Mapping Imaging Metadata to CharDM

1.- Presenting a Tutorial
2.- UTYPE Issues

Alberto Micol

ESA/VO - European Space Astronomy Centre (ESAC)

Rationale

Personal obviously-proud opinion of the 4 fathers and 2 mothers for their child:

CharDM is quite well thought through.

It can cover, thanks to its generality, all kinds of products (provided their axes are not intertwined): spectra, images, 3d data cube, event lists, etc.

Could this generality be source of confusion?

How will a Data Provider find her way through the list of CharDM UTYPEs?

- *What's the difference between:*

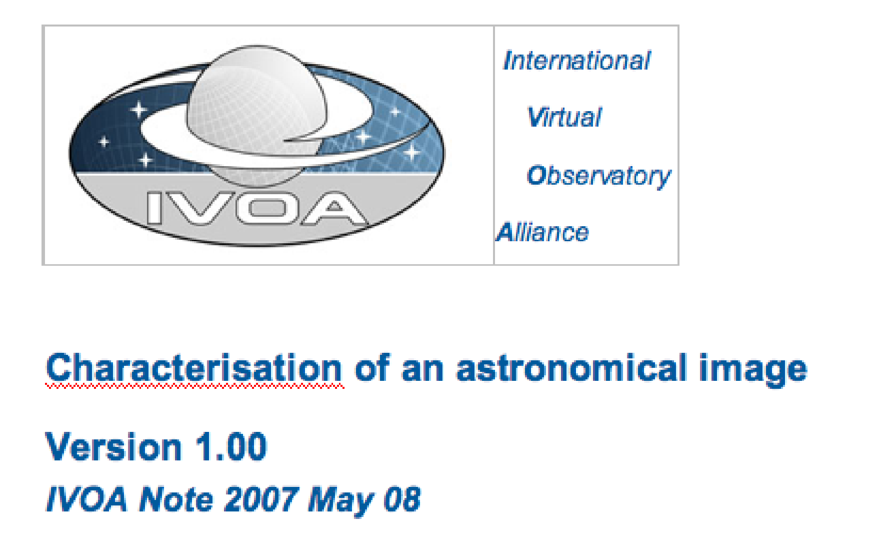
  timeAxis.coverage.bounds.extent and

  timeAxis.coverage.support.extent ?

Presenting a Tutorial

- ❑ IVOA Note associated to the release of Characterisation Data Model
- ❑ Purpose of the Tutorial

To illustrate how typical metadata pertaining to an astronomical image, being it product of an observation or of a simulation, can be mapped to elements of the Characterisation Data Model version 1.1 (CharDM)



International

Virtual

Observatory

Alliance

Characterisation of an astronomical image

Version 1.00

IVOA Note 2007 May 08

Presenting a Tutorial

- ❑ Motivations

By explaining the IVOA Characterisation Data Model concepts via a common and concrete example, we hope to help the data providers, the IVOA standard developers, and even ourselves, to understand the model, its completeness, and its usefulness.

Presenting a Tutorial

❑ The structure of the Tutorial

Introductory high level description of the CharDM

Organised by axis (spatial, temporal, spectral, observable), typical astronomical imaging metadata are considered

Mapping of relevant metadata to CharDM is provided in the form of UTYPEs

CharDM does not describe:
- instrumental setups (eg, amplifier gains, filter names)
- metadata pertaining to the sources observers were interested in (Provenance)

CharDM describes the N-dimensional space a data product is immersed in: Coverage, Resolution, and Sampling, with their accuracies.

Presenting a Tutorial

- On each axis, very symmetrically, the various typical quantities:

1 Introduction
2 The Characterisation Data Model in few words
3 The Spatial Axis
3.1 Spatial units
3.2 Coordinate system
3.3 Where on the sky?
3.4 Astrometric quality
3.5 Astrometric accuracy
3.6 Spatial resolution
3.7 Pixels: spatial sampling
4 The Spectral Axis
4.1 Spectral units
4.2 Coodinate system: where on the spectrum?
4.3 Wavelengths (Frequencies, Energies): spectral coordinates
4.4 Wavelength calibration quality: spectral axis calibration
4.5 Errors on wavelength calibration: spectral axis accuracy
4.6 Spectral resolution
4.7 Bin size: spectral sampling
… etc …

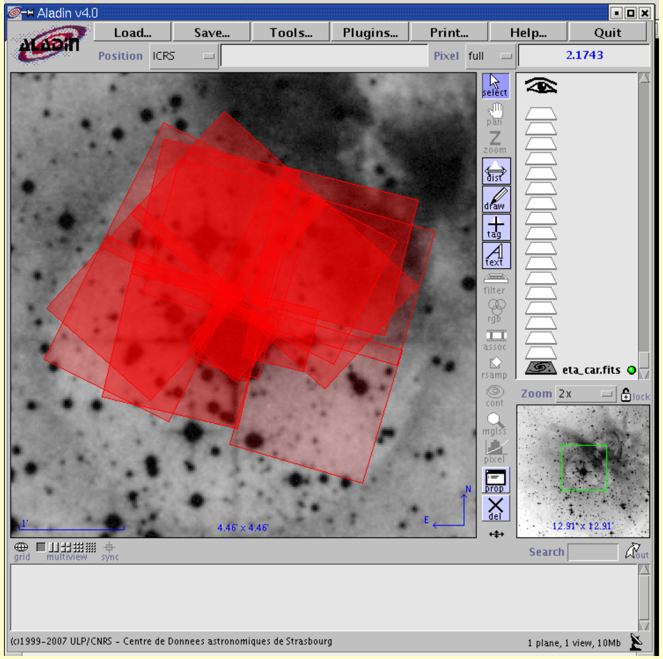
Presenting a Tutorial

- ❑ An example:

Cha

3 The Spatial Axis

3.3 Where on the sky?



Concept described in Data Providers terminology

The footprints […]

UTYPE mapping provided

The place where this is described in the CharDM is:

spatialAxis.coverage.support.area[] ***[R]***

Compliancy status

The ending square brackets are used to indicate that a number of parameters are to be provided in this case. (Specifically, the serialization of such element should obey to the STC:AstroCoordArea Region(2D) polygon specification).

Serialisation

Data Discovery and Serialisation

❑ Main goals of this version of CharDM:

- Data Discovery
- Serialisation

Other goals (Data Analysis) are only mildly touched upon (left to future version).

Data Discovery

- CharDM provides UTYPEs for all its described quantities
  - Examples:
    - Spatial resolution:

      spatialAxis.resolution.resolutionRefVal.ReferenceValue

    - Pixel scale:

      spatialAxis.samplingPrecision.samplingRefVal.samplingPeriod

    - Time of the observation (indicative time, loosely defined):

      timeAxis.coverage.location.coord

    - Astrometric accuracy:

      spatialAxis.accuracy.statError.ErrorRefVal

      including a value (.errorRefValue) and its documentation (.documentation)

    - Start and Stop times:

      timeAxis.coverage.bounds.limits

Data Discovery

❑ CharDM provides UTYPEs for all its described quantities

- Units and Coordinate Systems:
  - Time of the observation (indicative, loosely defined):

    timeAxis.coverage.location.coord

  - In which units and reference frame?

timeAxis.coverage.location.unit
timeAxis.coverage.location.coordsystem

timeAxis.coverage.unit
timeAxis.coverage.coordsystem

timeAxis.unit
timeAxis.coordsystem

STC Coordinate Systems, e.g.:

TT-ICRS-WAVELENGTH-TOPO

Serialisation

- ❑ Start and Stop times: timeAxis.coverage.bounds.limits → STC

Cha

```
<timeAxis>
    <coverage>
        <bounds>
            <limits coord_system_id="TT-ICRS-WAVELENGTH-TOPO">
                <stc:StartTime>
                    <stc:TimeInstant>
                        <stc:ISOTime>2000-12-02T17:59:44.00</stc:ISOTime>
                    </stc:TimeInstant>
                </stc:StartTime>
                <stc:StopTime>
                    <stc:TimeInstant>
                        <stc:ISOTime>2000-12-02T17:59:44.00</stc:ISOTime>
                    </stc:TimeInstant>
                </stc:StopTime>
            </limits>
```

Serialisation example by I.Chilingarian

Issues [1/4]

❑ Start and Stop times: timeAxis.coverage.bounds.limits → STC

```
<timeAxis>
    <coverage>
        <bounds>
            <limits coord_system_id="TT-ICRS-WAVELENGTH-TOPO">
                  <stc:StartTime>
                       <stc:TimeInstant> What it the UTYPE of this guy?
                             <stc:ISOTime>2000-12-02T17:59:44.00</stc:ISOTime>
                       </stc:TimeInstant>
                  </stc:StartTime>
                  <stc:StopTime>
                      <stc:TimeInstant>
                              <stc:ISOTime>2000-12-02T17:59:44.00</stc:ISOTime>
                       </stc:TimeInstant>
                  </stc:StopTime>
             </limits>
```

Issues [2/4]

Cha

❑ How to refer to just only Start Time?

- Using an ugly concatenation of elements with their namespaces?

  e.g.: timeAxis.coverage.bounds.limits.stc:StartTime ? Or even:

  timeAxis.coverage.bounds.limits.stc:StartTime.stc:TimeInstant.stc:ISOTime

  as in a XPATH from the previous slide (with '.' instead of '/')?

  Shouldn't CharDM have its own set of UTPYEs?

  And more generically…

❑ How can a DM re-use elements defined by some other DM?

How to formulate UTYPEs for such inherited elements?

(Do VOEvent people know the answer?

what are their UTYPEs when STC involved?)

How will VOQL make use of such UTYPEs?

Issues (more specific to CharDM) [3/4]

□ How open to interpretations are the elements of CharDM?

- Example: Is the fluxAxis.coverage.bounds.limits element for
  - Min and Max of flux levels?
  - Min is the estimated noise level and Max (if any) is saturation point?

- Can the SNR be provided using:
  - fluxAxis.coverage.location.coord and
  - fluxAxis.accuracy.statError.errorRefVal.errorRefValue

- What’s the difference between:
  timeAxis.coverage.bounds.extent and
  timeAxis.coverage.support.extent ?

How to ensure that all Data Providers use the same UTYPE for the same quantity?

Issues (more specific to CharDM) [4/4]

Cha

□ GENERIC vs HARDCODED Axes?

- spatialAxis, spectralAxis, timeAxis: HARDCODED names
- CharDM wishes to describe all kinds of axes, e.g. the observable (flux, polarization), or other independent axes, e.g. radial velocity, etc.

- Take the FLUX of a 2D image, should it be:
  - Hardcoded (I used fluxAxis in the tutorial)
  - Generic, distinguishable via its UCD ChAxis.ucd = phot

□ Is it premature to provide list of UTYPEs with the current version of the DM(s)?
Should we publish CharDM without UTYPEs (temporarily)?
(but: isn't a UTYPE-less DM like the Bible without the 10 commands?)

7) We definitively need a UTYPE discussion involving all WGs.