

Meeting with the VizieR team — 02 FEB 2018

Present: Sebastien, Gilles, Francois, Mireille, Laurent, Ada

The serialisation from Francois was presented in a detailed way.

- The fact that the serialisation is shown in a four table form looks a bit complicated for VizieR, and it should be easier using params.

```
- <VOTABLE version="1.2">
  - <RESOURCE type="results">
    + <TABLE ID="ndgnsolidgdea" utype="ts:TimeSeries"></TABLE>
    + <TABLE ID="char" name="characterisation" utype="cha:Char"></TABLE>
    + <TABLE ID="coosys" name="coordsys" utype="coord:coordsys"></TABLE>
    + <TABLE ID="data"></TABLE>
  </RESOURCE>
</VOTABLE>
```

- It was decided to show it like that since applications don't show params but this can be changed so as to have a unique table. It's a "visualisation" issue.
- 1st table — changes:
 - <TABLE ID="ndgnsolidgdea" utype="ts:TimeSeries"></TABLE> —>
 <RESOURCE type="results"> utype="ts:rootclass".
 - Like this it should be easier for Apps to know what to do afterwards with the other tables
 - Resource can have a Type attribute. This is possible according to VOTable1.3. Needs to be coherent with DataSetDM. We want to say that it is a root class of the model Time series: ts:RootClass.
- 2nd table — changes:
 - utype="cha:Char.SpatialAxis.Location.Coord.SpatialValue2D[0]" —> **ts:char**
 - utype="cha:Char.SpatialAxis.Location.Coord.SpatialValue2D[1]" —> ts:char
 - unit="deg" utype="cha:Char.SpatialAxis.Bounds.CharBox.Size2[0]" —> ts:char
 - unit="deg" utype="cha:Char.SpatialAxis.Bounds.CharBox.Size2[1]" —> ts:char
 - Characterisation is not complete: add spectral, and temporal information about the coverage.
 - Add **target** as in Obscore
 - This is compatible with ObsCore.
- 3rd table — changes:

```
<FIELD ID="TimeScale" name="TimeScale" ucd="time" utype="coord:coordsys.TimeFrame.TimeScale" datatype="char" arraysize="*" />
<FIELD ID="refPositionT" name="refPositionT" ucd="pos" utype="coord:coordsys.TimeFrame.refPosition" datatype="char" arraysize="*" />
<FIELD ID="SpaceReFrame" name="SpaceReFrame" ucd="pos" utype="coord:coordsys.SpaceFrame.spaceReFrame" datatype="char" arraysize="*" />
<FIELD ID="refPositionS" name="refPositionS" ucd="pos" utype="coord:coordsys.SpaceFrame.refPosition" datatype="char" arraysize="*" />
```

- Add the possibility of having an **offset**. Relative time.
 - refPositionT =NULL if not filled? Not sure this should be mandatory
 - Take an example — sent light curve of a source to Francois.
- 4th table — data itself — nothing to comment.
- There was no time to show the TAP-Schema.

Comparison between the proposed serialisations by Jiri, Francois and Laurent

The serialisation from Laurent was presented in a detailed way too.

```
<TEMPLATES tblref="ndgnsolidgdea">
  <TUPLE dmrole='root'>
    <TUPLE dmrole='timeseries:TimeSerie.dataSet'>
      <VALUE dmrole='timeseries:dataset.DataSet.calib_level' source='@calibLevel' />
      <VALUE dmrole='timeseries:dataset.DataSet.creator' source='@creat' />
      <VALUE dmrole='timeseries:dataset.DataSet.contributor' source='@cont' />
      <VALUE dmrole='timeseries:dataset.DataSet.publisher_did' source='@pubDID' />
      <VALUE dmrole='timeseries:dataset.DataSet.target' source='@targ' />
    </TUPLE>

    <TUPLE dmrole='timeseries:TimeSerie.spaceFrame' tblref='coosys' />
    <TUPLE dmrole='timeseries:TimeSerie.timeFrame' tblref='coosys' />
    <TUPLE dmrole='timeseries:TimeSerie.filter' tblref='coosys' />
    <TUPLE dmrole='timeseries:TimeSerie.refPosition' tblref='char' />

    <TUPLE dmrole='timeseries:TimeSerie.dependantModelDescriptor'>
      <VALUE dmrole='timeseries:dataset.DependantModelDescriptor.name' source='child'>Laurent</VALUE>
      <VALUE dmrole='timeseries:dataset.DependantModelDescriptor.ivoid' source='child'>ivo://vodml/lite/laurent</VALUE>
      <VALUE dmrole='timeseries:dataset.DependantModelDescriptor.url' source='child'>http://vodml/lite/laurent</VALUE>
    </TUPLE>

    <COLLECTION dmrole="timeseries:TimeSerie.pointsJ">
      <TUPLE dmrole="timeseries:data.PointJ" tblref="data" />
    </COLLECTION>
    <COLLECTION dmrole="timeseries:TimeSerie.pointsH">
      <TUPLE dmrole="timeseries:data.PointH" tblref="data" />
    </COLLECTION>
    <COLLECTION dmrole="timeseries:TimeSerie.pointsK">
      <TUPLE dmrole="timeseries:data.PointK" tblref="data" />
    </COLLECTION>
    <COLLECTION dmrole="timeseries:TimeSerie.pointsL">
      <TUPLE dmrole="timeseries:data.PointL" tblref="data" />
    </COLLECTION>
    <COLLECTION dmrole="timeseries:TimeSerie.pointsM">
      <TUPLE dmrole="timeseries:data.PointM" tblref="data" />
    </COLLECTION>
  </TUPLE>
</TEMPLATES>
```

- It's XML
- Pointers within a class of a model to columns in existing tables.
- Python API (prototype):
 - Parset init
 - Data extraction: the client has just to know a few keys pointing to real data
 - Plot
- See transparencies with the explanation of the serialisation and the python code.
- Template here is specific to each type of data, but so it is the TAP-Schema.

Conclusions:

- Showed Francois serialisation. Modifications have been proposed by Vizier. There was no time to show the TAP-Schema.
- Laurent's proposal seems to be clear and simple from Vizier's point of view. But need to see details in particular the Python API. Should be ok to write it in java.
- There was no time to see Jiri's. This needs to be done too.
- Input from Apps (Pierre's input/view?)
- Input from other projects (Gaia, Spanish VO, Marco).