International

Virtual

Observatory

Alliance

Photometry Data Model

Version 0.3

Working Draft 2011 May 11

Editor(s):
> Jesus Salgado
> Pedro Osuna

Author(s):
> Jesus Salgado
> Carlos Rodrigo
> Pedro Osuna
> Deborah Baines
> Mireille Louys
> Evanthia Hatziminaoglou
> Jonathan McDowell
> Mark Allen

Abstract

The Photometry Data Model (**PhotDM**) standard describes photometry filters, photometric systems, magnitude systems, zero points and its interrelation with
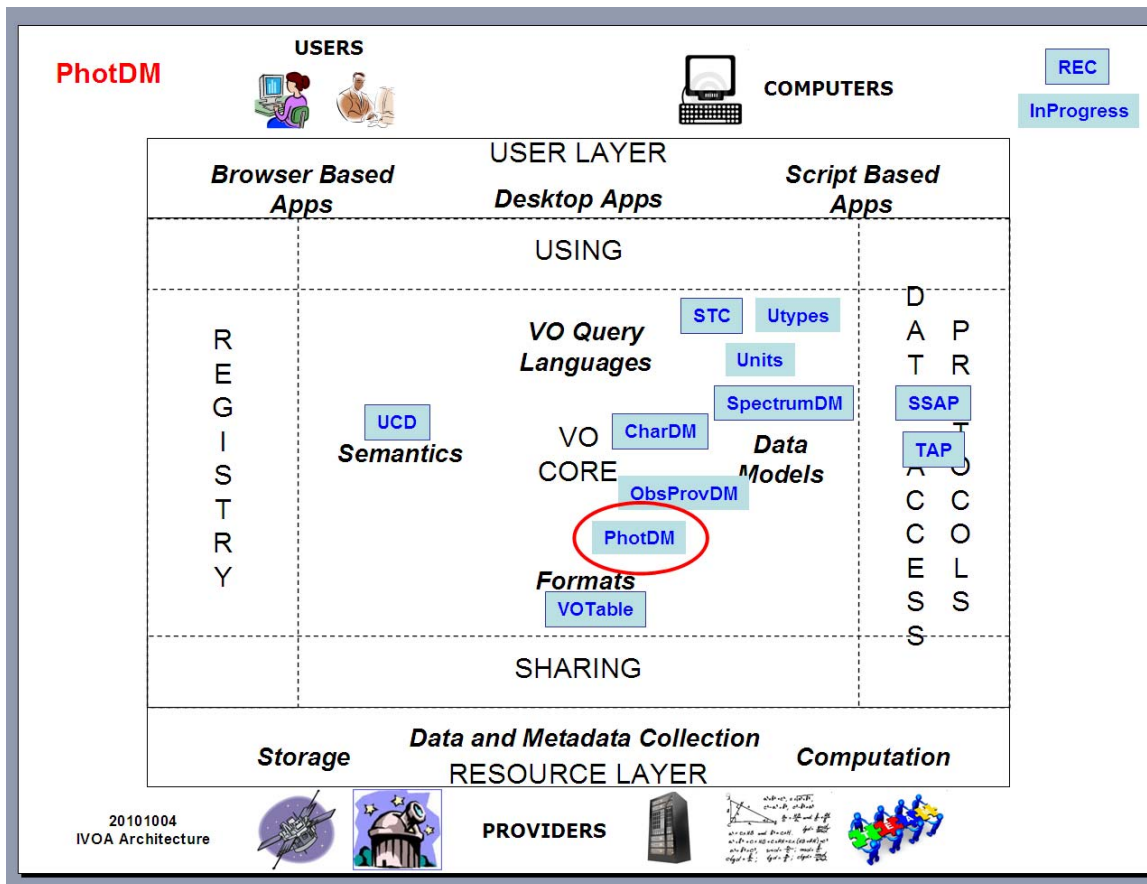
the other IVOA data models through a simple data model in order to allow the creation of protocols to access photometric data in magnitudes. PhotDM could be used in conjunction with other IVOA Data Access Protocol such as SSAP (Simple Spectra Access Protocol) or TAP (Table Access Protocol).

PhotDM makes reference to the CharDM (Characterization DM), to the SpectrumDM and to the provenance part of the ObsProvDM (Observation and Provenance DM).

As with most of the VO Data Models, PhotDM will make use of STC, Utypes, Units and UCDs. PhotDM will be serializable with a VOTable.

Link to IVOA Architecture

The figure below shows where Photometry DM fits within the IVOA architecture:

Status of This Document

Acknowledgements

Contents

1 Introduction

A key role of the VO is to facilitate finding data and combining them in a scientifically meaningful way. An example of usage of such an assembly of data is the construction of the spectral energy distribution (SED) of astrophysical sources from data at different wavelengths [1][2][3][4] that give the astronomer a glimpse on the underlying physical processes. The construction of an SED requires that photometric data be described in a standard way. This note proposes a data model that describes photometric measurements, and outlines how such measurements could be made available through the Simple Spectral Access Protocol (SSAP) or Table Access Protocol (TAP) services so that the photometric measurements can be used and combined in scientific software tools.

In order to construct a Spectral Energy Distribution (here after SED), all that is required is the measured flux densities of the source with associated flux errors, for a specific set of wavelengths and bandwidths. Many of the catalogues available in the VO already contain these measurements. The next step is to combine photometric data from different catalogues themselves originating from different instruments and provide comparable flux measurements. Due mostly to

historic reasons, many measurements available in catalogues are expressed in magnitudes in various filter systems, so an important part of the proposed photometry calibration data model focuses on the description of the filter and magnitude systems with sufficient information for making the transformation of magnitudes into flux densities.

The proposed model is based on the filter transmission curves (or system throughput) of a photometric system, source-independent values of the average wavelength and width at half maximum (bandwidth), and a photometric zero point, necessary for the conversion of magnitudes to flux densities. While the description of photometric systems and photometric measurements can be more complex than this, the proposed model is intended to describe these basic properties in order to support some simple SED use cases. More detailed descriptions are envisaged to be developed within the characterisation data model, and may eventually be able to support use cases requiring higher level of accuracy.

At the current stage, there is an important lack to discover photometry filter information in an automatic way and scientists are forced to look into the literature to find the needed details to handle photometric data from a specific photometry filter. This information could be extracted from a central repository of photometry filter information, *Filter Profile Service* henceforth, which would expose this information so software client applications could discover it. This current data model will also add the required data model support to allow the standardization of the protocol used by Filter Profile Services.

This model focuses on simple use-cases and provides basic descriptions for photometric systems and photometric measurements and can be considered as the core of other IVOA modeling efforts for Photometry.

In this note we first summarize the key points about astronomical photometry in section 2. We then detail the metadata structure with elements of the proposed model in section 3. Section 4 describes use cases in which the model description could be used in making photometry data available through VO protocols, and, very briefly how, scientific tools could use this information.

2 Astronomical Photometry

In astronomy, photometry refers to measuring the brightness, flux or intensity of an astrophysical object. Usually, photometry refers to measurement of radiation taken within broad wavelength bands. A set of well-defined band-passes (or filters) with a known sensitivity to incident radiation is called a *photometric system*. Because of historical and practical reasons, photometric data is not stored in flux format but in magnitudes. Expressing fluxes in terms of photometric magnitudes can be considered as a shortcut to expressing ratios of fluxes. For two stars with flux f_1 and f_2, their magnitude difference can be expressed as:

$$m_1 - m_2 = -2.5\log_{10}(f_1 / f_2)$$

Note: We will use the standard and common Pogson magnitudes conversion formulae. See ZeroPoint.type field description in the text.

A colour index for an object can be defined as e.g.

$$B - V = m_B - m_V$$

where m_B and m_V are the B and V magnitudes of the object, respectively.

Although some filters are considered standard, a certain implementation of a filter in a certain telescope can be close to, but vary from, the standard values. Apart from the band-pass covered by the filter, real filters are not ideal-square filters, i.e., the response over a wavelength range, $\Delta\lambda$, is not 100% over the entire band-pass. Furthermore, detectors detect only a fraction of the incident photons. As a result, all filters have an associated transmission curve that describes the fraction of detected per incident photons, taking into account all the elements along the optical path. Aperture corrections are described in the IVOA Spectral DM.

In order to calibrate the measurements and convert them into physical units, known objects with well-known emission in the relevant parts of the electromagnetic spectrum are observed with the same configuration (detector+filter). This produces a certain flux measurement which for this astronomical object can be considered as the reference magnitude. Then:

$$m - m_r = -2.5\log_{10}(f / f_r)$$

hence

$$f = f_r 10^{-\frac{m - m_r}{2.5}}$$

where f and m are the flux and magnitude, respectively, of the observed source, and f_r and m_r are the flux and magnitude, respectively, of the reference spectrum. If one expresses the flux as a function of the flux of a source that is considered to have zero magnitude, the equation would be:

$$f = f^0 10^{-\frac{m}{2.5}}$$

where f^0 is now called the zero point flux. This is the general conversion formulae for Pogson magnitudes.

The selection of certain f_r defines a magnitude system. The three most commonly used magnitude systems are the Vega magnitude, AB_v magnitude and ST_λ magnitude systems. The Vega magnitude system uses the spectrum of Vega (Alpha Lyrae) as the reference spectrum $F_R(x)$. The AB_v magnitude system uses reference spectrum defined by a constant f_v flux density, and the ST_λ magnitude system uses a reference spectrum of a constant f_λ flux density. The values of f_v and f_λ that respectively define the zero points AB_v=0 and ST_λ=0 have been chosen to be the mean flux density of Vega in the Johnson V band.

$$AB_v = 0; f_r = 3.63x10^{-20} erg.cm^{-2}.s^{-1}Hz^{-1}$$

$$ST_\lambda = 0; f_r = 3.63x10^{-9} erg.cm^{-2}.s^{-1}.Å^{-1}$$

A convenient graphical representation of these systems is shown in Figure 3.1 of the Synphot users manual:

(http://www.stsci.edu/resources/software_hardware/stsdas/synphot/SynphotManual.pdf).

For a photometric system that uses Vega magnitudes, the zero point flux for each filter is the average flux density of Vega over that passband, $F_{Vega}(x_0)$ with the appropriate x_0 for that bandpass. Some typical values of F_{Vega} are tabulated in Bessell and Murdin (Encyclopedia of Astronomy 2000, p. 1939) for the Johnson photometric system. The commonly referred to spectrum of Vega in digital form described in Bohlin and Gilliland 2004, A&A 2004 vol. 127 pp. 3508 is available as file alpha_lyr_stis_002.fits at:
http://www.stsci.edu/instruments/observatory/cdbs/calspec

Although this can be derived by the Pogson conversion formulae, in the AB system, the flux (in units of *erg cm^{-2} s^{-1} Hz^{-1}*) corresponding to a given magnitude is simply obtained via the shorthand formulae:

$$m_{AB,v} = -\log_{10}(f_v) - 48.6$$

And, in the same way, in the ST system, the flux (in units of *erg cm^{-2} s^{-1} Å^{-1}*) corresponding to a given magnitude is:

$$m_{ST,\lambda} = -\log_{10}(f_\lambda) - 21.1$$

(See equivalence to Pogson conversion formulae in section 5.1)

3 Photometry Data Model

The following Data Model describes photometry filters, photometric systems, magnitude systems, zero points and its interrelation with the other IVOA data models.

The main class in this diagram is Photometry Filter. This class contains all the attributes necessary to describe a filter from the data discovery point of view. Other general attributes to characterize the filter would be done through the IVOA Characterization Data Model.

A Photometric System is considered a container of Photometry Filters usually as belonging to the same observatory/telescope or because it belongs to the same standard system.

A magnitude system is characterized for a certain reference spectrum that will produce a certain zero point for a certain photometry filter. This reference spectrum could be an ideal one (as in STmag and ABmag systems), a Vega-like spectrum (as in Vegamag systems) (please notice that different Vega spectrum versions have been historically used) or any other. In many cases, the reference spectrum has been calculated as an average of spectra from several astronomical objects. This would be characterized by a set of Source instances.

A zero point would then be a flux value that can be considered as zero magnitude, so its value will allow conversions from fluxes to magnitudes and the other way around. It has associated a photometry filter and it also depends on the magnitude system (reference spectrum) used to calculate this magnitude.

There are different types of zero points (Pogson, asinh, linear etc) that will essentially differ in the way that getFluxFromMagnitude and getMagnitudeFromFlux operators are implemented plus extra information that could be needed to do these conversions.

An intermediate class, PhotCal, can be understood as a certain photometry filter instance, i.e., a certain photometry filter using a certain magnitude system and linked to a certain zero point class. This PhotCal class connects the relevant photometry filter component objects with the SpectralDM, allowing the handling of photometric data through e.g. SSAP or TAP services.

A Spectrum would have a Characterization Coordsys element that will have associated a certain PhotCal element in the case of photometry data.Using this information, magnitudes from different photometric systems could be compared between them or compared to spectroscopic data expressed in flux.
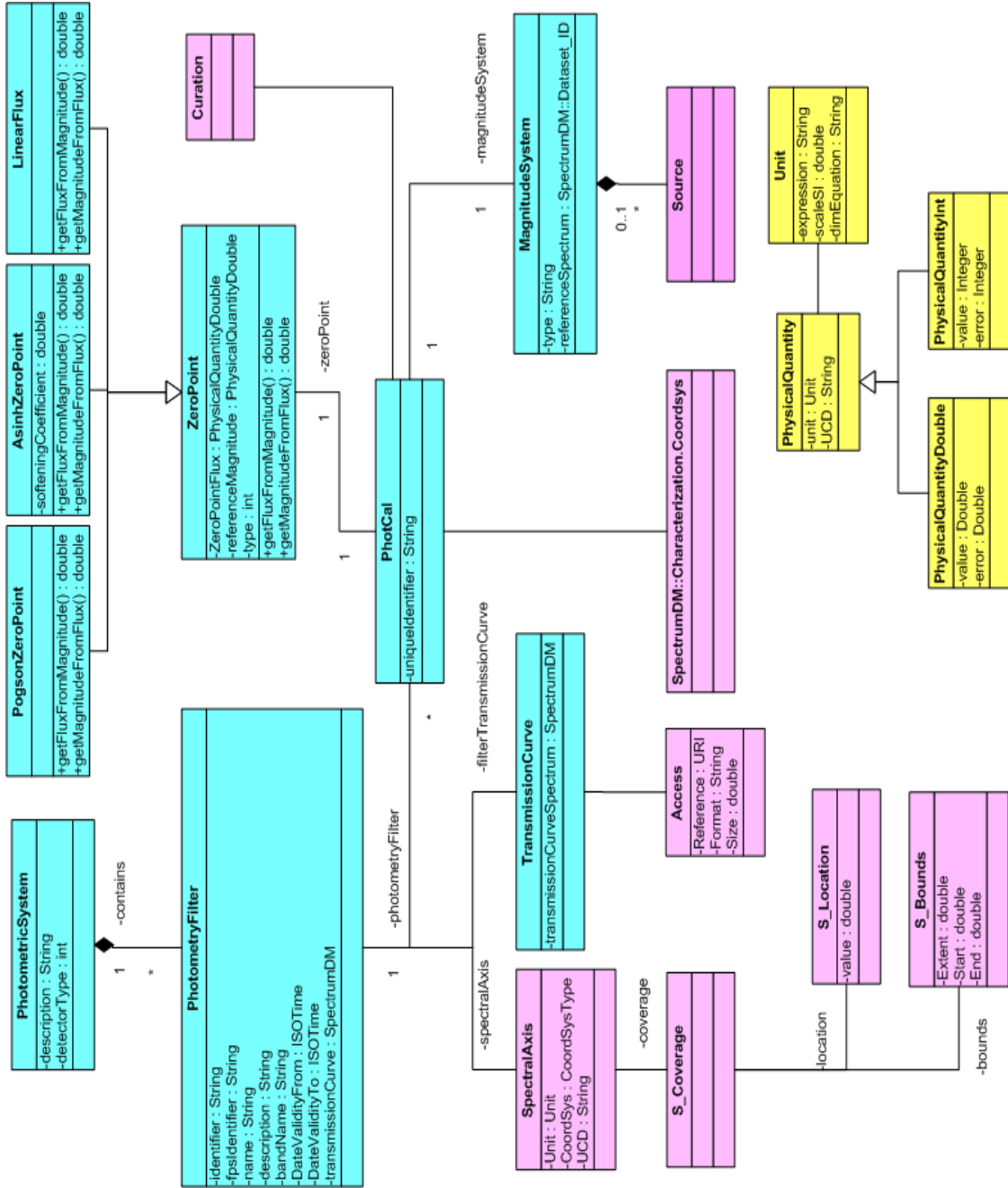
Figure 1 In blue, class diagram of the Photometry Calibration Data Model: reused classes from the Characterization data model are shown in pink. Top right corner shows the Curation class re-used from SpectrumDM. In yellow, simplified physical quantity class created to glue the different fields that describe a measurement.

In order to fully describe values of the magnitudes inside photometry point instances, the class diagram makes use of physical quantity classes. These classes glue all the basic fields that compose a physical measurement: value, error, units, etc. However, within the present specification, we will describe individual attributes of the different quantities and as a consequence. All the utypes will be also generated from individual physical quantity attributes what will facilitate the use within IVOA Data Access Layer protocols.

3.1 PhotometricSystem

This class briefly describes the photometric system that contains a set of photometry filters. Photometry filters can be contained in a certain photometric system as part of the same observatory/telescope or as part of a known system.

3.1.1 PhotometricSystem.description: String

This String contains a human readable short-text representation of the photometric system. This will allow client applications to display textual information to final users.

Examples:

Sloan
Johnson

3.1.2 PhotometricSystem.detectorType: enum

Detector type associated to this photometric system. Possible values are:

Type of detector	Value	Examples
Energy Counter	0 (default)	Energy amplifiers devices
Photon Counter	1	CCDs or photomultipliers

This will be used in order to decide how to calculate the flux average in, e.g., the synthetic photometry calculations. At current state, this list is exhaustive.
See photometry filter transmission curve description to understand how to use this field.

3.2 PhotometryFilter Class

This is the main class that describes a photometry filter.

3.2.1 PhotometryFilter.identifier: String

This field identifies, in a unique way, within a certain Photometry Filter Profile service, a filter. Although the main requirement of this data model field is to be unique within a Filter Profile Service, the suggested syntax would be:

```
Facility/Subcategory/Band[/Suffix]
```

where *Facility* is the telescope, observatory, space mission, etc that has this filter, *Subcategory* is a meaningful classification of filters within a facility (usually instrument), *Band* is the generic name used to describe the wavelength band used by this filter and *Suffix* is optional metadata added to the unique identifier string to ensure uniqueness within a Filter Profile Service.

Example:
```
SDSS/SDSS.G/G
```

3.2.2 PhotometryFilter.fpsIdentifier: String

IVOA identifier of the filter profile service where this photometry filter is registered to be used in the discovery of all the relevant photometry filter properties.

This identifier follows the IVOA syntax defined for IVOA identifiers [5] which gives a string built up as:

```
ivo ://<ivoa authority id>/<resource key>
```

Example:
```
ivo://svo/fps
```

where svo is the authority id, fps is the resource key of the service.

The service url of the filter profile service would be obtained from the registry by requesting the associated information of this registry resource, e.g., once registered the service URL associated to this Filter Profile Service would be, e.g.:

```
http://svo.cab.inta-csic.es/theory/fps/
```

3.2.3 PhotometryFilter.name: String

This String contains a human readable representation of the filter name. This will allow client applications to display information to the final user.

Example:
```
SDSS.G
```

3.2.4 PhotometryFilter.description: String

This String contains a verbose human readable string description of the filter. This will allow client applications to display text information to the final user.

3.2.5 PhotometryFilter.bandName: String

This String contains a standard representation of the spectral band associated to this filter (if any). This information is useful for human interpretation but it is discourage to use it for discovery purposes. The reason is that a filter is not always properly represented by a standard band so filters could be lost in a query response.

Examples:
```
U
B
V
```

Where U,B,V corresponds to ultraviolet, blue and visible respectively.

3.2.6 PhotometryFilter.SpectralAxis

This spectral axis object associated to a photometry filter describes the spectral coverage (band) for this filter. See IVOA Characterization DM for a detailed description.

3.2.6.1 PhotometryFilter.SpectralAxis.Unit: IVOA.Unit

This string represents the unit used by all data model fields used for the spectral axis components characterization (coverage bounds, effective wavelength, etc)

3.2.6.2 PhotometryFilter.SpectralAxis.UCD: String

This data model field contains a Unified Content Description string (UCD) [6] that specifies the nature of the spectral axis for this filter. This applies to the full spectral axis description of the filter.

Example:
```
em.wl
```

Where *em.wl* indicates that the spectral coordinate is provided in wavelength.

3.2.6.3 PhotometryFilter.SpectralAxis.Coverage.Location: Char.SpectralAxis.Coverage.Location

A reference position along the spectral axis coverage of the referenced photometry filter.

Although this will reuse the Char.SpectralAxis.Coverage.Location element of the Characterization Data Model, the basic elements of this object are described within the context of a photometry filter as follows.

3.2.6.4 PhotometryFilter.SpectralAxis.Coverage.Location.Value: double

A spectral coordinate value that can be considered by the data provider as the more representative for this specific filter band-pass. The selection of this value should take into account the filter transmission curve profile and in general should be close to the wavelength mean value, defined [7] as:

$$\lambda_{mean} = \frac{\int T(\lambda)\lambda d\lambda}{\int T(\lambda)d\lambda}$$

where λ_{mean} is the spectral bounds mean value, $T(\lambda)$ is the transmission curve (see below), λ is the wavelength. Please notice that, since the transmission curve will only be defined in a specific spectral range, the integrals will also be effectively defined in this spectral range.

Another convenient definition of an effective wavelength is the "pivot wavelength" defined as follows:

$$\lambda_{pivot} = \sqrt{\frac{\int T(\lambda)\lambda d\lambda}{\int T(\lambda)\,d\lambda/\lambda}}$$

It can be proved that the pivot wavelength fulfills the following relation between the F_λ and F_ν:

$$\langle F_\nu \rangle = \langle F_\lambda \rangle \lambda_{pivot}^2 / c$$

Other definitions for effective wavelengths commonly used in the literature are source dependent as, e.g., the isophotal wavelength:

$$\lambda_{mean} = \frac{\int \lambda F_\lambda(\lambda)T(\lambda)d\lambda}{\int F_\lambda(\lambda)T(\lambda)d\lambda}$$

Or the photon distribution based effective wavelength:

$$\lambda'_{mean} = \frac{\int \lambda^2 F_\lambda(\lambda)T(\lambda)d\lambda}{\int \lambda F_\lambda(\lambda)T(\lambda)d\lambda}$$

but these source dependent definitions have two caveats:

- o Real spectra do not necessarily satisfy the requirements of the mean value theorem, which could produce multiple values for the wavelength
- o Calculation of these wavelengths implies the knowledge of F_λ (usually what you want to measure) and it does not look like an intrinsic property of the photometry filter.

3.2.6.5 PhotometryFilter. SpectralAxis.Coverage.Bounds.Extent: double

For square filters (100% between the minimum and maximum wavelength and 0% otherwise), the bandwidth could be described as $\lambda_{max} - \lambda_{min}$.

However, for real filters, the bandwidth is not very usable to describe the band-pass of the filter, but the effective width, that can be described as follow:

$$w = \frac{\int T(\lambda)d\lambda}{Max(T(\lambda))}$$

where w is the effective width, $T(\lambda)$ is the transmission curve (see below) and $Max(T(\lambda))$ the maximum value of the transmission curve. As in previous points, please notice that, since the transmission curve will be only defined in a specific spectral range, the integrals will also be defined in this spectral range.

3.2.6.6 PhotometryFilter.SpectralAxis.Coverage.Bounds.Start: double

Also called λ_{min} in the rest of the document, this is a spectral value that better describes the minimum value of the spectral range of the filter band-pass. In general, although this will not be imposed in order to allow a better description for different types of transmission curves, this quantity will be close to:

$$\lambda_{min} = \lambda_{mean} - \frac{w}{2}$$

In practice, this could be taken as the minimum value of the filter transmission curve.

3.2.6.7 PhotometryFilter.SpectralAxis.Coverage.Bounds.Stop: double

Also called $\lambda_{\max}$ in the rest of the document, this is a spectral value that better describes the maximum value of the spectral range of the filter band-pass. In general, although this will not be imposed in order to allow a better description for different types of transmission curves, this quantity will be close to:

$$\lambda_{\max} = \lambda_{mean} + \frac{w}{2}$$

In practice, this could be taken as the maximum value of the filter transmission curve.

3.2.7 PhotometryFilter Time Validity Range

The following fields will be used to characterize the validity range of this specific photometry filter configuration. This is particularly useful for ground based telescopes where filter, electronics, etc could easily change generating versions of the same photometry filter..

3.2.7.1 PhotometryFilter.DateValidityFrom: ISOTime

Start time of the time coverage when this filter configuration is applicable. String time format accepted, ISO8601:

```
YYYY-MM-DD[T[hh[:mm[:ss[.s[TZD]]]]]]
```

3.2.7.2 PhotometryFilter.DateValidityTo: ISOTime

End time of the time coverage when this filter configuration is applicable. String time format accepted, ISO8601:

```
YYYY-MM-DD[T[hh[:mm[:ss[.s[TZD]]]]]]
```

3.2.8 PhotometryFilter.transmissionCurve

Here we consider how wavelengths/frequencies are filtered in the whole acquisition chain for a calibrated observation stemming from a given data collection.

This means that within the same data collection most observations will point to the same PhotometryFilter.transmissionCurve.

The effective transmission curve is represented as a 2-D graph that describes the transmission properties of the filter over a wavelength range defined by the filter bandpass.

It is composed of a spectral coordinate in the x-axis and a scalar in the y-axis This effective response curve encloses all the possible components that modifies the energy/photon collection.
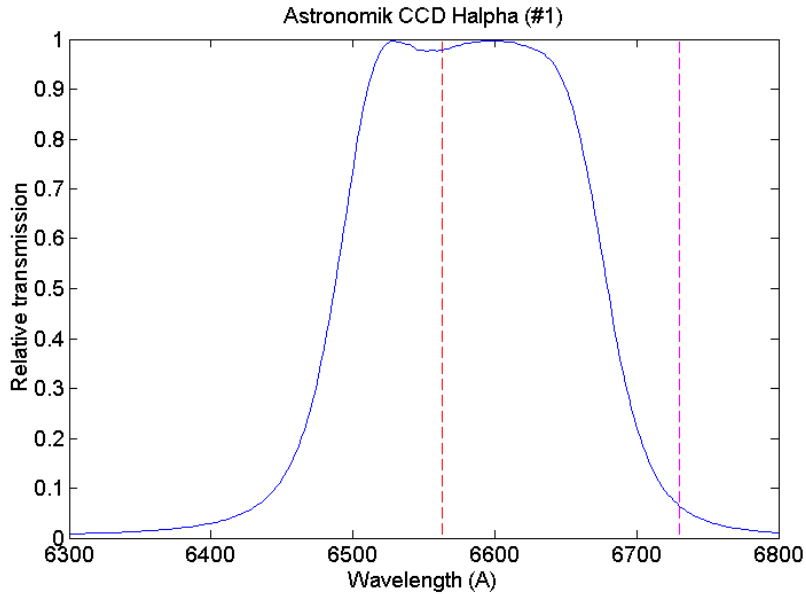
Figure 2: Transmission Curve example

This curve can be used, e.g. for the creation of synthetic photometry [8][9] from an observational or a theoretical spectrum by applying it to the spectrum in the filter band-pass. Taking as input a certain flux, the effective flux as seen using a certain filter would be, for energy counters [10]:

$$f(\lambda_{eff}) = \frac{\int T(\lambda)f(\lambda)d\lambda}{\int T(\lambda)d\lambda}$$

And for photon counters (like CCDs or photomultipliers):

$$f(\lambda_{eff}) = \frac{\int T(\lambda)f(\lambda)\lambda d\lambda}{\int T(\lambda)\lambda d\lambda}$$

Where $T(\lambda)$ is the transmission curve, $f(\lambda)$ is the flux of the spectrum. As the transmission curve is defined only in the filter band-pass, the limits of the

integrals corresponds to the spectral range where the transmission curve is defined (stored as *PhotometryFilter.SpectraAxis.Coverage.Bounds* in this data model)

The transmission curve can be closely (although not fully) identified as a spectrum. There are various ways to provide this information either directly in an embedded table, or using a reference to a spectrum serialized file.

Spectral and transmission coordinates can be gathered directly as a table using Spectrum Utypes:

- *spec:Data.SpectralAxis.Value* for the spectral coordinate (wavelength, energy or frequency) with units and ucd defined in spec:Char.SpectralAxis.units and spec:Char.SpectralAxis.ucd.

- spec:Data.FluxAxis.Value for the Flux coordinate, here unitless and containing the transmission between 0 and 1.

3.2.9 *PhotometryFilter.transmissionCurve.spectrum*

This data model field stores points of the curve in place in a simple table using spectrum data fields as shown above. See serialization example in Appendix C section 6.1

3.2.10   PhotometryFilter.transmissionCurve.Access: IVOA.Access

If the transmission curve is hooked as an external file, we use the *Access* class defined in the Observation CoreComponents data model [11] and inherited from the SSA specification [12]

3.2.10.1   PhotometryFilter.transmissionCurve.Access.Reference

The access reference is a URI (typically a URL) which can be used to retrieve the specific dataset described in a row of the query table response.

3.2.10.2   PhotometryFilter.transmissionCurve.Access.Format

The PhotometryFilter.transmissionCurve.Access.Format data model field tells the MIME type of the file pointed to and used to store the curve points. Values for this string can generally be:

:
```
application/fits
application/x-votable+xml
text/csv
text/xml
```

The filecontent will be a spectrum serialization with *Dataset.DataModel* set to "Spectrum1.1" for instance , and all necessary fields for the spectral and flux coordinates.

3.2.10.3 PhotometryFilter.transmissionCurve.Access.Size

Approximate estimated size of the dataset, specified in kilobytes. This would help the client estimate download times and storage requirements when generating execution plans. Only an approximate, order of magnitude value is required (a value rounded up to the nearest hundred KB would be sufficient).

3.3 PhotCal class

Class to describe the use of a photometry filter by using a certain magnitude system configuration. It has associated a certain zero point object.

3.3.1 PhotCal.identifier: String

This field identifies, in a unique way, within a certain Photometry Filter Profile service, a zero point assigned to a filter and a certain photometric system type. Although the main requirement of the uniqueIdentifier is to be unique within a Filter Profile Service, the suggested syntax would be:

Facility/Subcategory/Band/Photometric System Type[/Suffix]

where *Facility* is the telescope, observatory, space mission, etc that has this filter, *Subcategory* is a meaningful classification of filters within a facility (usually instrument), *Band* is the generic name used to describe the wavelength band used by this filter *Photometric System Type* makes reference to the type of system as per classification within this document and *Suffix* is optional metadata added to the unique identifier string to ensure uniqueness within a Filter Profile Service.

Please notice the suggested syntax of PhotCal unique identifier syntax corresponds with the Photometry Filter unique identifier concatenated with the photometric system type.

Example:

SDSS/SDSS.G/G/AB

3.3.2 PhotCal.zeroPoint: ZeroPoint

Zero point object associated to this PhotCal instance.

3.3.3 PhotCal.magnitudeSystem: MagnitudeSystem

Magnitude system object associated to this phot cal instance.

18

3.4 ZeroPoint class

This class is used to characterize a zero point flux obtained during the calibration of a certain photometry filter on a certain photometric system configuration. This object includes references to the relevant Photometric System and Photometry Filter objects.

3.4.1 ZeroPoint.zeroPointFlux.value: double

Flux of an astronomical object that produces a magnitude=0 for this particular filter and photometric system. This quantity is necessary to convert to flux a certain magnitude.

For Pogson magnitudes (see section 3.2.5) it will be used in the following way:

$$f = f^0 \, 10^{-m/2.5}$$

See ZeroPoint.type description for other definitions.

The flux could be expressed as f_λ or f_ν, leaving the characterization of the type of flux to the units in which this quantity is expressed.

3.4.2 ZeroPoint.zeroPointFlux.unit: String

Unit in which the zero point flux is expressed, corresponding to a magnitude equal to zero, unless reference magnitude indicates otherwise.

3.4.3 ZeroPoint.zeroPointFlux.error: double

Total error estimated on the zero point flux and expressed in the same units.

3.4.4 ZeroPoint.referenceMagnitude.value: double

Most of the time, the zero point flux is defined for a magnitude=0 value. However, to give room to other cases, another reference magnitude value can be given instead of zero. The use of this reference magnitude is described in the different getMagnitudeFromFlux and getFluxFromMagnitude zero point extension operations.

Please notice that, by default, reference magnitude will be zero unless specified otherwise.

Reference magnitude is a dimensionless variable.

3.4.5 ZeroPoint.referenceMagnitude.error: double

Total error estimated of the reference magnitude whenever applicable.
Reference Magnitude error is a dimensionless variable.

3.4.6 ZeroPoint.type: enum

Usual definition of magnitudes, also called Pogson magnitudes, can be improved for faint sources by replacing the usual logarithm with an inverse hyperbolic sine function. These kinds of magnitudes are called "asinh magnitudes" or "luptitudes" [9]

Zero point type	Value	Description
Pogson	0 (default)	Usual definition
Asinh	1	Used for faint sources, replacing the usual logarithm with an inverse hyperbolic sine function.
LinearFlux	2	Linear (not logarithmic) magnitudes used in Radio, Far Infrared, X-Ray spectral

The main difference between the three types of zero points is the conversion formulae to be used when translating magnitudes into flux and reverse.

In the ZeroPoint class we define two conversion functions; getMagnitudeFromFlux() and getFluxFromMagnitude() defined as:

- getMagnitudeFromFlux()
    - Input Parameters: Flux given in units defined in the ZeroPoint.unit data model field.
    - Output Result: Corresponding magnitude in double.

- getFluxFromMagnitude()
    - Input Parameters: Magnitude in double
    - Output Result: Corresponding flux given in units defined in the ZeroPoint.unit data model field.

3.5 PogsonZeroPoint

This is an extension of the ZeroPoint object to accommodate standard logarithm magnitudes. It has no supplementary attributes but specific conversion functions.

3.5.1 PogsonZeroPoint.getFluxFromMagnitude()

Operator to convert from a flux to a magnitude for Pogson magnitudes. For Pogson magnitudes, the usual definition should be used:

$$f = f_0 . 10^{-(m-m_0)/2.5}$$

Where f is the associated flux, f_0 is the flux of reference, m_0 is the magnitude of reference (by default equals to zero) and m is the observed magnitude.

3.5.2 PogsonZeroPoint.getMagnitudeFromFlux()

Operator to convert from a flux to a magnitude for Pogson magnitudes. For Pogson magnitudes, the usual definition should be used:

$$m = m_0 - 2.5 \log\left(\frac{f}{f_0}\right)$$

Where f is the associated flux, f_0 is the flux of reference, m_0 is the magnitude of reference (by default equals to zero) and m is the observed magnitude.

3.6 AsinhZeroPoint

Extension of zero point object to describe asinh magnitudes, a.k.a. luptitudes.

3.6.1 AsinhZeroPoint.softeningParameter: double

Parameter used to correct the calculation of magnitudes for faint sources. Usually called b. See [13] for a formal explanation.

Example:

Values used for SDSS DR5 asinh magnitudes:	
Band	Softening Parameters (b coefficients)
U	1.4×10^{-10}
G	0.9×10^{-10}
R	1.2×10^{-10}
I	1.8×10^{-10}
Z	7.4×10^{-10}

3.6.2 AsinhZeroPoint.getFluxFromMagnitude()

For asinh magnitudes, the operator to be used is:

$$f = f_0 . 10^{-(m-m_0)/2.5} \left[1 - b^2 . 10^{2.(m-m_0)/2.5} \right]$$

Where f is the flux of the observed source, f_0 is the zero point flux value, m is the magnitude assigned to this source, m_0 is the reference magnitude (default value to zero unless specified otherwise) and a new parameter appears, b, called the softening parameter which is referenced in this data model as the AsihnZeroPoint.softeningParameter.

3.6.3 AsinhZeroPoint.getMagnitudeFromFlux()

For asinh magnitudes, the operator to be used is:

$$m = m_0 - \frac{-2.5}{\ln(10)} \left[\sinh^{-1} \left(\frac{f}{2bf_0} \right) + \ln(b) \right]$$

Where m is the magnitude assigned to this source, m_0 is the reference magnitude (default value to zero unless specified otherwise) , f is the flux of the observed source, f_0 is the zero point flux value, and a new parameter appears, b, called the softening parameter, which is referenced in this data model as the AsihnZeroPoint.softeningParameter.

It can be seen that Pogson and Asinh magnitudes are the same if b=0 although, numerically it is recommended to use different equations to prevent infinites. See Appendix 5.2

3.7 LinearFluxZeroPoint

Extension of Zero Point to describe simple linear flux photometry, commonly used in Radio, Far Infrared and X-ray spectral ranges.

3.7.1 LinearFluxZeroPoint.getFluxFromMagnitude()

For Linear Flux magnitudes, conversion used would be a linear relation instead of a logarithmic one:

$$f = f_0 . \frac{m}{m_0}$$

Where f is the associated flux, f_0 is the flux of reference, m_0 is the magnitude of reference (default value to zero unless specified otherwise) and m is the observed magnitude.

3.7.2 LinearFluxZeroPoint.getMagnitudeFromFlux()

For Linear Flux magnitudes, linear conversion should be used to obtain the magnitudes:

$$m = m_0 . \frac{f}{f_0}$$

Where m is the observed magnitude, m_0 is the magnitude of reference (default value to zero unless specified otherwise), f is the associated flux and f_0 is the flux of reference.

3.8 MagnitudeSystem

The main difference between magnitude systems is the reference spectrum used to evaluate the magnitudes. In some occasions, the magnitude system will have a real spectrum of an existing source to calibrate all the magnitudes. In other occasions, a synthetic spectrum will be used.

3.8.1 MagnitudeSystem.type: String

Photometric system type used to calculate the associated zero point. Possible values are:

MagnitudeSystem Type
VEGAmag
ABMag
STMag

The list is not exhaustive. The principal difference between these photometric systems is the reference spectrum used to calculate the zero point. See section 2.2.3 for a detailed description.

3.8.2 MagnitudeSystem.referenceSpectrum: IVOA.SpectralDM.DataID.DataSetID

This describes the spectrum of an astronomical object used as reference to perform photometric calibration.

This points to a Spectrum object as defined in the IVOA spectrum data model []. Instead of having the whole spectrum attached, we define a link to it as referenceSpectrumURI.

This is a URL, pointing to a published IVOA resource location containing the reference spectrum used.

The value of this link can be computed or derived from the spectrum data model field spec:DataID.DataSetID for instance or re-use Curation.PublisherDID which is a unique identifier within the IVOA scope.

This mechanism offers a fully general representation of a magnitude system.

Some typical types of photometric systems are:

- o VEGmag: Makes use of Vega (αLyr) as the primary calibrating star. PhotometricSystem.referenceSpectrum would be the Vega SED

- o ABmag: Makes use of a reference spectrum of constant flux density per unit frequency:

$$f_{AB,\nu}^{0} = 3.631x10^{-20}\,erg\ s^{-1}\,cm^{-2}\,Hz^{-1}$$

- o STmag: Introduced for the HST project, it makes use of a reference spectrum of constant flux density per unit of wavelength:

$$f_{ST,\lambda}^{0} = 3.631x10^{-9}\,erg\ s^{-1}\,cm^{-2}\,\text{Å}^{-1}$$

4 Use Cases

4.1 Conversion from magnitude to flux, using a Filter Profile Service

The following fields are the minimal information needed in a DAL service response (SSAP or TAP) or into a serialization of the magnitude information in a catalog in order to allow the conversion from magnitudes to fluxes if a filter profile service is used:

- o It MUST have one field with Utype="**spec:Spectrum.Data.FluxAxis.Value**" and UCD="phot.mag" by measurement that includes the magnitude associated to this measurement.
Attributes to characterize the error of the measurement like spec:Spectrum.Data.FluxAxis.Accuracy.StatError,

spec:Spectrum.Data.FluxAxis.Accuracy.SysError, etc could also be present in the response.

- o It MUST have one field per catalog or measurement with utype="photdm:PhotCal.identifier" that includes the identifier within the filter profile service of the filter.

The normal workflow used by an application to do the conversion would be:
- o Go to the registry to obtain registration details of the Filter Profile service, using the IVOA identifier. In particular, the service URL of the service will be used to query this service using the uniqueIdentifier.
- o Query the Filter Profile Service to obtain basic information of this filter. This information would be, at least:

  - o photdm:PhotCal.ZeroPoint.zeroPointFlux.value
  - o photdm:PhotCal.ZeroPoint.zeroPointFlux.units
  - o photdm:PhotCal.ZeroPoint.type
  - o photdm:PhotCal.PhotometryFilter.SpectralAxis.Coverage.Location. Value

  And optionally, any other information that could be used for a better use of the selected data, as, e.g. the Photometry Filter related information.

Please notice that all the information of the Filter Profile Service can be overwritten either in the DAL service or in the data serialization. As an example, it could be decided that the zeroPointFlux to be used was not the general one for this filter within the filter profile service but the night one. In this case, this corrected value would appear in the DAL response or in the data serialization so this value, and not the one on the FPS will be used for the conversions.

Flux could be then calculated as (for Pogson magnitudes, i.e. Zeropoint.type=0 and reference magnitude = 0)

$$f = f^0 \, 10^{-m/2.5}$$

Where f^0 is the ZeroPoint.zeroPointFlux.value, m is the magnitude associated to the measurement and f is the associated flux. The type of flux (f_λ or f_ν) and the associated units, although they can be indirectly deduced from the field MagnitudeSystem, will be the same as those for the ZeroPoint.zeroPointFlux.

In case Zeropoint.type=1 (asinh magnitudes) the value of AsinhZeroPoint.softeningParameter.value should also be used to modify the conversion formula to:

$$f = f^0 10^{-m/2.5} \left[1 - b^2 . 10^{2.m/2.5} \right]$$

5 Appendix A

5.1 Zero point magnitude and zero point flux

The zero point flux can also be interpreted as a magnitude in the following way. Taking the previous equation and clearing the magnitude:

$$m = -2.5\log_{10}(f/f^0) = -2.5\log_{10}(f) + 2.5\log_{10}(f^0) = -2.5\log_{10}(f) + m_0$$

Where we have defined m_0, zero point magnitude, as the magnitude associated to the zero point flux:

$$m_0 = 2.5\log_{10}(f^0)$$

e.g. for ABmag photometric systems, the magnitudes are usually defined as:

$$m_{AB,v} = -2.5\log_{10}(f_v) - 48.6$$

Which is consistent with the definition of a zero point flux of the monochromatic flux:

$$f_{AB,v}^0 = 3.63 x 10^{-20} erg.s^{-1} cm^{-2} Hz^{-1}$$

As:

$$m_0 = 2.5\log_{10}(f_v^0) = 2.5\log_{10}(3.63 x 10^{-20}) = -48.6$$

Other systems usually define the zero point flux as a f_λ flux, as it is usually done by, e.g., STMag systems. For these systems, the reference flux would be a monochromatic f_λ flux:

$$f_{ST,\lambda}^0 = 3.631 x 10^{-9} erg\ s^{-1}\ cm^{-2}\ Å^{-1}$$

The usual definition of magnitudes for this photometric system is:

$$m_{ST,\lambda} = -2.5\log_{10}(f_\lambda) - 21.1$$

Which corresponds to, as in the previous example, a zero point flux of

$$f_{ST,\lambda}^0 = 3.631 x 10^{-9} erg\ s^{-1}\ cm^{-2}\ Å^{-1}$$

as:

$$m_0 = 2.5\log_{10}(f_\lambda^0) = 2.5\log_{10}(3.63 x 10^{-9}) = -21.1$$

In the present model and in order to provide a uniform treatment for all the different photometric systems, we have used the zero point flux as the quantity to characterize the photometry filter. The type of flux (f_ν or f_λ) and the units of any converted to flux magnitude would coincide with the ones used to express the zero point flux, i.e., the zero point flux contains information lost in the zero point magnitude.

5.2 Interrelation between Pogson and Asinh magnitudes

It can be proved that, if b=0, Pogson and Asinh magnitudes are the same:

$$\frac{f}{f_0} = 10^{-m/2.5}\left[1 - b^2 . 10^{2.m/2.5}\right]\big|_{b=0} = 10^{-m/2.5}$$

and

$$m = \frac{-2.5}{\ln(10)}\left[\sinh^{-1}\left(\frac{f}{2bf_0}\right) + \ln(b)\right]\big|_{b=0} =$$

$$\frac{-2.5}{\ln(10)}\left[\ln\left(\frac{f}{2bf_0} + \sqrt{1 + \frac{f^2}{4b^2 f_0^2}}\right) + \ln(b)\right]\big|_{b=0} =$$

$$\frac{-2.5}{\ln(10)}\left[\ln\left(\frac{f}{2f_0} + \sqrt{b^2 + \frac{f^2}{4f_0^2}}\right)\right]\big|_{b=0} =$$

$$\frac{-2.5}{\ln(10)}\left[\ln\left(\frac{f}{f_0}\right)\right] =$$

$$-2.5\log\left(\frac{f}{f_0}\right)$$

Although, as can be seen in the previous calculation, the use in code of the general asinh formula for both Pogson (b=0) and asinh (b>0) magnitudes is not recommended, as it could easily produce numerical infinites during the evaluation.

Appendix B: Data Model Summary

Utype	UCD+1	Meaning	Default value	Data type
Datamodel.name	meta.id	data model identification	PhotCalDM-v1.0	string
PhotometricSystem.description	meta.note	String representation Photometric system		string
PhotometricSystem.detectorType	meta.code	Type of detector (e.g energy or photon counter)	1	enum int
PhotometryFilter.identifer	meta.ref.ivorn	Unique identifer of filter within a Filter Profile Service (FPS)		string
PhotometryFilter.fpsIdentifier	meta.ref.ivorn	IVOA identifier of the Filter Profile Service		string
PhotometryFilter.filtername	meta.id;instr.filter	Filter Name in the instrumental configuration		string
PhotometryFilter.description	meta.note	Text description of the filter band		string
PhotometryFilter.transmissionCurve	meta.ref.ivorn	uri to the effective transmission curve		uri type
PhotometryFilter.bandName	instr.bandpass	Generic name for the filter spectral band		string
PhotometryFilter.SpectralAxis.unit	meta.unit	Unit of the spectral axis used to characterize it	angstrom	string
PhotometryFilter.SpectralAxis.UCD	meta.ucd	UCD for the nature of spectral axis wl, freq, energy	em.wl	string
PhotometryFilter.SpectralAxis.Coverage.Location.Value	em.wl;meta.main	Reference position along the spectral axis. Spectral coordinate of the Zero Point		string
PhotometryFilter.SpectralAxis.Coverage.Bounds.Extent	instr.bandwidth	Spectral axis extent of the filter		double
PhotometryFilter.SpectralAxis.Coverage.Bounds.Start	em.wl;start	Minimum value of the filter spectral coverage		double
PhotometryFilter.SpectralAxis.Coverage.Bounds.Stop	em.wl;.end	Maximum value of the filter spectral coverage		double

PhotometryFilter.TimeAxis.Coverage.Bounds.Start	time.start	Time stamp for Start of validity for this filter in ISOTime format		*string*
PhotometryFilter.TimeAxis.Coverage.Bounds.Stop	time.stop	Time stamp for Stop of validity for this filter in ISOTime format		*string*
PhotometryFilter.DateValidityFrom	time.start	Time stamp for Start of validity for this filter in ISOTime format		string
PhotometryFilter.DateValidityTo	time.end	Time stamp for Stop of validity for this filter in ISOTime format		string
PhotCal.identifier	meta.ref.ivorn	Unique identifier of the Photometry Calibration instance within a FPS		string
PhotCal.ZeroPoint.Flux.unit	meta.unit	unit for Zero point flux	Jy	string
PhotCal.ZeroPoint.Flux.ucd	meta.ucd	ucd for Zero point flux	phot.flux.density	string
PhotCal.ZeroPoint.Flux.value	phot.flux.density	flux value at Zero point associated to this filter		double
PhotCal.ZeroPoint.referenceMagnitude.unit	meta.unit	unit for reference magnitude	mag	string
PhotCal.ZeroPoint.referenceMagnitude.ucd	meta.ucd	ucd for reference magnitude	phot.mag	string
PhotCal.ZeroPoint.referenceMagnitude.value	phot.mag	Reference magnitude used for zero point	0.0	double
PhotCal.ZeroPoint.type	meta.code	Type of zero point	0.0	enum int
PhotCal.AsinhZeroPoint.softeningParameter	obs.param	Correction parameter for luptitudes	0	double
PhotCal.MagnitudeSystem.type	meta.code	Type of magnitude system	VEGAMag	string
PhotCal.MagnitudeSystem.ReferenceSpectrum.uri	meta.ref.ivorn	Reference SED or spectrum for this magnitude system		uri type

6 Appendix C: Data Model Serializations

6.1 Filter Profile Service Serialization

The following serialization is an example of a response of a filter profile service making use of the Photometry Filter DM through utypes:

```xml
<?xml version="1.0"?>
<VOTABLE version="1.1"
xsi:schemaLocation="http://www.ivoa.net/xml/VOTable/v1.1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <INFO name="QUERY_STATUS" value="OK"/>
  <RESOURCE type="results">
    <PARAM name="fpsID" utype=" photdm:PhotometryFilter.fpsIdentifier"
value="ivo://svo.cab/fps" datatype="char" arraysize="*"/>
    <PARAM name="filID" utype=" photdm:PhotometryFilter.identifer"
value="2MASS/2MASS.H" datatype="char" arraysize="*"/>
    <PARAM name="fname" utype=" photdm:PhotometryFilter.name"
value="2MASS.H" datatype="char" arraysize="*"/>

    <PARAM name="xunit"
utype="photdm:PhotometryFilter.SpectralAxis.Unit.expression"
value="Angstrom" datatype="char" arraysize="*"/>
    <PARAM name="xucd"
utype="photdm:PhotometryFilter.SpectralAxis.UCD" value="em.wl"
datatype="char" arraysize="*"/>
    <PARAM name="xmain"
utype="photdm:PhotometryFilter.SpectralAxis.Coverage.Location.Value"
value="16620" unit="Angstrom" datatype="float"/>
    <PARAM name="xwid"
utype="photdm:PhotometryFilter.SpectralAxis.Coverage.Bounds.Extent"
value="2509.40236716" datatype="float"/>
    <PARAM name="xmin"
utype="photdm:PhotometryFilter.SpectralAxis.Coverage.Bounds.Start"
value="15370" datatype="float"/>
    <PARAM name="xmax"
utype="photdm:PhotometryFilter.SpectralAxis.Coverage.Bounds.Stop"
value="17870" datatype="float"/>

    <PARAM name="detector"
utype="photdm:PhotometricSystem.detectorType" value="0" datatype="int"
/>

    <PARAM name="photcal" utype=" photdm:PhotCal.identifier"
value="2MASS/2MASS.H:Vega" datatype="char" arraysize="*"/>
    <PARAM name="magtype" utype="photdm:PhotCal.MagnitudeSystem.type"
value="VEGmag" datatype="int" />

    <PARAM name="zpvalue"
utype="photdm:PhotCal.ZeroPoint.zeroPointFlux.value" value="1024"
unit="Jy" datatype="float" />
```

```
    <PARAM name="zpunit"
utype="phfdm:PhotCal.ZeroPoint.zeroPointFlux.Unit.expression"
value="Jy" datatype="char" arraysize="*" />
    <PARAM name="zptype"  utype="photdm:PhotCal.ZeroPoint.type"
value="0" datatype="float" />

    <PARAM name="Facility" ucd="instr.obsty" value="2MASS"
datatype="char" arraysize="*"/>
    <PARAM name="ProfileReference"
value="http://www.ipac.caltech.edu/2mass/releases/allsky/doc/sec6_4a.ht
ml" datatype="char" arraysize="*"/>
    <PARAM name="CalibrationReference"
value="http://adsabs.harvard.edu/cgi-bin/nph-
bib_query?bibcode=2003AJ....126.1090C" datatype="char" arraysize="*"/>
<!—here the points of the response curve are stored directly in an
attached table➔
    <TABLE utype="photdm:PhotometryFilter.transmissionCurve.spectrum">
      <FIELD name="Wavelength" utype="spec:Data.SpectralAxis.Value"
ucd="em.wl" unit="Angstrom" datatype="float"/>
      <FIELD name="Transmission" utype="spec:Data.FluxAxis.Value"
ucd="phys.transmission" datatype="float"/>
      <DATA>
        <TABLEDATA>
          <TR>
            <TD>12890.0</TD>
            <TD>0.0000000000</TD>
          </TR>
          <TR>
            <TD>13150.0</TD>
            <TD>0.0000000000</TD>
          </TR>
          <TR>
            <TD>13410.0</TD>
            <TD>0.0000000000</TD>
          </TR>
          <TR>
            <TD>18930.0</TD>
            <TD>0.0000000000</TD>
          </TR>
          <TR>
            <TD>19140.0</TD>
            <TD>0.0000000000</TD>
          </TR>
        </TABLEDATA>
      </DATA>
    </TABLE>
  </RESOURCE>
</VOTABLE>
```

6.2 Photometric Data in Cone Search

Catalogs could include photometric measurements in some columns. In order to allow the publication of these measurements in a. e.g., cone search service, the creation of a new capability has been proposed.

The workflow to make use of this capability will be as follows:

- A cone search (or a future TAP service) will be registered with a certain agreed capability, e.g., Photometry.
- The response of this service will contains some VOTable groups that make use of Photometry, Spectral and Characterization data model utypes (it could also make use of links to a Filter Profile Service).
- Client applications able to process this photometric information will first look for services with this capability and make use of the information attached in the VOTable groups to handle it, e.g. by the conversion from magnitude to fluxes.

As an example, the serialization of the 2MASS catalog in a cone search service, could have the following information in the VOTable header:

```
<GROUP name="Flux1" ucd="phot.mag" utype="spec:Data.FluxAxis.Value">
<DESCRIPTION>2MASS J magnitude.</DESCRIPTION>

<PARAM name="ID" ucd="meta.id;instr.filter"
utype="photdm:PhotometryFilter.identifier"
unit="" datatype="char" arraysize="*" value="2MASS/2MASS.J" />

<PARAM name="WavelengthMean" ucd="em.wl.effective"
utype="photdm:PhotometryFilter.SpectralAxis.Coverage.Location.Value"
unit="Angstrom"
datatype="float" value="12410.5176673" />

<PARAM name="WavelengthMin" ucd="em.wl;stat.min"
utype="photdm:PhotometryFilter.SpectralAxis.Coverage.Bounds.Start"
unit="Angstrom"
datatype="float" value="10660" />

<PARAM name="WavelengthMax" ucd="em.wl;stat.max"
utype="photdm:PhotometryFilter.SpectralAxis.Coverage.Bounds.Stop"
unit="Angstrom"
datatype="float" value="14420" />

<PARAM name="PhotSystem" ucd=""
utype="photdm:PhotometricSystem.description" unit=""
datatype="char" arraysize="*" value="2MASS" />

<PARAM name="WidthEff" ucd="instr.bandwidth"
utype="photdm:PhotometryFilter.SpectralAxis.Coverage.Bounds.Extent"
unit="Angstrom"
datatype="float" value="1624.31986357" />

<PARAM name="ZeroPoint" ucd="phot.mag;arith.zp"
utype="photdm:PhotCal.ZeroPoint.zeroPointFlux"
unit="Jy" datatype="float" value="1614.45260952" />

<PARAM name="Description" ucd="meta.note"
```

```
utype="photdm:PhotometryFilter.description" unit="" datatype="char"
arraysize="*"
value="2MASS J" />

<PARAM name="TransmissionCurve" ucd="DATA_LINK"
utype="photdm:PhotometryFilter.transmissionCurve.uri" datatype="char"
arraysize="*"
value="http://svo.laeff.inta.es//theory/fr/fps.php?ID=2MASS/2MASS.J" />

<FIELDref ref="phot_m1" ucd="phot.mag;em.IR.J"
utype="spec:Data.FluxAxis.Value" />

<FIELDref ref="phot_e1" ucd="stat.error;phot.mag;em.IR.J"
utype="spec:Data.FluxAxis.Accuracy.StatError" />
</GROUP>
```

Exact details on how to serialize the response are contained in Derriere's IVOA note [14].

References

[1] **Bessell, M.S.** (2005) **Standard Photometric Systems**, Ann.Rev.Astron.Astrophys, 43:293-336

[2] **Longo G., Notes on photometric systems zero-points**, IVOA internal report (2008)

[3] **O'Connell, Magnitude and Color Systems**, ASTR 511/O'Connell Lec 14

[4] **Bruijne** (2003)**Stellar fluxes: transformations and calibrations**, ESA Tech. Note

[5] **Plante R et a**l (2007) **IVOA identifiers** http://www,ivoa.net/Documents/latest/IDs.html

[6] **Preite Martínez, A** (2007) **The UCD1+ Controlled Vocabulary** , http://www.ivoa.net/Documents/latest/UCDlist.html

[7] **Oke J.B., Wade R.A., A** (1982) **Spectrophotometric Survey of Cataclysmic Variable Stars,** The Astrophysical Journal, V87, N4, 670-679

[8] **Straizys V.** (1996) **The method of Synthetic Photometry**, Baltic Astronomy, V5, 459 476

[9] **Girardi et al** (2002) **Theoretical isochrones in several photometric systems,** A&A 391, 195-212

[10] **Maiz, J.** (2007) **A Uniform Set of Optical/NIR Photometric Zero Points to be Used with CHORIZOS**, ASP Conference Series, V999

[11] **Louys M. et al** (2011) **Observation Data Model Core Components and its Implementation in the Table Access Protocol v1.0,** http://www.ivoa.net/Documents/ObsCore/

[12] **Tody D. et al** (2011) **Simple Spectral Access Protocol (SSAP) v1.1**, http://www.ivoa.net/Documents/SSA/20110417/PR-SSA-1.1-20110417.pdf

[13] **Lupton, Robert H.;   Gunn, James E.;   Szalay, Alexander S.** (1999) **A Modified Magnitude System that Produces Well-Behaved Magnitudes, Colors, and Errors Even for Low Signal-to-Noise Ratio Measurements**, The Astrophysical Journal, V118, Issue 3, pp. 1406-1410

[14] **Derriere, S** (2010) **Providing Photometric Data Measurements Description in VOTables,** http://www.ivoa.net/internal/IVOA/PhotometryDataModel/NOTE-PPDMDesc-0.1-20101202.pdf