the meaning of such URL found out in a table?s

But there are some other places too:

- how do I propagate this sky position to a future epoch (what columns contain pmra, pmdec, and maybe all the associated errors and correlation coefficients)
- what is the error ellipse/oid to use for a sky/Cartesian crossmatch (what columns provide the relevant errors and, if available, correlations)

This usage shows that MANGO must be designed in a way that individual measurements or quantities can be easily be identified as such and manipulated independently of the whole instance.

### 2.1.8 Xmatch tool

The basic cross-match of two astronomical tables consists in associating pairs of sources – one from each table – fulfilling a given angular distance based criterion. In relational algebra terms, it is a theta-join on a distance predicate.

More generally, a cross-match is the association of sources from different tables given their proximity in an astrometric (but also possibly photometric, statistical, ...) parameter space (?) .

If proper motions (plus parallax and radial velocities) are available, the cross-match tool may propagate the positions of each table to a common epoch. It may also take into account positional uncertainties to reject the statistically unlikely associations.

In the latter case (cross-match between two tables taking into account positional errors), the tool needs to be able to retrieve the errors associated to the each position in each table.

UCDs may help in identifying the errors associated to a positional columns as shown in table 1.

But this is not sufficient to table with more complex cases based on multi-parameter cases:

| **Error type<br>Parameters** | **UCD** | **Description** |
|---|---|---|
| **Circular error** | | |
| epos | stat.error;pos.eq | See "possible ambiguity for circular errors" |
| **Uncorrelated errors** | | |
| eRA | stat.error;pos.eq.ra | Error on ‘RA cos(Dec)‘ |
| eDec | stat.error;pos.eq.dec | Error on ‘Dec‘ |
| **Correlated errors** | | |
| eRa | stat.error;pos.eq.ra | Error on ‘RA cos(Dec)‘ |
| eDec | stat.error;pos.eq.dec | Error on ‘Dec‘ |
| corRADec | stat.covariance;<br>pos.eq.ra; pos.eq.dec | Correlation factor |
| **Oriented Ellipse** | | |
| a | phys.angSize.smajAxis;<br>pos.errorEllipse | Error ellipse semi-major axis |
| b | phys.angSize.sminAxis;<br>pos.errorEllipse | Error ellipse semi-minor axis |
| theta | pos.posAng;<br>pos.errorEllipse | Error ellipse position angle |

*Table 1:* Table of the different possible representations of positional errors that can be found in astronomical catalogues

- Catalogues like AllWISE provides a co-sigma instead of the correlation factor of the covariance matrix. Co-sigma is the sign of the correlation factor time the square root of the covariance.

- Table fields UCDs may be too loose: for example `stat.error;pos.eq` is often used in place of `phys.angSize.smajAxis;pos.errorEllipse` or `phys.angSize.sminAxis;pos.errorEllipse`.

- The location of the column to be used for the Xmatch can be ambiguous. For instance, if several pairs of position are provided in a table, there is currently no way to associate unambiguously uncertainties with the (right) pair of coordinates.

- Possible ambiguity for circular errors. When the provided uncertainty is the parameter of a circular error, it may be:
  - either the 1 dimensional component on each axis of a symmetric 2-dimensional Gaussian distribution: $\sigma = \sigma_{\alpha\cos\delta} = \sigma_\delta$;
  - or the parameter of the radial error distribution (i.e. of the Rayleigh distribution): $\sigma = \sqrt{\sigma^2_{\alpha\cos\delta} + \sigma^2_\delta}$ with $\sigma_{\alpha\cos\delta} = \sigma_\delta$

- Possible ambiguity on the confidence level The provided error is usually the 1sigma error. It (theoretically) means that the "true" position has:
  - either 68% chances to be at a distance lower than the radial error from the position's mean value.
  - or 39% chances to be inside the error ellipse (or circle) around the position's mean value.

  But depending on the catalogue, the provided error parameters can correspond to different confidence levels.

## 2.2 Requirements

### 2.2.1 Parameters and Associated Data

From the use-cases' description, two categories of features must provided or foreseen by the projects:

- The source *parameters* astronomers will investigate for their science. They are measures provided as numerical values or classification tags exposed as numbers or simple strings. Usually on measure corresponds to one individual column or one group of columns .

- The *Associated data* are generally science ready data-products either from the same project, or shared by other projects within the IVOA interoperability framework. They bring a complement of information to interpret the source's parameters under study and compare visually (or computationally) the sky neighbourhood of detected sources, their variation through time or spectral behaviour.

  Referencing such datasets by a URI or by a service endpoint already works for existing VO products. ( It has been promoted and recommended in Obscore / Spectrum Dm, . etc for access URIs, and DataLink for service endpoints REFS). Associated data can also present a complex structure designed to bring a very advanced context of interpretation (CTA model assumptions, energy model for Xray sources, etc. REFS) which need to be described in the attachement.

### 2.2.2 R01: Supported Quantities

- MANGO must provide unique source identifiers.

- MANGO must provide modelling classes for both parameters and associated data.

- The number of parameters attached to a MANGO instance must be free.

- The number of associated data attached to a MANGO instance must be free.

### 2.2.3 R02: Parameters

The concept of `Parameter` matches the concept of measure of the Meas model. MANGO may support Parameter classes that are not Meas classes though.

- MANGO must support explicit classes imported from an IVOA datamodel for the most used parameters.
- MANGO must provide a generic way to support parameters that do not enter the above category.
- MANGO instances must support multiple instances of the same parameter class.
- The presence of any parameter in MANGO instances must be optional.
- MANGO must provide a way to identify the role of each parameter.
- MANGO must provide a way to describe the meaning of flags or qualifiers.
- The role of each parameter should be machine-readable.
- It must be possible to group parameters in a free way. This allows to tag quantities with timestamps or a flags.

### 2.2.4 R03: Associated Data

The notion of associated data relates to any sort of complex data. This can be a pointer to a service or a data set, a data table or other data structure.

- MANGO must support references to external datasets.
- MANGO must support references to external services.
- MANGO must support references to other MANGO instances.
- MANGO must support references to instances of models serialised in VO-DML.
- MANGO instances must support multiple instances of the same associated data class.
- The presence of any associated data in MANGO instances must be optional.

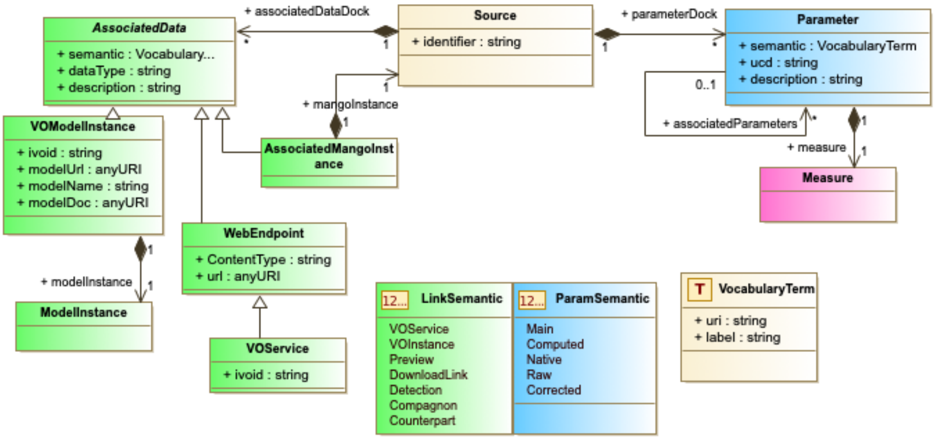


*Figure 2:* MANGO overview

- MANGO must provide a way to identify the role of each associated data in order to explain which purpose is served when associating this data to the source object.
- The role of each associated data should be machine-readable.

## 3 Model Overview

Sky objects are represented by instances of the class `MangoObject` which has only one mandatory attribute, the source `identifier`. It is recommended the identifiers to be unique within a source collection e.g. a catalog.

The `MangoObject` is a dock for all source parameters and for all data associated to that source. The pattern of either parameters or associated data attached to a source is not specified by the model. It depends of the data set on which the model is applied.

The `MangoObject` has one connector for the parameters, the `parameters` relation, and another for the associatedData, the `associatedData` relations. Both connectors have an open-ended cardinality.

Each source parameter is hooked to the `MangoObject` by a wrapping class (`Parameter` ) that contains anything necessary to identify both its nature and its role.

Parameters can be linked together to represent logical parameter sets. These logical sets have no semantic. Their interpretation is in charge of the clients.

Each associated data or data pointer is hooked to the `MangoObject` by a wrapping class (`AssociatedData`) that contains anything necessary to identify its nature and its role.

## 3.1 Parameters

`Parameter` connectors are used to bind measures with the source. A parameters is composed by semantic tags in addition to the measure itself, instance of `Measure`.

`Measure` is an abstract class imported from `meas` model. Concrete classes referenced by a `Parameter` instances are either `meas` built-in classes or `Measure` sub-classes being part of Mango.

### 3.1.1 Parameters Identification

As the parameter set attached to a particular instance is not defined by Mango, the model must provide an accurate parameter description to allow the client to figure out what it can do with each of them. Mango provides 5 description levels for each parameter:

- **Measure class (vodml type)**: measures are modelled by specific `Measure` sub-classes. Knowing that class tells the clients how to interpret the corresponding measure but does not help much to get its role e.g. a position can be either a source position or a pointing direction. Furthermore, unusual measures e.g. magnetic field, are represented by `GenericMeasure` . In that case, the vodml type does not help at all.

- **UCD**: A valid UCD must be attached to each parameter. Mango provides a UCD space for each `Measure` sub-class . UCDs used for specific measures must be compliant with table 2. For generic measures, the UCD choice is in charge of the data provider. In any cases, the consistance between UCDs and measure is the responsibility of the data provider.

- **Reduction status (model enumeration)**: A reduction level of the parameter (e.g. a parameter can be calibrated or not or it can be a computed qualifier) may be attached to the parameter.

- **Semantic**: A reference to a valid vocabulary word may be attached to each parameter. The choice of that vocabulary is totally free as long as it is published.

- **Description**: A free text description may be attached to each parameter.

**TODO:**
TBC phys.luminosity vs phot.flux

| Parameter | Original model | UCDs 1+ first word |
|---|---|---|
| `GenericMeasure` | Measure | Appropriate UCD |
| `Position` | Measure | pos.* |
| `Velocity` | Measure | phys.veloc |
| `Proper motion` | Measure | pos.pm |
| `Time` | Measure | time.epoch |
| `Polarization` | Measure | phys.polarization |
| `LonLatSkyPosition` | MANGO | pos.eq |
| `Redshift` | MANGO | src.redshift |
| `Photometry` | MANGO | phot.* |
| `HardnessRatio` | MANGO | phot.flux;arith.ratio |
| `Shape` | MANGO | phys.area |
| `Flag` | MANGO | meta.code |
| `Orbit` | MANGO | src.orbital |
| `GenericStringMeasure` | MANGO | Appropriate UCD |
| `GenericVocabMeasure` | MANGO | Appropriate UCD |

*Table 2:* UCDs space to be used for the supported measures

### 3.1.2 Measure Extension

All `Measure` classes are built upon the MCT pattern (see Fig 3).

- `Measure` instances are made with an `Error` and a `Coordinate` that contains the measure value(s).
- The `Coordinate` includes a `CoordSys` instance describing the coordinate system relevant for that measure.
- The coordinate system has two components : the space ( `CoordSpace` class) that describes the axis and the frame (`CoordFrame` class).

MANGO parameters are all based on this pattern. Native `Measure` classes are used whenever possible. Others Parameters are built by extending this class as shown in figure 3).

The scope of the MCT `Measure` concept has been extended to quantities that are not physical measurements such as eg. flags, object types or orbits. This lexical deviation allows us to place all quantities describing a source in a single container since all measurements in the Mango sense are derived from the MCT `Measure` class. The same approach has been used for the coordinate systems since e.g the way to serialise a complex shape (e.g. as a MOC) is considered as a coordinate system in sense of `Coords`.

These extended classes are part of MANGO.

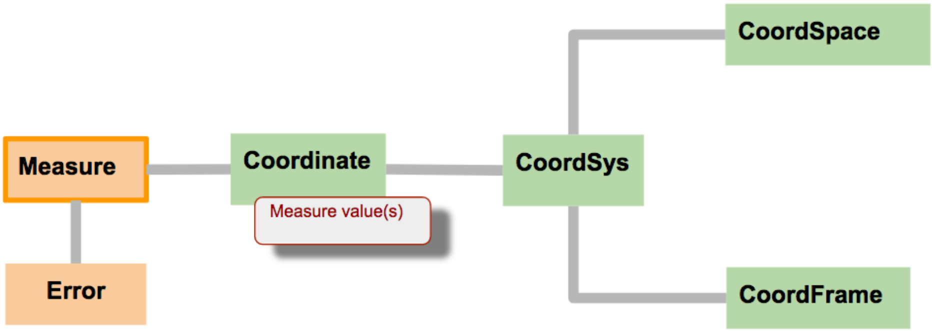


*Figure 3:* Measure/Coordinate pattern (simplified view)

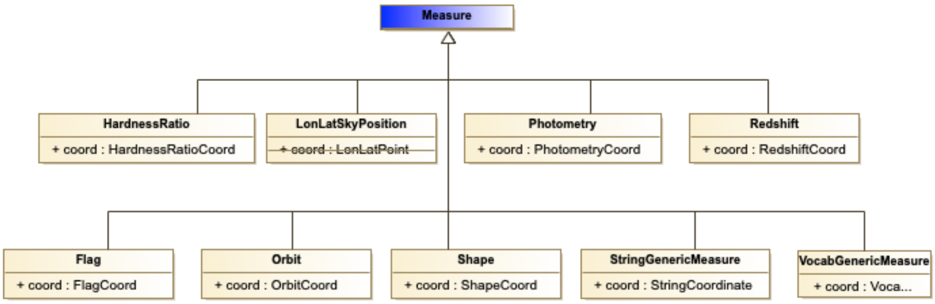


*Figure 4:* Mango extensions of Measure.

## 3.2 Associated Data

`AssociatedData` connectors are used to bind any sort of complex data with the source. One connector can only refer to one dataset. Associated data can be either URIs (VO services or not), other mango instances or reference to instances of other VO models (e.g. Obscore, Provenance…).

### 3.2.1 AssociatedData Identification

As the associated data set attached to a particular instance is not defined by Mango, the model must provide an accurate parameter description to allow the client to figure out what it can do with each of them. Mango provides 3 description levels for each category of associated data:

- **Data class (vodml type)**: Knowing the class representing associated data tells the clients how to interpret it but does help to get its role.
- **Semantic**: A reference to a valid vocabulary word may be attached

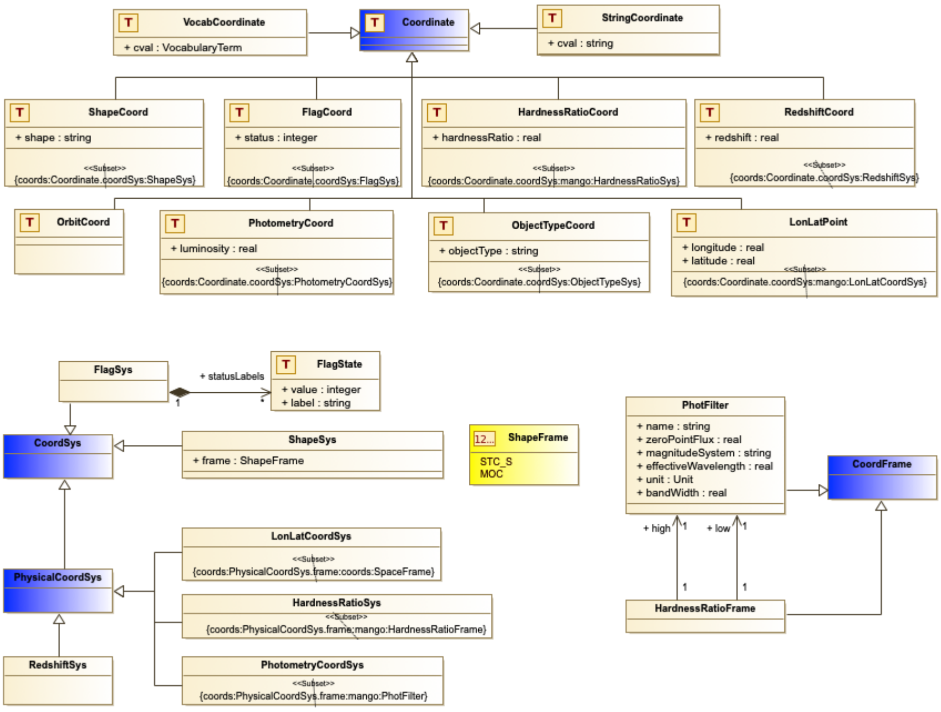


*Figure 5:* Mango extensions of Coordinates.

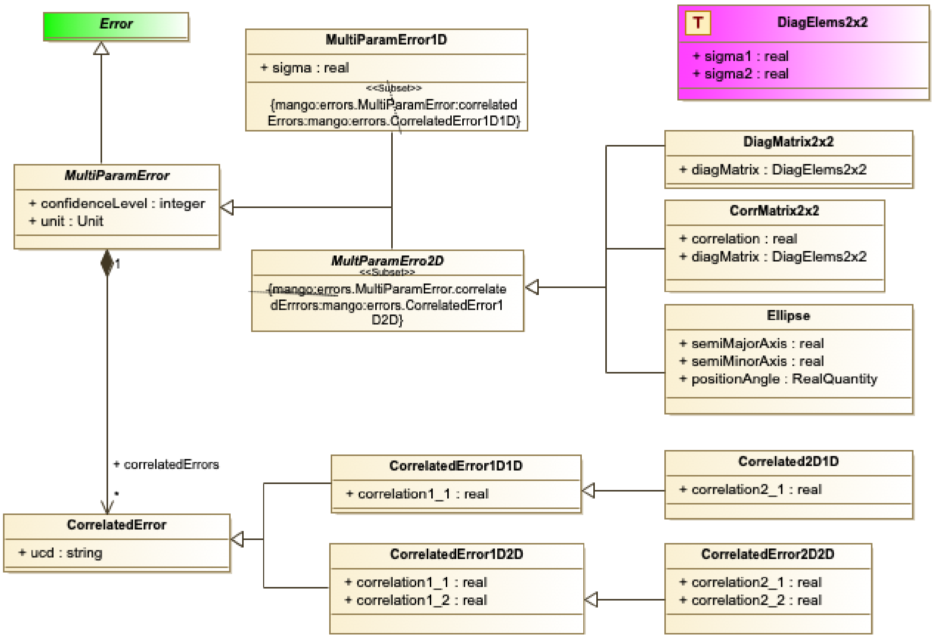


*Figure 6:* Mango correlated errors.

to each associated data set. The choice of that vocabulary is totally free as long as it is published.

- **Description**: A free text description may be attached to each associated data .

### 3.2.2 Associated Model Instances

The way to attach VO model instances to MANGO sources is very specific in a sense of that Mango gives the model reference by tells nothing about the way import the instance. The model of associated instances must be available as a vodml file. The `ModelInstance` is an empty class without sub classes.

# 4 Model: mango

Data model based oon components and data association for source data

## 4.1 AssociatedData (Abstract)

Abstract reference to a particular dataset associated to the Source. This class is used to specify the type of the dataset as well as its role.

### 4.1.1 AssociatedData.semantic

**vodml-id: AssociatedData.semantic**
**type: mango:VocabularyTerm**
**multiplicity: 1**
Reference to a semantic concept giving the nature of the associated data. As long as the vocabulary is not set, the possible values of this attribute are given by the LinkSemantic enumeration.

### 4.1.2 AssociatedData.dataType

**vodml-id: AssociatedData.dataType**
**type: ivoa:string**
**multiplicity: 1**
Type of the associated data (not defined yet)

### 4.1.3 AssociatedData.description

**vodml-id: AssociatedData.description**
**type: ivoa:string**
**multiplicity: 1**
Free text description of the associated data

## 4.2 AssociatedMangoInstance

Reference to another MANGO instance that is part of the associated data.

### 4.2.1 AssociatedMangoInstance.mangoInstance

**vodml-id: AssociatedMangoInstance.mangoInstance**
**type: mango:Source**
**multiplicity: 1**
Composition link pointing on one cab_msd instance associated with the source.

## 4.3 ModelInstance

Placeholder for the mapping of the model instance

## 4.4 Parameter

Reference to a particular measure of the Source. This class is used to specify the type of the measure as well as its role.
**constraint**
**detail: Parameter.One association at the time**

### 4.4.1 Parameter.semantic

**vodml-id: Parameter.semantic**
**type: mango:VocabularyTerm**
**multiplicity: 1**
Reference to a semantic concept giving the nature of the parameter As long as the vocabulary is not set, the possible values of this attribute are given by the ParamSemantic enumeration.

### 4.4.2 Parameter.ucd

**vodml-id: Parameter.ucd**
**type: ivoa:string**
**multiplicity: 1**
UCD1+ giving the type of the physical measure

### 4.4.3 Parameter.description

**vodml-id: Parameter.description**
**type: ivoa:string**
**multiplicity: 1**
Free text description of the measure

### 4.4.4 Parameter.measure

**vodml-id: Parameter.measure**
**type: meas:Measure**
**multiplicity: 1**
Composition link pointing to the meas:Measure instance

### 4.4.5 Parameter.associatedParameters

**vodml-id: Parameter.associatedParameters**
**type: mango:Parameter**
**multiplicity: 0..***
TODO : Missing description : please, update your UML model asap.

## 4.5 Source

Root class of the model. MANGO instance are meant ot be Source instances. A source ha an identifier and to sets of hooks: one for the parameters and one for the associated data.

### 4.5.1 Source.identifier

**vodml-id: Source.identifier**
**type: ivoa:string**
**multiplicity: 1**
Unique identifier for a Source. The uniqueness of that identifier is not managed by the model. The format is free.

### 4.5.2 Source.associatedDataDock

**vodml-id: Source.associatedDataDock**
**type: mango:AssociatedData**
**multiplicity: 0..***
Composition link pointing on all data associated with the source.

### 4.5.3 Source.parameterDock

**vodml-id: Source.parameterDock**
**type: mango:Parameter**
**multiplicity: 0..***
Composition link pointing on all parameters attached to the source.

## 4.6 VOModelInstance

Reference to a VO model instance that is part of the associated data.

### 4.6.1 VOModelInstance.ivoid

**vodml-id: VOModelInstance.ivoid**
**type: ivoa:string**
**multiplicity: 1**
VO-DML id of the referenced model

### 4.6.2 VOModelInstance.modelUrl

**vodml-id: VOModelInstance.modelUrl**
**type: ivoa:anyURI**
**multiplicity: 1**
URL on the VO-DML model

### 4.6.3 VOModelInstance.modelName

**vodml-id: VOModelInstance.modelName**
**type: ivoa:string**
**multiplicity: 1**
Name of the referenced model

### 4.6.4 VOModelInstance.modelDoc

**vodml-id: VOModelInstance.modelDoc**
**type: ivoa:anyURI**
**multiplicity: 1**
Documentation URL of the model

### 4.6.5 VOModelInstance.modelInstance

**vodml-id: VOModelInstance.modelInstance**
**type: mango:ModelInstance**
**multiplicity: 1**
Composition link pointing on one VO instance instance associated with the source.

## 4.7 VOService

Class for associated data referenced by a fixed URL that is a VO service.

### 4.7.1 VOService.ivoid

**vodml-id: VOService.ivoid**
**type: ivoa:string**
**multiplicity: 1**
IVOA id of the service (for example in the registry)

## 4.8 VocabularyTerm

Datatype for vocabulary word. Provides a pointer to the word description and a label.

### 4.8.1 VocabularyTerm.uri

**vodml-id: VocabularyTerm.uri**
**type: ivoa:string**
**multiplicity: 1**
URI extracted from the DRF document and referring to the word

### 4.8.2 VocabularyTerm.label

**vodml-id: VocabularyTerm.label**
**type: ivoa:string**
**multiplicity: 1**
RDF label. Matched the URL fragment for IVOA vocabularies

## 4.9 WebEndpoint

Class for associated data referenced by an URL

### 4.9.1 WebEndpoint.ContentType

**vodml-id: WebEndpoint.ContentType**
**type: ivoa:string**
**multiplicity: 1**
Mime type of the URL

### 4.9.2 WebEndpoint.url

**vodml-id: WebEndpoint.url**
**type: ivoa:anyURI**
**multiplicity: 1**
Web endpoint

## 4.10 LinkSemantic

Literal enumeration of the possible values for the associated data semantic. This stands for an example before we have defined a vocabulary.
Enumeration Literals

**VOService** : **vodml-id:** LinkSemantic.VOService
**description:** Data returned by a VO service

**VOInstance** : **vodml-id:** LinkSemantic.VOInstance
**description:** Data Serialized in a VO model

**Preview** : **vodml-id:** LinkSemantic.Preview
**description:** data preview

**DownloadLink** : **vodml-id:** LinkSemantic.DownloadLink
**description:** Data download link

**Detection** : **vodml-id:** LinkSemantic.Detection
**description:** Particular detection

**Compagnon** : **vodml-id:** LinkSemantic.Compagnon
**description:** Compagnon source

**Counterpart** : **vodml-id:** LinkSemantic.Counterpart
**description:** Counter part source

### 4.11 ParamSemantic

Literal enumeration of the possible values for the parameter semantic. This stands for an example before we have defined a vocabulary.
Enumeration Literals

**Main** : **vodml-id:** ParamSemantic.Main
**description:** Main measurment

**Computed** : **vodml-id:** ParamSemantic.Computed
**description:** Computed measurement

**Native** : **vodml-id:** ParamSemantic.Native
**description:** Mative measurement

**Raw** : **vodml-id:** ParamSemantic.Raw
**description:** raw measure

**Corrected** : **vodml-id:** ParamSemantic.Corrected
**description:** Corrected measure

### 4.12 ShapeFrame

TODO : Missing description : please, update your UML model asap.
Enumeration Literals

**STC_S** : **vodml-id:** ShapeFrame.STC_S
**description:** TODO : Missing description : please, update your UML model asap.

**MOC** : **vodml-id:** ShapeFrame.MOC
**description:** TODO : Missing description : please, update your UML model asap.

# 5 Package: errors

This package contains all `meas:Error` class extensions

## 5.1 CorrMatrix2x2

Variance matrix with correlation between errors on individual axes.

### 5.1.1 CorrMatrix2x2.correlation

**vodml-id: errors.CorrMatrix2x2.correlation**
**type: ivoa:real**
**multiplicity: 1**
Correlation between the errors on the 2 axes. The covariance is given by $cov_{XY} = corr_{XY}\sigma_X\sigma_Y$

### 5.1.2 CorrMatrix2x2.diagMatrix

**vodml-id: errors.CorrMatrix2x2.diagMatrix**
**type: mango:errors.DiagElems2x2**
**multiplicity: 1**
Diagonal elements of the matrix. The unit is given by `mango:error.MultiParamError.unit`

## 5.2 Correlated2D1D

Correlation coefficients between the error of a 1D host parameter and a 2D associated parameters.

### 5.2.1 Correlated2D1D.correlation2_1

**vodml-id: errors.Correlated2D1D.correlation2_1**
**type: ivoa:real**
**multiplicity: 1**
Correlation between the error on the first axis of the host parameter and the error on the second axis of the associated parameter. The covariance is given by $cov_{XY} = corr_{XY}\sigma_X\sigma_Y$

## 5.3 CorrelatedError

Correlation coefficients between the error of the host parameter and one of its associated parameters. The host parameter is one of the `mango:Parameter` of the `mango:ParameterDock` (a position typically) of the Mango object. The associated parameter is one of the `mango:Parameter.associatedParameters` of that parameter (typically a proper motion) There is no logical link between the correlated error instance and the associated parameter it refers to.

The associated parameter is identified by the `UCD field`. The client is in charge of solving this dependency.

### 5.3.1 CorrelatedError.ucd

**vodml-id: errors.CorrelatedError.ucd**
**type: ivoa:string**
**multiplicity: 1**
UCD of the associated parameter. This UCD must be identical to this of the associated parameter the `CorrelatedError` refers to.

## 5.4 CorrelatedError1D1D

Correlation coefficients between the error of one 1D host parameter and a one 1D associated parameters.

### 5.4.1 CorrelatedError1D1D.correlation1_1

**vodml-id: errors.CorrelatedError1D1D.correlation1_1**
**type: ivoa:real**
**multiplicity: 1**
Correlation between the error on the first axis of the host parameter and the error on the first axis of the associated parameter. The covariance is given by $cov_{XY} = corr_{XY}\sigma_X\sigma_Y$

## 5.5 CorrelatedError1D2D

Correlation coefficients between the error of one 2D host parameter and one 1D associated parameters.

### 5.5.1 CorrelatedError1D2D.correlation1_1

**vodml-id: errors.CorrelatedError1D2D.correlation1_1**
**type: ivoa:real**
**multiplicity: 1**
Correlation between the error on the first axis of the host parameter and the error on the associated parameter. The covariance is given by $cov_{XY} = corr_{XY}\sigma_X\sigma_Y$

### 5.5.2 CorrelatedError1D2D.correlation1_2

**vodml-id: errors.CorrelatedError1D2D.correlation1_2**
**type: ivoa:real**
**multiplicity: 1**
Correlation between the error on the second axis of the host parameter and

the error on the associated parameter. The covariance is given by $cov_{XY} = corr_{XY}\sigma_X\sigma_Y$

## 5.6 CorrelatedError2D2D

Correlation coefficients between the error of a 2D host parameter and a 2D associated parameters.

### 5.6.1 CorrelatedError2D2D.correlation2_1

**vodml-id: errors.CorrelatedError2D2D.correlation2_1**
**type: ivoa:real**
**multiplicity: 1**
Correlation between the error on the first axis of the host parameter and the error on the second axis of the associated parameter. The covariance is given by $cov_{XY} = corr_{XY}\sigma_X\sigma_Y$

### 5.6.2 CorrelatedError2D2D.correlation2_2

**vodml-id: errors.CorrelatedError2D2D.correlation2_2**
**type: ivoa:real**
**multiplicity: 1**
Correlation between the error on the second axis of the host parameter and the error on the second axis of the associated parameter. The covariance is given by $cov_{XY} = corr_{XY}\sigma_X\sigma_Y$

## 5.7 DiagMatrix2x2

Simple diagonal matrix of variances

### 5.7.1 DiagMatrix2x2.diagMatrix

**vodml-id: errors.DiagMatrix2x2.diagMatrix**
**type: mango:errors.DiagElems2x2**
**multiplicity: 1**
Diagonal elements of the matrix. The unit is given by `mango:errors.MultiParamError.unit`

## 5.8 Ellipse

Elliptical error. The regular ellipse orientation is East of the North

### 5.8.1 Ellipse.semiMajorAxis

**vodml-id: errors.Ellipse.semiMajorAxis**
**type: ivoa:real**
**multiplicity: 1**
Semi major axis of the ellipse. The unit is given by `mango:errors.MultiParamError.unit`

### 5.8.2 Ellipse.semiMinorAxis

**vodml-id: errors.Ellipse.semiMinorAxis**
**type: ivoa:real**
**multiplicity: 1**
Semi minor axis of the ellipse. The unit is given by `mango:errors.MultiParamError.unit`

### 5.8.3 Ellipse.positionAngle

**vodml-id: errors.Ellipse.positionAngle**
**type: ivoa:RealQuantity**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 5.9 MultParamErro2D (Abstract)

This class models errors on a one 2 axes parameter with possible correlations with errors on associated parameters. A classical use-case is an error on a position that is coupled with errors on the proper motion and/or the parralax.

**subset**
**role: mango:errors.MultiParamError.correlatedErrrors**
**type: mango:errors.CorrelatedError1D2D**

## 5.10 MultiParamError (Abstract)

This class models errors with possible correlations between different axes and with errors of associated parameters. The standard use-case for such errors is a positional error where e.g. errors on position, proper motion and parallax are correlated to each other.

### 5.10.1 MultiParamError.confidenceLevel

**vodml-id: errors.MultiParamError.confidenceLevel**
**type: ivoa:integer**
**multiplicity: 1**
Error confidence level, expressed in $\sigma$.

### 5.10.2 MultiParamError.unit

**vodml-id: errors.MultiParamError.unit**
**type: ivoa:Unit**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

### 5.10.3 MultiParamError.correlatedErrors

**vodml-id: errors.MultiParamError.correlatedErrors**
**type: mango:errors.CorrelatedError**
**multiplicity: 0..***
TODO : Missing description : please, update your UML model asap.

## 5.11 MultiParamError1D

This class models errors on a one axis parameter with possible correlations with errors on associated parameters. A classical use-case is an error on a radial velocity that is coupled with an error on the proper motion.
**subset**
**role: mango:errors.MultiParamError:correlatedErrors**
**type: mango:errors.CorrelatedError1D1D**

### 5.11.1 MultiParamError1D.sigma

**vodml-id: errors.MultiParamError1D.sigma**
**type: ivoa:real**
**multiplicity: 1**
Variance of the parameter error. The unit is given by `mango:errors.MultiParamError.unit`

## 5.12 DiagElems2x2

Datatype containing the 2 diagonal elements of a 2x2 matrix. Attributes are named $\sigma$ because this datatype is mostly used in the context of complex errors.

# 6 Package: measures

This package contains all `meas` class extensions

## 6.1 Flag

Measure to be used for status parameters

### 6.1.1 Flag.coord

**vodml-id: measures.Flag.coord**
**type: mango:coordinates.FlagCoord**
**multiplicity: 1**
Coordinate holding the status value

## 6.2 HardnessRatio

TODO : Missing description : please, update your UML model asap.

### 6.2.1 HardnessRatio.coord

**vodml-id: measures.HardnessRatio.coord**
**type: mango:coordinates.HardnessRatioCoord**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 6.3 LonLatSkyPosition

Measure to used for sky points expressed with a spherical coordinate system

### 6.3.1 LonLatSkyPosition.coord

**vodml-id: measures.LonLatSkyPosition.coord**
**type: mango:coordinates.LonLatPoint**
**multiplicity: 1**
Coordinate of spherical sky position

## 6.4 Orbit

TODO : Missing description : please, update your UML model asap.

### 6.4.1 Orbit.coord

**vodml-id: measures.Orbit.coord**
**type: mango:coordinates.OrbitCoord**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 6.5 Photometry

TODO : Missing description : please, update your UML model asap.

### 6.5.1 Photometry.coord

**vodml-id: measures.Photometry.coord**
**type: mango:coordinates.PhotometryCoord**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 6.6 Redshift

TODO : Missing description : please, update your UML model asap.

### 6.6.1 Redshift.coord

**vodml-id: measures.Redshift.coord**
**type: mango:coordinates.RedshiftCoord**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 6.7 Shape

Measure giving the shape of a source. The shape is a string with a specific frame (STS_S or MOC)

### 6.7.1 Shape.coord

**vodml-id: measures.Shape.coord**
**type: mango:coordinates.ShapeCoord**
**multiplicity: 1**
String serialization of the source shape

## 6.8 StringGenericMeasure

Generic measure which value is a string

### 6.8.1 StringGenericMeasure.coord

**vodml-id: measures.StringGenericMeasure.coord**
**type: mango:coordinates.StringCoordinate**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 6.9 VocabGenericMeasure

TODO : Missing description : please, update your UML model asap.

### 6.9.1 VocabGenericMeasure.coord

**vodml-id: measures.VocabGenericMeasure.coord**
**type: mango:coordinates.VocabCoordinate**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

# 7 Package: coordinates

This package contains all `coords` class extensions

## 7.1 FlagCoord

Coordinate of a status Measure
**subset**
    **role: coords:Coordinate.coordSys**
    **type: FlagSys**

### 7.1.1 FlagCoord.status

**vodml-id: coordinates.FlagCoord.status**
**type: ivoa:integer**
**multiplicity: 1**
Value of the status

## 7.2 FlagState

Possible value of a status

### 7.2.1 FlagState.value

**vodml-id: coordinates.FlagState.value**
**type: ivoa:integer**
**multiplicity: 1**
Status value

### 7.2.2 FlagState.label

**vodml-id: coordinates.FlagState.label**
**type: ivoa:string**
**multiplicity: 1**
Label attached to that status value

## 7.3 FlagSys

Coordinate system to be used for statur measures.

### 7.3.1 FlagSys.statusLabels

**vodml-id: coordinates.FlagSys.statusLabels**
**type: mango:coordinates.FlagState**
**multiplicity: 0..***
Composition loink to all possible status values for this system

## 7.4 HardnessRatioCoord

TODO : Missing description : please, update your UML model asap.

**subset**

**role: coords:Coordinate.coordSys**
**type: mango:HardnessRatioSys**

### 7.4.1 HardnessRatioCoord.hardnessRatio

**vodml-id: coordinates.HardnessRatioCoord.hardnessRatio**
**type: ivoa:real**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 7.5 HardnessRatioFrame

Hardness ratio frame. Defined by 2 energy bands Eheigh ELow. HR = (Eheigh - Elow)/(Eheigh + Elow) Energy bands are deemed to special photometric filters

### 7.5.1 HardnessRatioFrame.low

**vodml-id: coordinates.HardnessRatioFrame.low**
**type: mango:coordinates.PhotFilter**
**multiplicity: 1**
Low energy band

### 7.5.2 HardnessRatioFrame.high

**vodml-id: coordinates.HardnessRatioFrame.high**
**type: mango:coordinates.PhotFilter**
**multiplicity: 1**
Heigh energy band

## 7.6 HardnessRatioSys

TODO : Missing description : please, update your UML model asap.
**subset**
  **role: coords:PhysicalCoordSys.frame**
  **type: mango:HardnessRatioFrame**

## 7.7 LonLatCoordSys

TODO : Missing description : please, update your UML model asap.
**subset**
  **role: coords:PhysicalCoordSys.frame**
  **type: coords:SpaceFrame**

**constraint**
  **detail: LonLatCoordSys.coordSpace[0]**

## 7.8 LonLatPoint

Coordinate of a point on the sky sphere expressed in spherical coordinates.
**subset**
  **role: coords:Coordinate.coordSys**
  **type: mango:LonLatCoordSys**

### 7.8.1 LonLatPoint.longitude

**vodml-id: coordinates.LonLatPoint.longitude**
**type: ivoa:real**
**multiplicity: 1**
longitude of the point

### 7.8.2 LonLatPoint.latitude

**vodml-id: coordinates.LonLatPoint.latitude**
**type: ivoa:real**
**multiplicity: 1**
Latitude of the point

## 7.9 ObjectTypeCoord

TODO : Missing description : please, update your UML model asap.
**subset**
**role: coords:Coordinate.coordSys**
**type: ObjectTypeSys**

### 7.9.1 ObjectTypeCoord.objectType

**vodml-id: coordinates.ObjectTypeCoord.objectType**
**type: ivoa:string**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 7.10 OrbitCoord

TODO : Missing description : please, update your UML model asap.
**subset**
**role: coords:Coordinate.coordSys**
**type: coords:SpaceSys**

## 7.11 PhotFilter

Photometric filter description, compliant with photDM

### 7.11.1 PhotFilter.name

**vodml-id: coordinates.PhotFilter.name**
**type: ivoa:string**
**multiplicity: 1**
Filter name

### 7.11.2 PhotFilter.zeroPointFlux

**vodml-id: coordinates.PhotFilter.zeroPointFlux**
**type: ivoa:real**
**multiplicity: 1**
Zero point flux of the filter

### 7.11.3 PhotFilter.magnitudeSystem

**vodml-id: coordinates.PhotFilter.magnitudeSystem**
**type: ivoa:string**
**multiplicity: 1**
Magnitude system used by the filter

### 7.11.4 PhotFilter.effectiveWavelength

**vodml-id: coordinates.PhotFilter.effectiveWavelength**
**type: ivoa:real**
**multiplicity: 1**
Effective wavelength of the filter

### 7.11.5 PhotFilter.unit

**vodml-id: coordinates.PhotFilter.unit**
**type: ivoa:Unit**
**multiplicity: 1**
Wavelength unit used for that filter

### 7.11.6 PhotFilter.bandWidth

**vodml-id: coordinates.PhotFilter.bandWidth**
**type: ivoa:real**
**multiplicity: 1**
Band width of the filter

## 7.12 PhotometryCoord

TODO : Missing description : please, update your UML model asap.
**subset**
  **role: coords:Coordinate.coordSys**
  **type: PhotometryCoordSys**

### 7.12.1 PhotometryCoord.luminosity

**vodml-id: coordinates.PhotometryCoord.luminosity**
**type: ivoa:real**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 7.13 PhotometryCoordSys

TBC with photDM
**subset**
  **role: coords:PhysicalCoordSys.frame**
  **type: mango:PhotFilter**

## 7.14 RedshiftCoord

TODO : Missing description : please, update your UML model asap.
**subset**
  **role: coords:Coordinate.coordSys**
  **type: RedshiftSys**

### 7.14.1 RedshiftCoord.redshift

**vodml-id: coordinates.RedshiftCoord.redshift**
**type: ivoa:real**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 7.15 RedshiftSys

TODO : Missing description : please, update your UML model asap.

## 7.16 ShapeCoord

TODO : Missing description : please, update your UML model asap.
**subset**
  **role: coords:Coordinate.coordSys**
  **type: ShapeSys**

### 7.16.1 ShapeCoord.shape

**vodml-id: coordinates.ShapeCoord.shape**
**type: ivoa:string**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 7.17 ShapeSys

Coordinate systen to be used for shape measure

### 7.17.1 ShapeSys.frame

**vodml-id: coordinates.ShapeSys.frame**
**type: mango:coordinates.ShapeFrame**
**multiplicity: 1**
Frame of the shape measure. Gives a enumeration of the supported serializations.

## 7.18 StringCoordinate

TODO : Missing description : please, update your UML model asap.

### 7.18.1 StringCoordinate.cval

**vodml-id: coordinates.StringCoordinate.cval**
**type: ivoa:string**
**multiplicity: 1**
TODO : Missing description : please, update your UML model asap.

## 7.19 VocabCoordinate

TODO : Missing description : please, update your UML model asap.

## 7.20 ShapeFrame

Enumeration of the possible options to encode a shape in a string.
Enumeration Literals

**MOC** : **vodml-id:** coordinates.ShapeFrame.MOC
**description:** MOC serialization

**STCs** : **vodml-id:** coordinates.ShapeFrame.STCs
**description:** STCs serialization

# 8 TAP and MANGO

This not normative section gives possible tips to save and discover MANGO instances in TAP services. We suppose that the TAP service hosts catalogs which sources are MANGO instances. These catalogs are named *MANGO Catalogs*.

## 8.1 Storing MANGO Catalogs in TAP

For now this section only concerns the parameter. The associated data will be taken into account later.

- One master table for the catalogs with various meta-data out of the MANGO scope plus a unique identifier (primary key)
- One master sources table for the source instances with the catalog identifier and a primary key safer than the MANGO identifier.
- One table for each supported parameter with a foreign key for the join with the master source table

Although the model of the measures is hierarchical, it should be possible to flatten them in one single table considering that the model structure can be retrieved with the TAP_SCHEMA annotations (TBC)

This schema requires the server to explore all the parameter tables to retrieve whole MANGO instances. This process can be speed up by using the *MANGOCore* table.

### 8.2 *MANGOCore* Table

The discovery of *MANGO Catalogs* can be helped by a *MANGOCore* table located in the *schema* schema. As MANGO is not dedicated to any specific domain, we cannot define a set of core parameters, but parameters can be flagged as *Core Parameter*. This selection is left at the discretion of the curator. The *MANGOCore* table has set of columns per parameter class plus one for the catalog ID. It has one row per stored catalog. Each parameter has at least 2 columns: one with the UCD and one with the *Core* flag. TBC

## A Imported Models Instance

## B Changes from Previous Versions

No previous versions yet.