



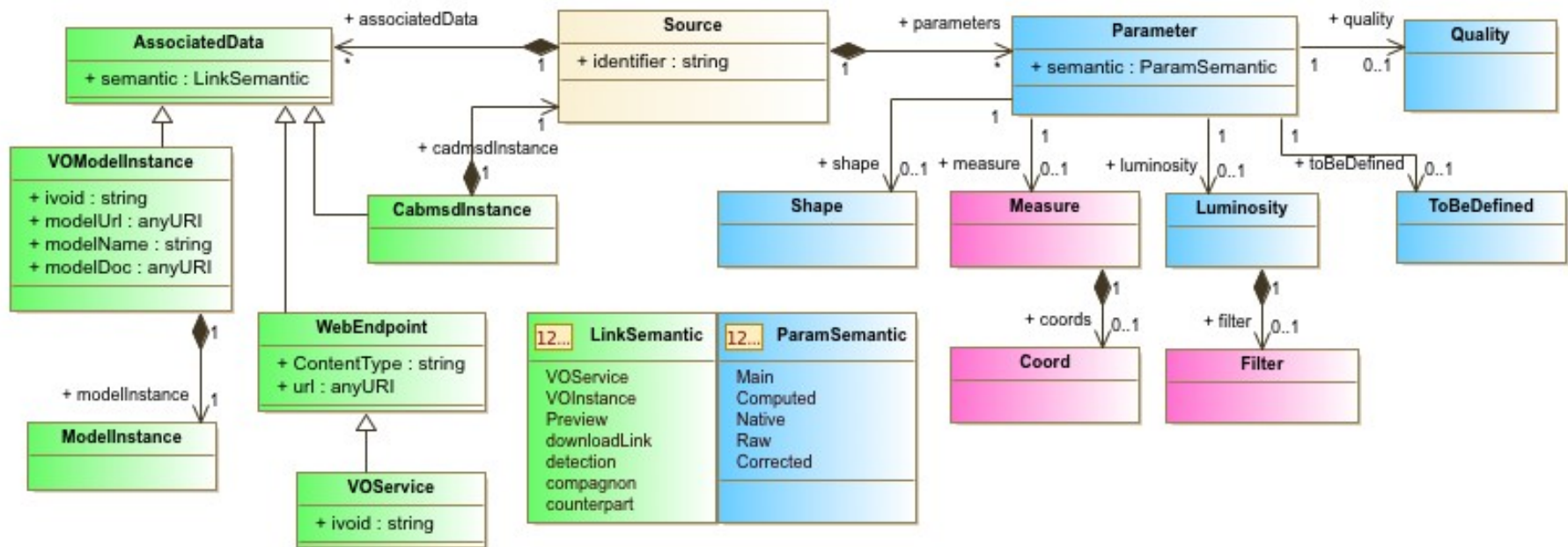
---

# Annotating ZTF data (TimeSeries) with CAB-MSD. DM in TAP

F.Bonnarel, L.Michel, A.Nebot, M.Louys,  
G.Landais



# □ CAB-MDS DataModel (see next talk)





# “VOData service “tableset” data model mapping



- Reuse some possibilities of a well established VO standard
- Create a VOData service « tableset » (=heart of a TAP schema)
  - table with type =« base\_table» describing the native VOTables
  - tables (hereafter « virtual » tables) mapping the model structure.

# “VODataservice “tableset” data model mapping



```
vosi:tableset~
  xmlns:vosi="http://www.ivoa.net/xml/VOSITables/v1.0"~
  xmlns:vod="http://www.ivoa.net/xml/VODataService/v1.1"~
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ivoa.net/xml/VODa
<schema>~
  <name>ZPN-native-annotated</name>~
  <description>ZPN native tableset + views for CAB-SDM + meas_set annotation</description>~
  <table type="base_table">~
    <name>ZPN-native-table</name>~
    <description>>ZPN Native table</description>~
    <column>~
      <name>objectid</name>~
      <description>document internal source id</description>~
      <dataType xsi:type="vod:TAPType">INTEGER</dataType>~
      <ucd>meta.id</ucd>~
    </column>~
    <column>~
      <name>oid</name>~
      <description>published source id</description>~
      <dataType xsi:type="vod:TAPType">BIGINT</dataType>~
      <ucd>meta.id</ucd>~
    </column>~
    <column>~
      <name>expid</name>~
      <dataType xsi:type="vod:TAPType">INTEGER</dataType>~
      <ucd>meta.id;obs-exposure</ucd>~
    </column>~
    <column>~
      <name>hjd</name>~
      <dataType xsi:type="vod:TAPType">DOUBLE</dataType>~
      <ucd>time.epoch</ucd>~
```

# “VODataService “tableset” data model mapping



```
<table type="view">~
  <name>meas_set:TimeSeries</name>~
  <column>~
    <name>vomodelInstance</name>~
    <utype>cab-sdm:VOModelInstance.ModelInstance</utype>~
  </column>~
  <column>~
    <name>mjd</name>~
    <utype>meas_set:Point;meas:Time;coords:TimeStamp.date</utype>~
  </column>~
  <column>~
    <name>hjd</name>~
    <utype>meas_set:Point;meas:Time;coords:TimeStamp.date</utype>~
  </column>~
  <column>~
    <name>mag</name>~
    <utype>meas_set:Point;cab-sdm:Luminosity.value</utype>~
  </column>~
  <column>~
    <name>magerr</name>~
    <utype>meas_set:Point;cab-sdm:Luminosity.error</utype>~
  </column>~
  <foreignKey>~
    <targetTable>ZTF-native-table</targetTable>~
    <fkColumn>~
      <fromColumn>mjd</fromColumn>~
      <targetColumn>mjd</targetColumn>~
    </fkColumn>~
  </foreignKey>~
  <foreignKey>~
    <targetTable>ZTF-native-table</targetTable>~
    <fkColumn>~
      <fromColumn>hjd</fromColumn>~
      <targetColumn>hjd</targetColumn>~
    </fkColumn>~
  </foreignKey>~
  <foreignKey>~
    <targetTable>ZTF-native-table</targetTable>~
    <fkColumn>~
      <fromColumn>mag</fromColumn>~
      <targetColumn>mag</targetColumn>~
    </fkColumn>~
  </foreignKey>~
  <foreignKey>~
    <targetTable>ZTF-native-table</targetTable>~
    <fkColumn>~
      <fromColumn>magerr</fromColumn>~
      <targetColumn>magerr</targetColumn>~
    </fkColumn>~
  </foreignKey>~
  <foreignKey>~
    <targetTable>TimeFrame</targetTable>~
    <fkColumn>~
      <fromColumn>mjd</fromColumn>~
      <targetColumn>frameid</targetColumn>~
    </fkColumn>~
  </foreignKey>~
  <foreignKey>~
    <targetTable>TimeFrame-HELIO</targetTable>~
    <fkColumn>~
      <fromColumn>hjd</fromColumn>~
      <targetColumn>frameid</targetColumn>~
    </fkColumn>~
  </foreignKey>~
  </foreignKey>~
</table>~
```



# “VOData service “tableset” data model mapping



- Rule to define the virtual table applied to the main table :
  - identify the columns associated to cab-msd instance attributes or VOInstance model
  - create virtual tables with virtual columns related to native columns by foreign keys.
  - complete the virtual table with unused model attributes. (will be PARAMS in the serialisation)
  - put vo-dml.id into utype in the simple case
  - in case of 1 to 1 references : replace hierachy by a flat table going to the leave
  - replace any “ivoa type”, ucd, unit attribute in the model by corresponding element inside the « column » element.

# □ “FLAT” utypes generation



- **Derive** « flat » utypes from the tableset this way :
- **Look** if the FIELD is referred directly in one of the « view » tables and if yes :
  - → **add** the utype of this « view » column to the utype of the actual column
- **Look** if the « view » table is referred from another « view » table column
  - → **Concatenate** the utype of the referring column before the result table
- **Iterate** the last step until we reach to top view class

# □ “FLAT” utypes generation



- `<FIELD name="ra" datatype="double" ucd="pos.eq.ra" unit="deg"/>`
  - utype = « meas:EquatorialPosition.ra »
  - utype = « cab-msd:Source/meas:EquatorialPosition.ra »

- So now we have :  
`<FIELD name="ra" datatype="double" ucd="pos.eq.ra" unit="deg" utype = « cab-msd:Source/meas:EquatorialPosition.ra » />`

- In the same way

`<FIELD name="hjd" datatype="double" ucd="time.epoch" unit="d">`

→ utype = « meas\_set:Point.meas:Time.coords:TimeStamp.date »

→ utype = « cab-msd:VOModelInstance.modelInstance/meas\_set:Point/meas:Time/coords:TimeStamp.date »

→ utype = « cab-msd:Source.VOModelInstance.modelInstance/meas\_set:Point/meas:Time/coords:TimeStamp.date »

`<FIELD name="hjd" datatype="double" ucd="time.epoch" unit="d" utype = « cab-msd:Source.VOModelInstance.modelInstance/meas_set:Point/meas:Time/coords:TimeStamp.date » />`



# □ “FLAT” utypes generation



```
<TABLE>
<FIELD name="oid" datatype="long" ucd="meta.id" utype="cab-msd:Source.identifier" >
  <DESCRIPTION>Object ID</DESCRIPTION>
</FIELD>
<FIELD name="expid" datatype="int" ucd="meta.id;obs.exposure" utype="cab-msd:Source.Parameter">
  <DESCRIPTION>Exposure ID</DESCRIPTION>
</FIELD>
<FIELD name="hjd" datatype="double" ucd="time.epoch" unit="d" utype = "cab-msd:VOModelInstance.modelInstance/meas_set:Point/meas:Time/coords:TimeStamp.date" ref="hjtime" >
  <DESCRIPTION>Heliocentric Julian date (computed from mjd and the mean ra and dec of the input catalog)</DESCRIPTION>
</FIELD>
<FIELD name="mjd" datatype="double" ucd="time.epoch;obs.exposure" unit="d" utype = "cab-msd:VOModelInstance.modelInstance/meas_set:Point/meas:Time/coords:TimeStamp.date" ref="time" >
  <DESCRIPTION>Modified Julian date</DESCRIPTION>
</FIELD>
<FIELD name="mag" datatype="float" ucd="phot.mag;em.opt" unit="mag" utype = "cab-msd:VOModelInstance.modelInstance/meas_set:Point/cab-msd:Luminosity.value">
  <DESCRIPTION>Magnitude</DESCRIPTION>
</FIELD>
<FIELD name="magerr" datatype="float" ucd="stat.error;phot.mag;em.opt" unit="mag" utype = "cab-msd:VOModelInstance.modelInstance/meas_set:Point/cab-msd:Luminosity.error">
  <DESCRIPTION>Uncertainty in mag measurement. Includes correction to conform to photometric repeatability RMS derived from &quot;non-variable&quot; population.</DESCRIPTION>
</FIELD>
<FIELD name="catflags" datatype="int" ucd="meta.code" utype="cab-msd:Source.Parameter">
  <DESCRIPTION>Catalog flags for source from PSF-fitting catalog</DESCRIPTION>
</FIELD>
<FIELD name="filtercode" datatype="char" arraysize="*" ucd="instr.bandpass" utype="cab-msd:Source.Parameter">
  <DESCRIPTION>Filter code (abbreviated name)</DESCRIPTION>
</FIELD>
<FIELD name="ra" datatype="double" ucd="pos.eq.ra" unit="deg" utype = "cab-msd:Source/meas:EquatorialPosition.ra" ref="ICRS" >
  <DESCRIPTION>Right Ascension of source</DESCRIPTION>
</FIELD>
<FIELD name="dec" datatype="double" ucd="pos.eq.dec" unit="deg" utype = "cab-msd:Source/meas:EquatorialPosition.dec" ref="ICRS" >
  <DESCRIPTION>Declination of source</DESCRIPTION>
</FIELD>
<FIELD name="chi" datatype="float" ucd="stat.parameter" utype="cab-msd:Source.Parameter">
  <DESCRIPTION>Chi-squared of source</DESCRIPTION>
</FIELD>
<FIELD name="sharp" datatype="float" ucd="stat.parameter" utype="cab-msd:Source.Parameter">
  <DESCRIPTION>Sharpness of source</DESCRIPTION>
</FIELD>
<FIELD name="filefracday" datatype="long" ucd="time.epoch;obs.exposure" utype="cab-msd:Source.Parameter">
  <DESCRIPTION>Exposure file timestamp, with decimal representation YYYYMMDDdddd: year, month, day, and fractional day</DESCRIPTION>
</FIELD>
<FIELD name="field" datatype="int" ucd="meta.id;obs.field" utype="cab-msd:Source.Parameter">
  <DESCRIPTION>Field ID</DESCRIPTION>
</FIELD>
<FIELD name="ccdId" datatype="unsignedByte" ucd="meta.id;instr.det" utype="cab-msd:Source.Parameter">
  <DESCRIPTION>CCD number (1..16)</DESCRIPTION>
</FIELD>
```

# □ “FLAT” utype



- Pros :
  - no structure to read before the table
  - Simple pointer to the structure in tableset (→ model)
- Cons :
  - strings may be long
  - difficult to distinguish columns with similar utypes
  - « association » of columns can be made by refs :  
But refs may have several other meanings and are unique
  - no easy “group of rows” to define instances (= no filter or grouping facility).

# □ GROUP serialisation



- **Derive** GROUPs from the tableset this way :
- **Look** if the FIELD is referred directly in one of the « view » tables and if yes :
- → **create** a GROUP with the virtual table utype, create a FIELDref inside and use the column utype of this « view » column for the FIELDref
- **Look** if the « view » table is referred from another « view » table column
- → **create** an upper level GROUP containing the first one.
- **Add** appropriate PARAMS and FIELDref inside.
- **Iterate** the last step until we reach to top view class

# GROUP serialisation



```
<TIMESYS ID="time" refposition="BARYCENTER" timeorigin="" timescale="TCB"/>
<TIMESYS ID="htime" refposition="HELIOCENTER" timeorigin="" timescale="TCB"/>
<COOSYS ID="ICRS" system="ICRS" epoch="J2015"/>
<GROUP utype="cab-msd:Source" />
  <FIELDref utype="cab-msd:Source.identifier" ref="oid"/>
  <GROUP utype="cab-msd:VOModelInstance">
    <PARAM utype="cab-msd:VOModelInstance.vodmlid" value="adhoc://meas_set" />
    <PARAM utype="cab-msd:VOModelInstance.modelName" value="Measures Set" />
    <PARAM utype="cab-msd:VOModelInstance.loc" value="http://adhoc.doc.html" />
    <GROUP utype="meas_set:Point">
      <GROUP utype="meas:Time;coords:TimeStamp" ref="htime">
        <FIELDref "meas:Time;coords:TimeStamp.date" ref="hjd"/>
      </GROUP>
      <GROUP utype="meas:Time;coords:TimeStamp" ref="time">
        <FIELDref "meas:Time;coords:TimeStamp.date" ref="mjd"/>
      </GROUP>
      <GROUP utype="cab-msd:Luminosity" >
        <FIELDref "cab-msd:Luminosity.value" ref="mag"/>
        <FIELDref "cab-msd:Luminosity.error" ref="magerr"/>
      </GROUP>
    </GROUP>
  </GROUP>
  <GROUP utype="meas:EquatorialPosition">
    <FIELDref utype="meas:EquatorialPosition.ra" ref="ra"/>
    <FIELDref utype="meas:EquatorialPosition.dec" ref="dec"/>
  </GROUP>
  <FIELDref utype="Source.Parameter" ref="expid" />
  <FIELDref utype="Source.Parameter" ref="catflags" />
  <FIELDref utype="Source.Parameter" ref="filtercode" />
  <FIELDref utype="Source.Parameter" ref="chi" />
  <FIELDref utype="Source.Parameter" ref="sharp" />
  <FIELDref utype="Source.Parameter" ref="ccd" />
  <FIELDref utype="Source.Parameter" ref="field" />
</GROUP>
```

# □ GROUP serialisation



- Pros :
  - no change to the main table
  - association between column unambiguous
- Cons :
  - structure to parse
  - no easy “group of rows” to define instances (= no filter or grouping facility).

# □ VODML-“like” serialisation



- From Tableset to VODML-“like” (Laurent Michel proposal)
  - 1 ) Transform attributes of the FIELD into an ivoa:Quantity :

```
<INSTANCE dm-role= »utype » dm-type= « ivoa:?Quantity » />
```

```
<VALUE dmrole="ivoa:Quantity.unit" dmtype="ivoa:Unit" value=" ?"/>
```

```
<VALUE dmrole="ivoa:?Quantity.value" dmtype="ivoa:?" ref=" ?"/>
```

```
</INSTANCE>
```

- 2 ) Transform each column into an INSTANCE level.
  - Dm-role is the utype of the level
  - Dm-type is deduced from lower level Dm-role eg :
    - lower level : dm-role: TimeStamp.date
    - higher level : dm-type : TimeStamp

# □ VODML-“like” serialisation



- From Tableset to VODML-Lite
  - 3 ) Create a COMPOSITION for each table
  - 4 ) create references when tableset contains reference columns
  - 4 ) When two tables have the same column content create a single INSTANCE with a FILTER
  - 5 ) And again up to the top

# □ VODML-“like” serialisation



- Pro :
  - Full model structure available :
    - distinction composition/instance
    - dm-types and dm-roles
  - Possibility of addressing rows in the table :
    - allow references to lines in existing tables
  - No change in native VOTABLE
- Cons :
  - Lot of levels (less human readable)
  - The inverse process doesn't reproduce easily the “native VODML-Lite” transcription of vo-dml-xml.



# □ Conclusions



- Flat utypes and groups are easy to generate from the tableset
- VO-dml-like generation from the tableset maybe sometime difficult but his richer in modern content
- The fact that tableset can be derived from VO-DML-Like makes it compatible with TAP. → virtutal Tableset proposal maybe part of the TAP schema