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

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 (Bradner, 1997).

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 (Salgado and Lemson et al., 2016) as well as for catalogs (Osuna et al., 2006). Other proposals, more model-agnostic, were focused on the data annotation in VOTables (Demleitner and Ochsenbein et al., 2016) (Derriere, 2016). 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 modeled. 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 organized depends on the targeted science case. Principal investigators and archive designers set

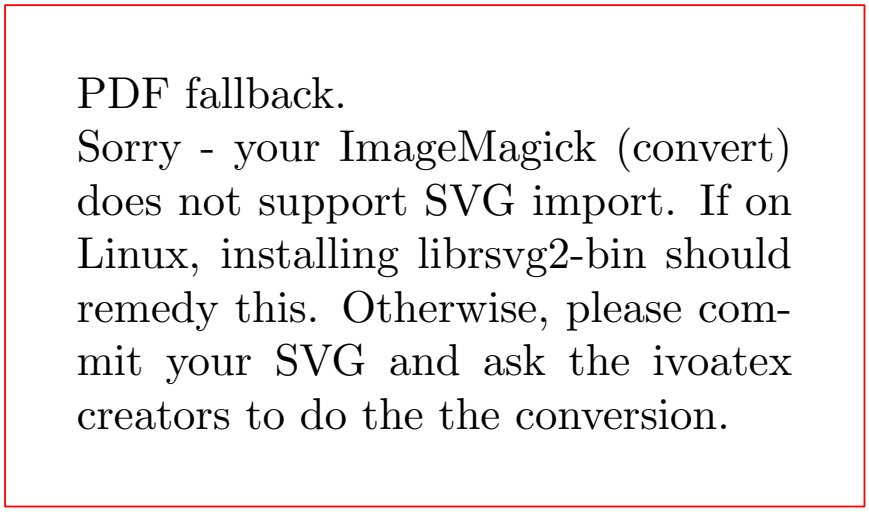

Figure 1: Architecture diagram for this document

up the data profile and structure it according to this goal which varies from 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 existings 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 (Arviset and Gaudet et al., 2010).

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 missions

- position
- orbit
- different source level (source types instead?)(star, planet, moon)
- status and classification
- orbiting system description

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

2.1.4 Morphologically Complex Structures

- morphology

TODO:
to be developed ...

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 ???

2.1.6 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. visualization), 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 heterogneneity
- Table content heterogneneity
- Huge variety of possible measures
- Huge variety of patterns of measure groups
- Use of different coodinate 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, timeseries, 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 aloso 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 lat and lon, 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 astrometrical (but also possibly photometric, statistical, ...) parameter space (Pineau and Derriere et al., 2017) .

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.

Error type 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; pos.eq.ra; pos.eq.dec	Correlation factor
Oriented Ellipse		
a	phys.angSize.smajAxis; pos.errorEllipse	Error ellipse semi-major axis
b	phys.angSize.sminAxis; pos.errorEllipse	Error ellipse semi-minor axis
theta	pos.posAng; pos.errorEllipse	Error ellipse position angle

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

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

- 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 ambigous. 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-dimentional 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 todifferent 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 dataproducts 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 computationnaly) 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 modeling 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 serialized in VO-DML.
- MANGO instances must support multiple instances of the same associated data class.

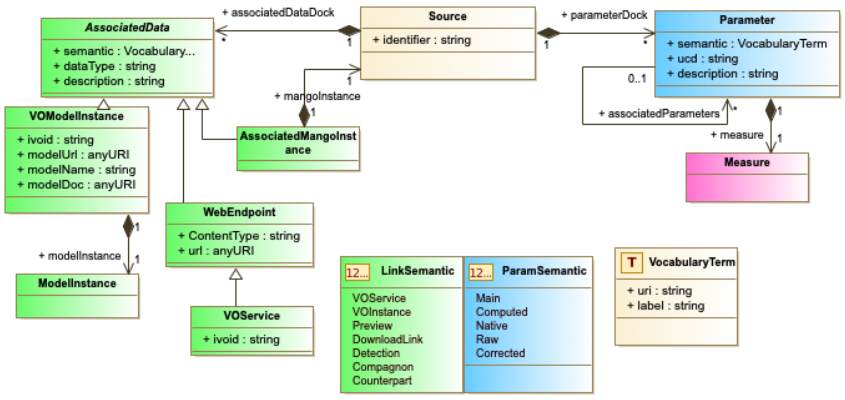

Figure 2: MANGO overview

- The presence of any associated data in MANGO instances must be optional.
- 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 paremeters, the `parameters` relation, and another for the associatedData, the `associatedData` relations. Both connectors have an open-ended cardinality.

Each source paremeter is hooked to the `MangoObject` by a wrapping class (`Parameter`) that contains anything necessary to identify its nature and roler. 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 referended 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 modeled 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 postion 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 responsability 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
`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
`Luminosity`	MANGO	phot.qqqchose
`HardnessRatio`	MANGO	phot.flux;arith.ratio
`Shape`	MANGO	phys.area
`Flag`	MANGO	meta.code
`Orbit`	MANGO	src.orbital
`Generic(String)Measure`	Measure	Appropiate physical UCD

Table 2: UCDs soace to be used for the supported parameters

3.1.2 Measure Extension

All `Measure` classes are built upon the same 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). These extended classes are part of MANGO.

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. . .).

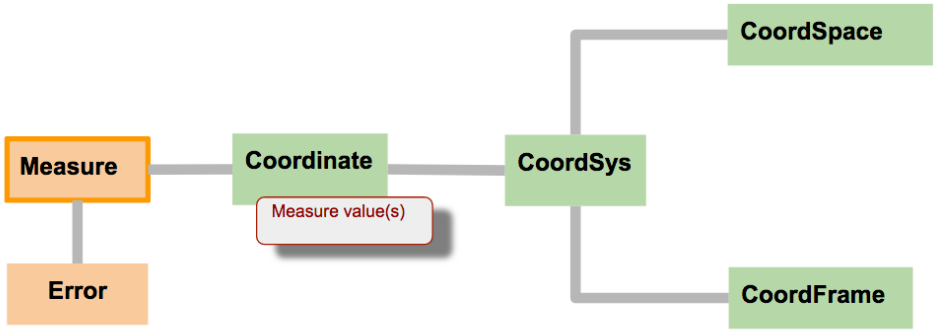

Figure 3: Measure/Coordinate pattern (simplified view)

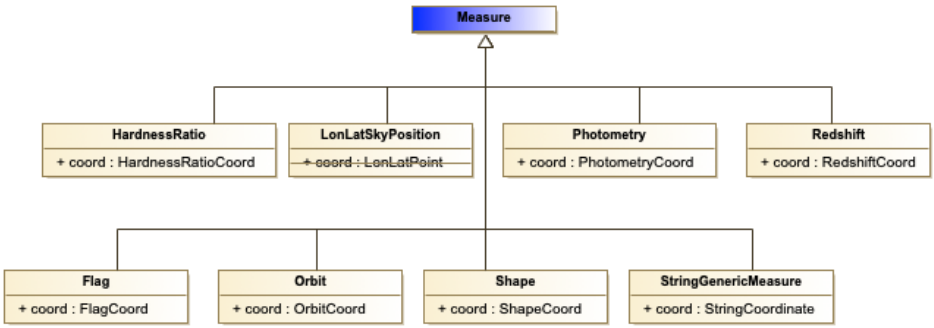

Figure 4: Mango extensions of Measure.

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 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 .

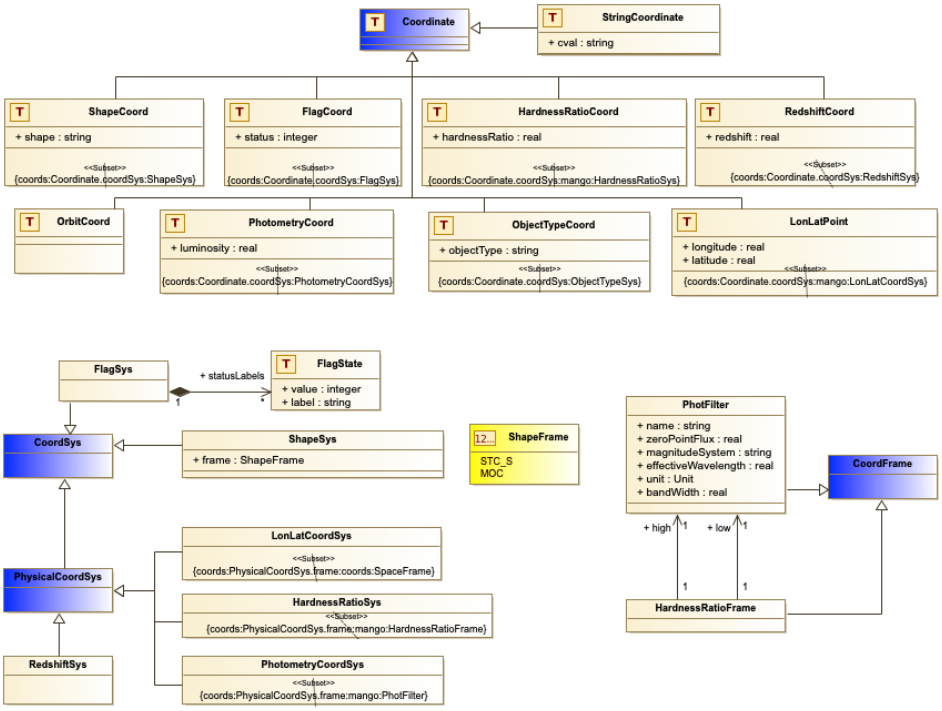

Figure 5: Mango extensions of Coordinates.

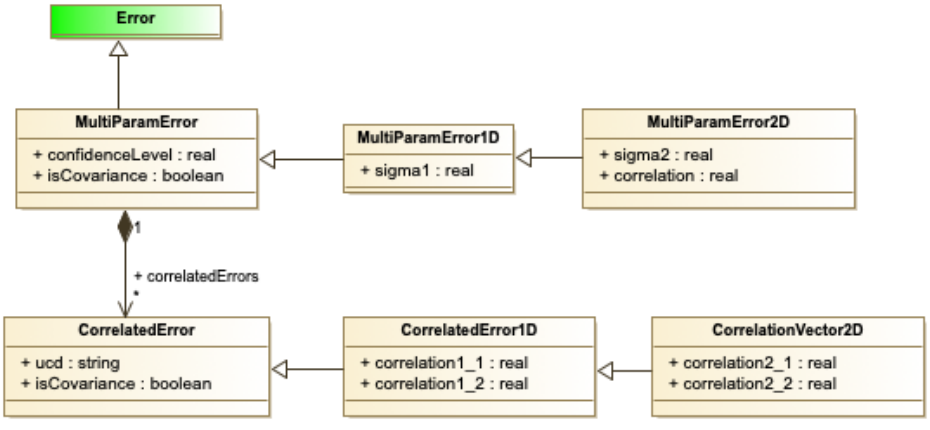

Figure 6: Mango correlated errors.

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 AssociatedMangoObject

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

4.1.1 AssociatedMangoObject.role

vodml-id: AssociatedMangoObject.role
type: mango:MangoInstanceRole
multiplicity: 0..1
Role of the associated Mango Source

4.1.2 AssociatedMangoObject.associatedMangoObject

vodml-id: AssociatedMangoObject.associatedMangoObject
type: mango:MangoObject
multiplicity: 1
Composition link pointing on one cab_msd instance associated with the source.

4.2 DataDock (Abstract)

Abstract reference to a particular dataset associated to the Source. These data set have a life cycle different from the this of the Mango object. This class is used to specify the type of the dataset as well as its role.

4.2.1 DataDock.semantic

vodml-id: DataDock.semantic
type: mango:stcextend.VocabularyTerm
multiplicity: 0..1
Optional vocabulary term giving a standard description of the associated data. This term does not necessary relate to a VO vocabulary e.g. UAT.

4.2.2 DataDock.description

vodml-id: DataDock.description
type: ivoa:string
multiplicity: 0..1
Free text description of the associated data

4.3 MangoObject

Root class of the model. Mango instance are meant ot be Source instances. A source has an identifier and to sets of hooks either for parameters or for associated data.

4.3.1 MangoObject.identifier

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

4.3.2 MangoObject.parameters

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

4.3.3 MangoObject.dataDock

vodml-id: MangoObject.dataDock
type: mango:DataDock
multiplicity: 0..*
Link pointing on all data associated with the source. This is an association because associated data have likely their proper live cycles.

4.4 ModelInstance

Placeholder for the mapping block of the model instance

4.5 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

4.5.1 Parameter.ucd

vodml-id: Parameter.ucd
type: ivoa:string
multiplicity: 1
UCD1+ giving the type of the physical measure. The model defines a UCD space for each Measure class.

4.5.2 Parameter.reductionStatus

vodml-id: Parameter.reductionStatus
type: mango:ReductionStatus
multiplicity: 0..1
Reduction status of the Measure carried by the Parameter.

4.5.3 Parameter.semantic

vodml-id: Parameter.semantic
type: mango:stcextend.VocabularyTerm
multiplicity: 0..1
Optional vocabulary term giving a standard description of the Parameter. This term does not necessary relate to a VO vocabulary e.g. UAT.

4.5.4 Parameter.description

vodml-id: Parameter.description
type: ivoa:string
multiplicity: 0..1
Free text description of the measure.

4.5.5 Parameter.measure

vodml-id: Parameter.measure
type: meas:Measure
multiplicity: 1
Reference to the meas:Measure subclass instance carrying the Parameter value(s)

4.5.6 Parameter.associatedParameters

vodml-id: Parameter.associatedParameters
type: mango:Parameter
multiplicity: 0..*
This relation allows to build logical sets of parameters e.g. associating a flag with a position. Such parameter sets have no particular semantic.

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
Link pointing on one VO instance instance associated with the source. This is an association because associated instances have likely their proper live cycles

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 WebEndpoint

Class for associated data referenced by an URL

4.8.1 WebEndpoint.ContentType

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

4.8.2 WebEndpoint.url

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

4.9 MangoInstanceRole

Enumeration of the possible roles of associated Mango Instances
Enumeration Literals

Detection : **vodml-id:** MangoInstanceRole.Detection
description: The associated Mango instance is another dectection of the same object

Companion : **vodml-id:** MangoInstanceRole.Companion
description: The associated Mango instance is a companion object e.g. binary star

Counterpart : **vodml-id:** MangoInstanceRole.Counterpart
description: The associated Mango instance is a source counterpart taken out of another dataset e.g cross-match result

4.10 ReductionStatus

Enumeration of the possible reduction status of a parameter
Enumeration Literals

Raw : **vodml-id:** ReductionStatus.Raw
description: The parameter carry a raw value

Calibrated : **vodml-id:** ReductionStatus.Calibrated
description: The parameter value is calibrated

Corrected : **vodml-id:** ReductionStatus.Corrected
description: The paramter value has been corrected this the reduction software. The operation usually comes after the calibration process e.g. adding a systematic error.

Computed : **vodml-id:** ReductionStatus.Computed
description: The parameter value has been computed. It is not directly derived from the raw data e.g. deetction flag

5 Package: stcextend

This package contains all object and type classes that has been extended from the Measure and Coordinates datamodels. This extension mechanism is used to add new types of measures while staying whithin the Mes/Coords pattern.

5.1 Flag

Measure to be used for status parameters

5.1.1 Flag.coord

vodml-id: stcextend.Flag.coord
type: mango:stcextend.FlagCoord
multiplicity: 1
Coordinate holding the statsu value

5.2 FlagCoord

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

5.2.1 FlagCoord.status

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

5.3 FlagState

Possible value of a status

5.3.1 FlagState.value

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

5.3.2 FlagState.label

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

5.4 FlagSys

Coordinate system to be used for statur measures.

5.4.1 FlagSys.statusLabel

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

5.5 HRFrame

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

5.5.1 HRFrame.low

vodml-id: stcextend.HRFrame.low
type: mango:stcextend.PhotFilter
multiplicity: 1
Low energy band

5.5.2 HRFrame.high

vodml-id: stcextend.HRFrame.high
type: mango:stcextend.PhotFilter
multiplicity: 1
Heigh energy band

5.6 HardnessRatio

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

5.6.1 HardnessRatio.coord

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

5.7 HardnessRatioCoord

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

5.7.1 HardnessRatioCoord.hardnessRatio

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

5.8 HardnessRatioSys

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

5.9 LonLatCoordSys

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

constraint
detail: LonLatCoordSys.coordSpace[0]

5.10 LonLatPoint

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

5.10.1 LonLatPoint.longitude

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

5.10.2 LonLatPoint.latitude

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

5.11 LonLatSkyPosition

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

5.11.1 LonLatSkyPosition.coord

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

5.12 ObjectType

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

5.12.1 ObjectType.coord

vodml-id: stcextend.ObjectType.coord
type: mango:stcextend.OrbitCoord
multiplicity: 1
TODO : Missing description : please, update your UML model asap.

5.13 ObjectTypeCoord

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

5.13.1 ObjectTypeCoord.objectType

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

5.14 ObjectTypeSys

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

5.15 Orbit

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

5.15.1 Orbit.coord

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

5.16 OrbitCoord

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

5.17 PhotFilter

Photometric filter description, compliant with photDM

5.17.1 PhotFilter.name

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

5.17.2 PhotFilter.zeroPointFlux

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

5.17.3 PhotFilter.magnitudeSystem

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

5.17.4 PhotFilter.effectiveWavelength

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

5.17.5 PhotFilter.unit

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

5.17.6 PhotFilter.bandWidth

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

5.18 Photometry

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

5.18.1 Photometry.coord

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

5.19 PhotometryCoord

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

5.19.1 PhotometryCoord.luminosity

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

5.20 PhotometryCoordSys

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

5.21 Redshift

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

5.21.1 Redshift.coord

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

5.22 RedshiftCoord

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

5.22.1 RedshiftCoord.redshift

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

5.23 RedshiftSys

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

5.24 Shape

Measure giving the shape of a source

5.24.1 Shape.coord

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

5.25 ShapeCoord

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

5.25.1 ShapeCoord.shape

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

5.26 ShapeSys

Coordinate systen to be used for shape measure

5.26.1 ShapeSys.shapeFrame

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

5.27 Temperature

TBC

5.27.1 Temperature.coord

vodml-id: stcextend.Temperature.coord
type: ivoa:string
multiplicity: 1
TODO : Missing description : please, update your UML model asap.

5.28 VocabularyTerm

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

5.28.1 VocabularyTerm.uri

vodml-id: stcextend.VocabularyTerm.uri
type: ivoa:string
multiplicity: 1
URI extarcted from the DRF document and refering ot the word

5.28.2 VocabularyTerm.label

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

5.29 ShapeFrame

Enumeration of the possible options to encode a shape in a string.
Enumeration Literals
MOC : **vodml-id:** stcextend.ShapeFrame.MOC
description: MOC serialization

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

6 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*.

6.1 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.

6.2 *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 an one with the *Core* flag. TBC

A Imported Models Instance

B Changes from Previous Versions

No previous versions yet.

References

Arviset, C., Gaudet, S. and the IVOA Technical Coordination Group (2010), 'IVOA architecture', IVOA Note.
http://www.ivoa.net/documents/Notes/IVOAArchitecture

Bradner, S. (1997), 'Key words for use in RFCs to indicate requirement levels', RFC 2119.
http://www.ietf.org/rfc/rfc2119.txt

Demleitner, M., Ochsenbein, F., McDowell, J. and Rots, A. (2016), 'Referencing stc in votable', Note 2010-06-18.
http://ivoa.net/documents/Notes/VOTableSTC/

Derriere, S. (2016), 'Referencing stc in votable', Note 2011-05-12.
https://wiki.ivoa.net/internal/IVOA/PhotometryDataModel/NOTE-PPDMDesc-0.2-20110512.pdf

Osuna, P. et al. (2006), 'Catalog data model', Catalog Data Model Work Package.
https://wiki.ivoa.net/twiki/bin/view/IVOA/IVAODMCatalogsWP

Pineau, F. X., Derriere, S., Motch, C., Carrera, F. J., Genova, F., Michel, L., Mingo, B., Mints, A., Nebot Gómez-Morán, A., Rosen, S. R. and Ruiz Camuñas, A. (2017), 'Probabilistic multi-catalogue positional cross-match', *aap* **597**, A89, arXiv:1609.00818.
http://doi.org/10.1051/0004-6361/201629219

Salgado, J., Lemson, G. and Demleitner, M. (2016), 'Source dm', Source DM 2016-05-10.
https://wiki.ivoa.net/twiki/bin/view/IVOA/SourceDataModel