A Component and Association Based Model For Source Data

L. Michel F. bonnarel in collaboration with TDIG and CDS
and others VO members

Why Doing a Source Model?

- **For the Science**
  - Make sure that data, scientifically relevant for a particular use case, are well described.

- **To make sure that data sent by a service will be properly understood by the clients.**
  - With a system of unambiguous annotations
  - By enabling clients to understand these annotations
  - By making sure that clients could take advantage of these annotations

The model design is not a goal in itself,
it is just a (powerfull) tool

J/A+A/532/A103/IC4665 Photometry and proper motions in IC4665 (Lodieu+, 2011)   2011A&A
Post annotation   Coordinates, photometry, proper motions, and physical parameters for IC4665 stars *(tables A1, B1 and C1 of paper) (1533 rows)*

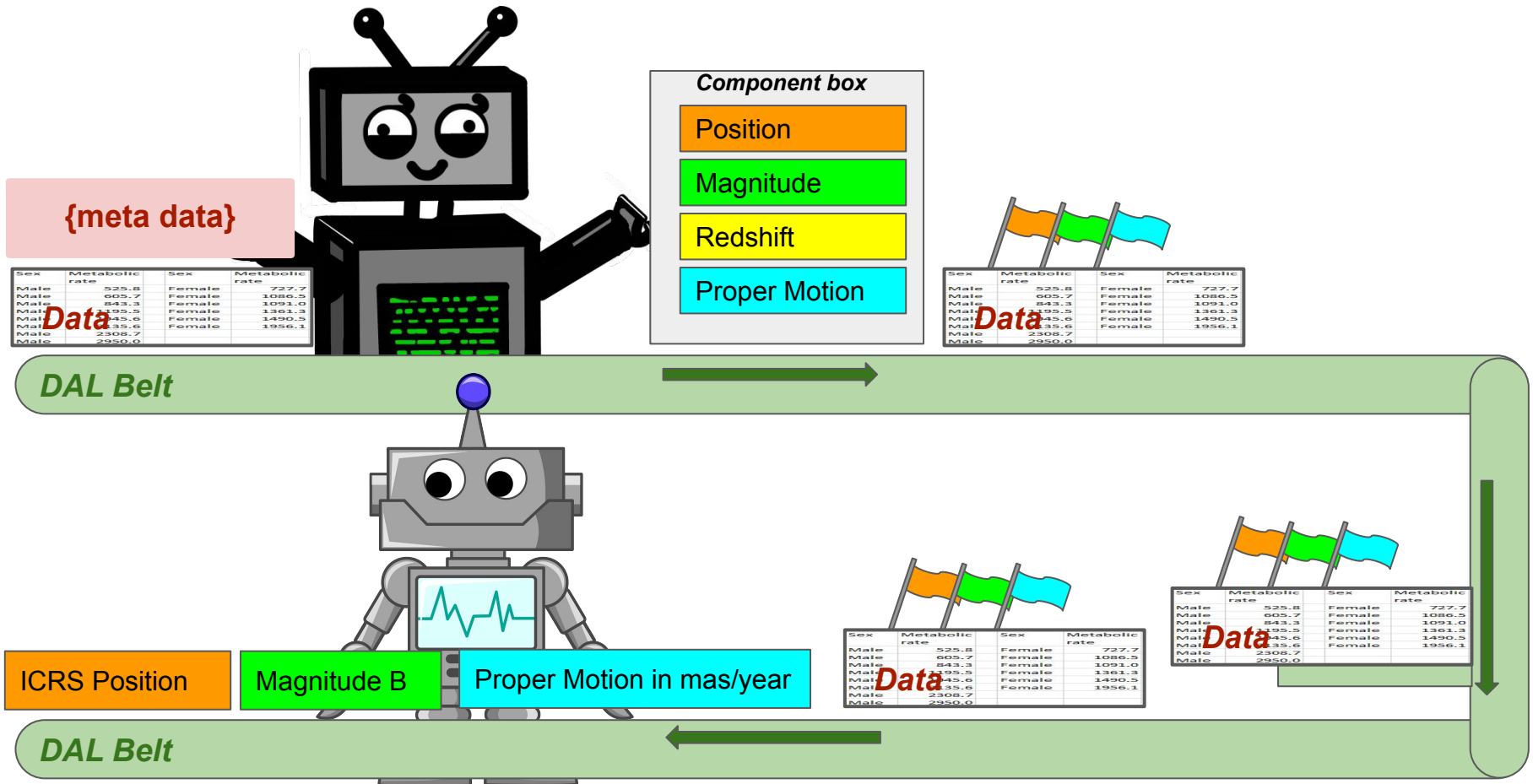start AladinLite   plot the output   query using TAP/SQL

Full	RAJ2000 "h:m:s"	DEJ2000 "d:m:s"	Mm	Name	Zmag mag	Ymag mag	Jmag mag	Hmag mag	Kmag mag	pmRA mas/yr	pmDE mas/yr	logL [Lsun]	Mass Msun	GCS	Simbad	RAJ2000 "h:m:s"	DEJ2000 "d:m:s"
1	17 42 05.9300	+05 24 13.900	C	174205.93+052413.9	15.113	14.535	13.849	13.200	12.831	-1.80	-19.56	-1.29	0.448	GCS	Simbad	17 42 05.93	+05 24 13.9
2	17 42 06.0200	+05 14 17.900	C	174206.02+051417.9	15.720	15.304	14.737	14.173	13.850	-56.48	24.63	-1.67	0.241	GCS	Simbad	17 42 06.02	+05 14 17.9
3	17 42 09.5800	+05 21 12.600	C	174209.58+052112.6	15.693	15.200	14.587	13.944	13.615	-36.93	-34.57	-1.61	0.267	GCS	Simbad	17 42 09.58	+05 21 12.6
4	17 42 16.9500	+05 26 51.300	C	174216.95+052651.3	15.670	15.269	14.676	14.082	13.761	-3.44	3.13	-1.65	0.251	GCS	Simbad	17 42 16.95	+05 26 51.3
5	17 42 17.7800	+05 56 26.200	C	174217.78+055626.2	16.616	16.024	15.344	14.767	14.377	-25.95	-12.29	-1.93	0.160	GCS	Simbad	17 42 17.78	+05 56 26.2
6	17 42 18.0000	+05 49 25.500	C	174218.00+054925.5	14.923	14.523	13.970	13.353	13.095	-8.36	4.92	-1.34	0.412	GCS	Simbad	17 42 18.00	+05 49 25.5
7	17 42 18.1900	+05 53 53.300	C	174218.19+055353.3	17.047	16.495	15.845	15.246	14.900	17.51	-8.99	-2.14	0.115	GCS	Simbad	17 42 18.19	+05 53 53.3
8	17 42 20.2900	+05 55 56.500	C	174220.29+055556.5	14.734	14.267	13.654	13.059	12.740	8.68	2.01	-1.21	0.508	GCS	Simbad	17 42 20.29	+05 55 56.5
9	17 42 20.7900	+05 46 35.600	C	174220.79+054635.6	15.508	15.061	14.449	13.906	13.586	-27.37	-2.79	-1.55	0.293	GCS	Simbad	17 42 20.79	+05 46 35.6
10	17 42 21.0800	+05 43 13.900	C	174221.08+054313.9	17.377	16.697	15.988	15.443	15.087	6.06	33.92	-2.20	0.104	GCS	Simbad	17 42 21.08	+05 43 13.9
11	17 42 23.5500	+05 38 23.500	C	174223.55+053823.5	15.344	14.881	14.291	13.705	13.403	1.01	-31.80	-1.48	0.328	GCS	Simbad	17 42 23.55	+05 38 23.5
12	17 42 24.8900	+05 06 06.100	C	174224.89+050606.1	16.389	15.827	15.163	14.587	14.272	0.98	-0.62	-1.85	0.181	GCS	Simbad	17 42 24.89	+05 06 06.1
13	17 42 25.4100	+06 21 05.300	C	174225.41+062105.3	14.731	14.381	13.831	13.177	12.885	-4.17	4.60	-1.29	0.453	GCS	Simbad	17 42 25.41	+06 21 05.3
14	17 42 25.6900	+05 29 47.200	C	174225.69+052947.2	16.944	16.476	15.847	15.208	14.889	10.33	10.36	-2.14	0.114	GCS	Simbad	17 42 25.69	+05 29 47.2
15	17 42 26.6000	+06 22 19.800	C	174226.60+062219.8	14.758	14.378	13.873	13.210	12.946	0.05	-6.42	-1.30	0.440	GCS	Simbad	17 42 26.60	+06 22 19.8
16	17 42 26.9300	+06 20 14.600	C	174226.93+062014.6	16.782	16.238	15.635	15.019	14.640	-4.73	11.58	-2.05	0.132	GCS	Simbad	17 42 26.93	+06 20 14.6
17	17 42 28.0300	+05 26 40.700	C	174228.03+052640.7	14.683	14.311	13.740	13.129	12.833	4.01	21.98	-1.25	0.481	GCS	Simbad	17 42 28.03	+05 26 40.7
18	17 42 28.9300	+05 54 53.800	C	174228.93+055453.8	17.691	17.048	16.427	15.888	15.495	28.31	79.04	-2.39	0.078	GCS	Simbad	17 42 28.93	+05 54 53.8
19	17 42 28.9400	+06 20 28.000	C	174228.94+062028.0	15.234	14.810	14.260	13.592	13.286	2.95	7.72	-1.47	0.336	GCS	Simbad	17 42 28.94	+06 20 28.0
20	17 42 31.9100	+06 18 49.500	C	174231.91+061849.5	14.560	14.139	13.606	12.988	12.679	-41.07	0.18	-1.19	0.524	GCS	Simbad	17 42 31.91	+06 18 49.5

How could a client process or even plot Position/Mags/PM/Mass entries of this catalog **without taking into consideration it comes from Vizier**?

2 Stakeholders: Data Provider/Consumer:

Component box

Position

Magnitude

Redshift

Proper Motion

{meta data}

Data

Data

DAL Belt

ICRS Position | Magnitude B | Proper Motion in mas/year

Data

Data

DAL Belt

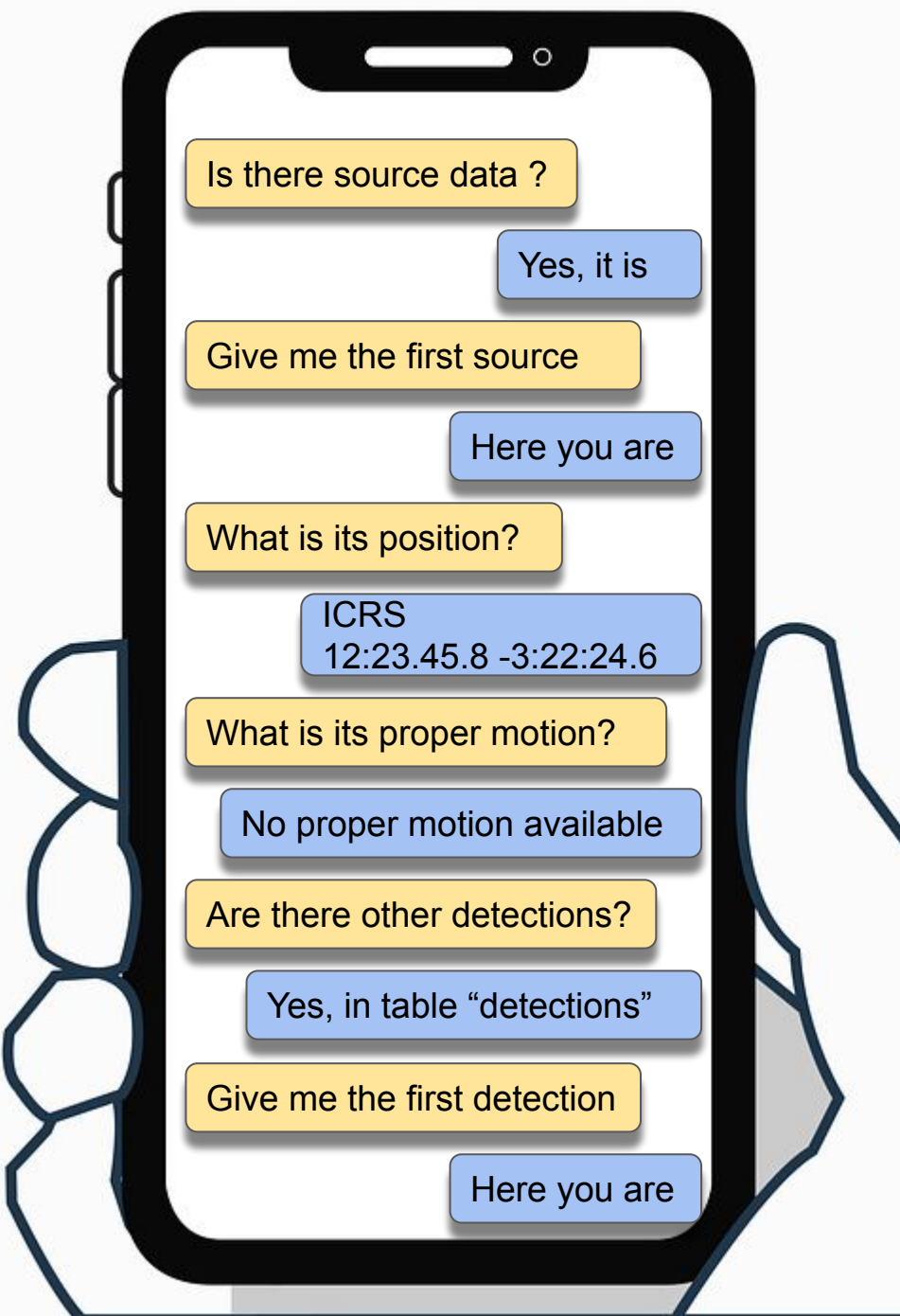The model is so discreet in this diagram that one may wonder if it exists

Client Perspective

```
SourceDM — -bash — 80×24
MacBook-Pro-de-Laurent-MICHEL:SourceDM laurentmichel$

parser = Parser("My VoTable")
sources = parser.getInstanceSet("CABMSD");

while( sources.hasNext())
    source = sources.next()
    print(source.get("position"))
    print(source.get("mag.G"))


MacBook-Pro-de-Laurent-MICHEL:SourceDM laurentmichel$
```

Is there source data ?

Yes, it is

Give me the first source

Here you are

What is its position?

ICRS
12:23.45.8 -3:22:24.6

What is its proper motion?

No proper motion available

Are there other detections?

Yes, in table "detections"

Give me the first detection

Here you are

We have to consider:

1. **The annotation content**
   a. Data modeling
      i. Including the import of VO standards (Meas/Coord NDCube, semantic…,)
   b. Serialization

2. **The data annotation process**
   a. Data provider point of view
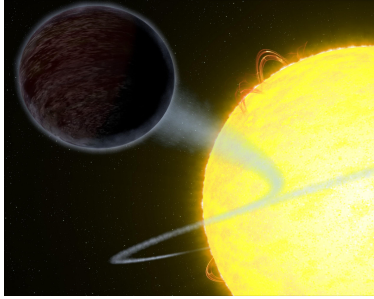   b. No hope to use the model as long as no data provider implements it

3. **The annotated data processing**
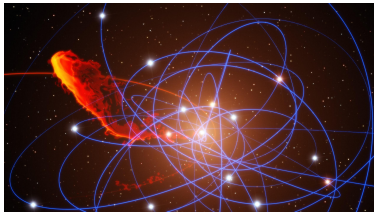   a. Client developer point of view

All of these are parts of the CAB-MSD project

Exoplanets

Orbiting stars

Complex shaped objects
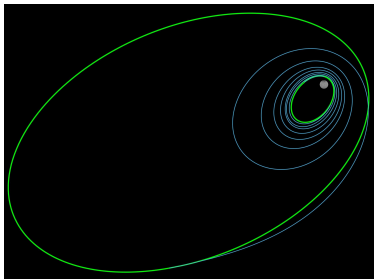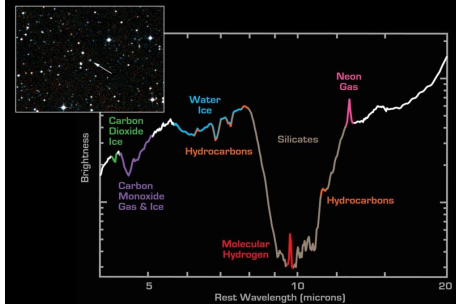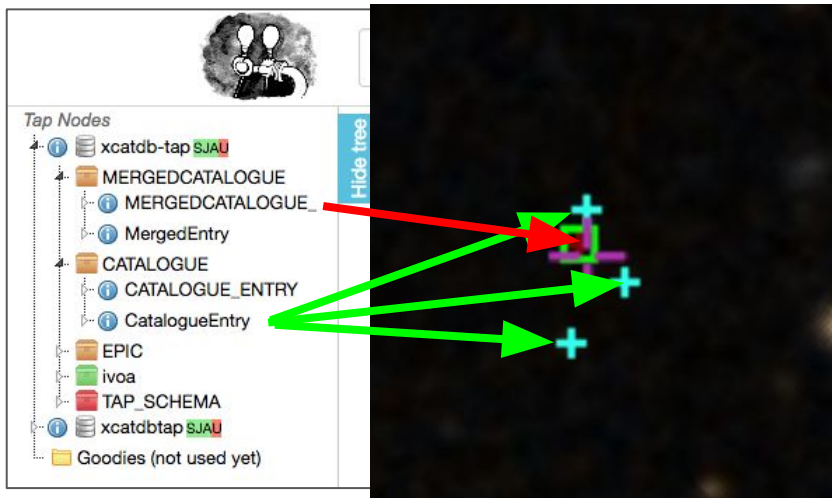
Complex errors

The Model: Associated Data
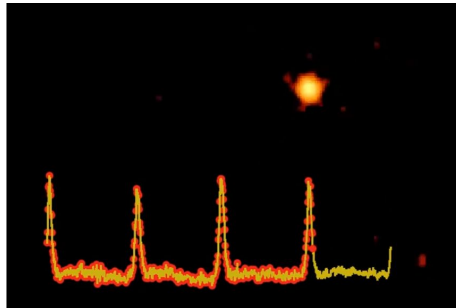
Time Series

Spectrum
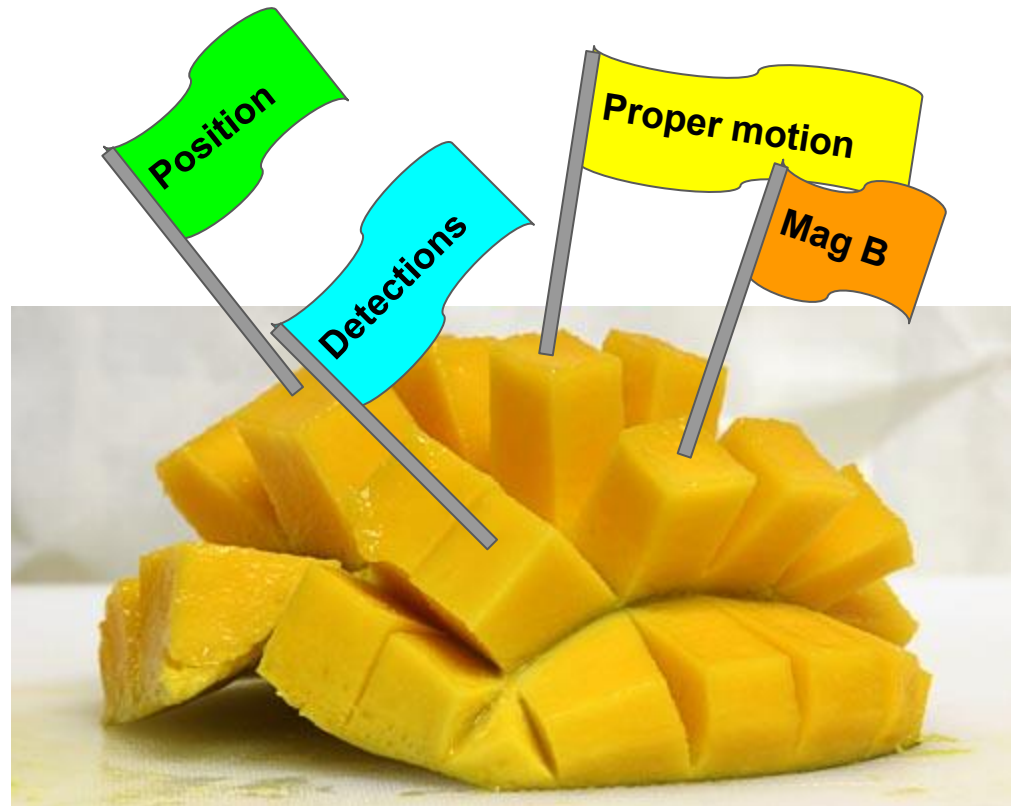
Multiple detections Correlations
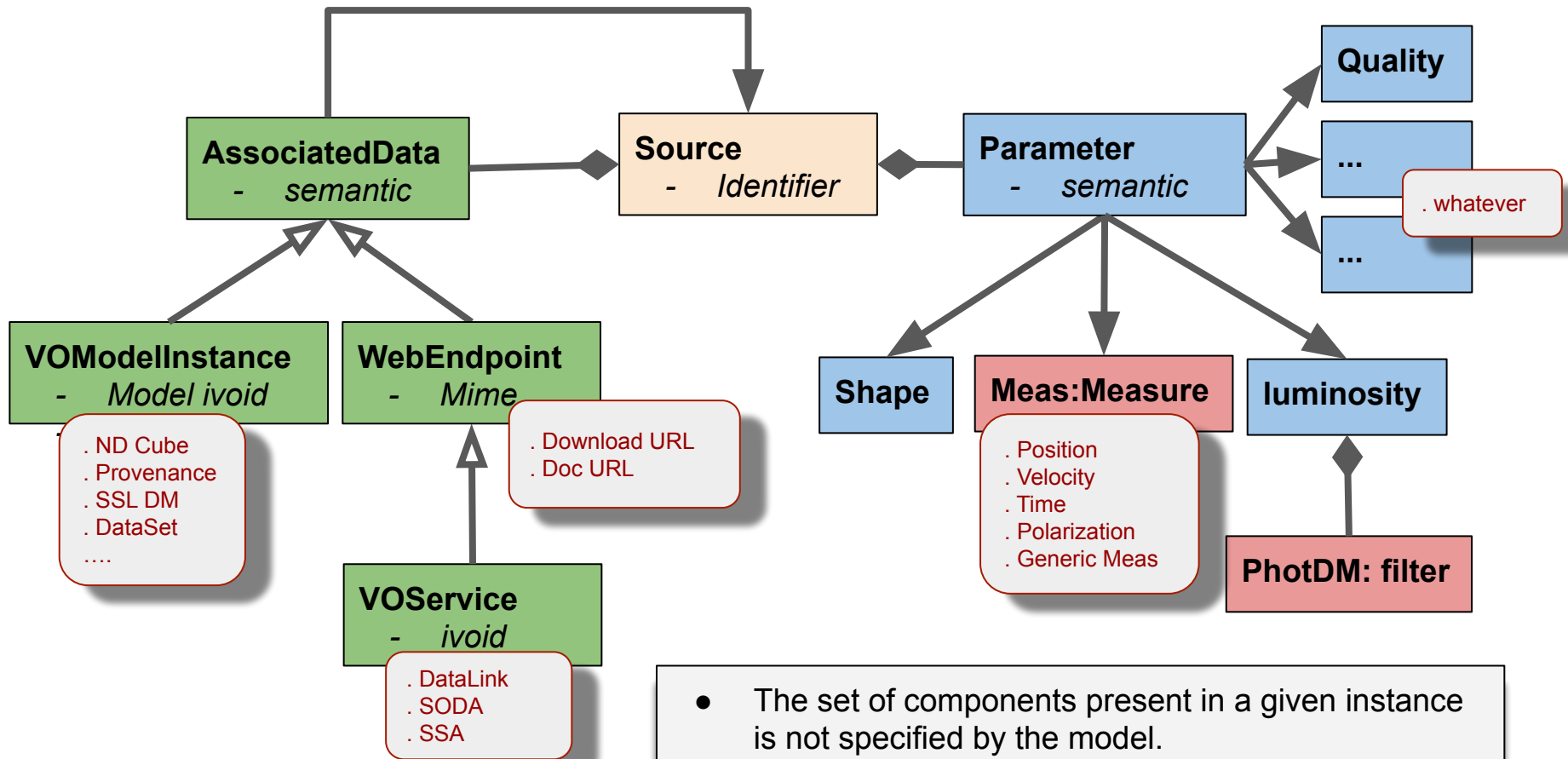
The Model: The Fruity Version

Source Data Model
- Everything is well packed in a model instance

Model for Source Data
- All available data can be discovered and retrieved in the model instance

Position
Detections
Proper motion
Mag B

The Model: The UML Version

AssociatedData
- *semantic*

Source
- *Identifier*

Parameter
- *semantic*

Quality

...

...

. whatever

VOModelInstance
- *Model ivoid*

. ND Cube
. Provenance
. SSL DM
. DataSet
….

WebEndpoint
- *Mime*

. Download URL
. Doc URL

VOService
- *ivoid*

. DataLink
. SODA
. SSA

Shape

Meas:Measure

. Position
. Velocity
. Time
. Polarization
. Generic Meas

**luminosity**

PhotDM: filter

- The set of components present in a given instance is not specified by the model.
  - Components can be duplicated
    - E.g. multiple positions
  - They have to be discovered by the client
- The structure of individual component is given by the model

Data Annotation: Bridge connecting model with Data

Bridge between data and model
Must be able to set missing data

Model

Meas:error

Meas:Point

Coord:frame

Coord:coord

Coord:coord
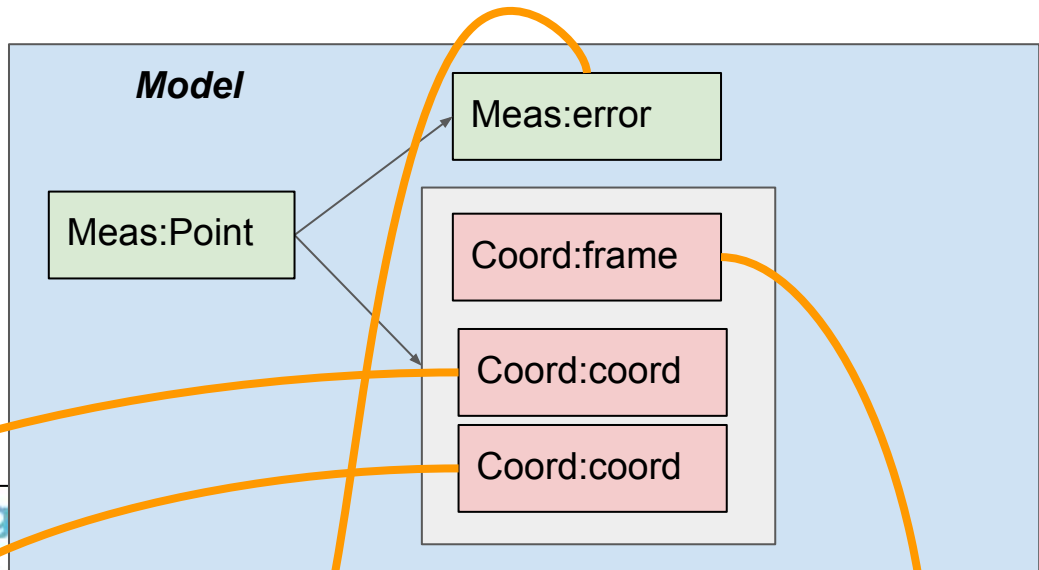
xcatdb-tap>MERGEDCATALOGUE>Merg

Show 5 entries   Showing 1 to 5 of 100 entries   🔍 Search

_iauname	_sc_ra	_sc_dec	_sc_poserr	_sc_ep_1_flux	_sc_ep_1_flux_err
3XMM J061628.5+223000	94.118827	22.500138	2	0	1
3XMM J061640.9+222438	94.170480	22.410767	5	6	6
3XMM J061704.2+090924	94.267916	9.1566678	1	0	1
3XMM J061706.7+224815	94.278027	22.804341	4	0	2
3XMM J061713.8+222914	94.307609	22.487336	1	0	6

?
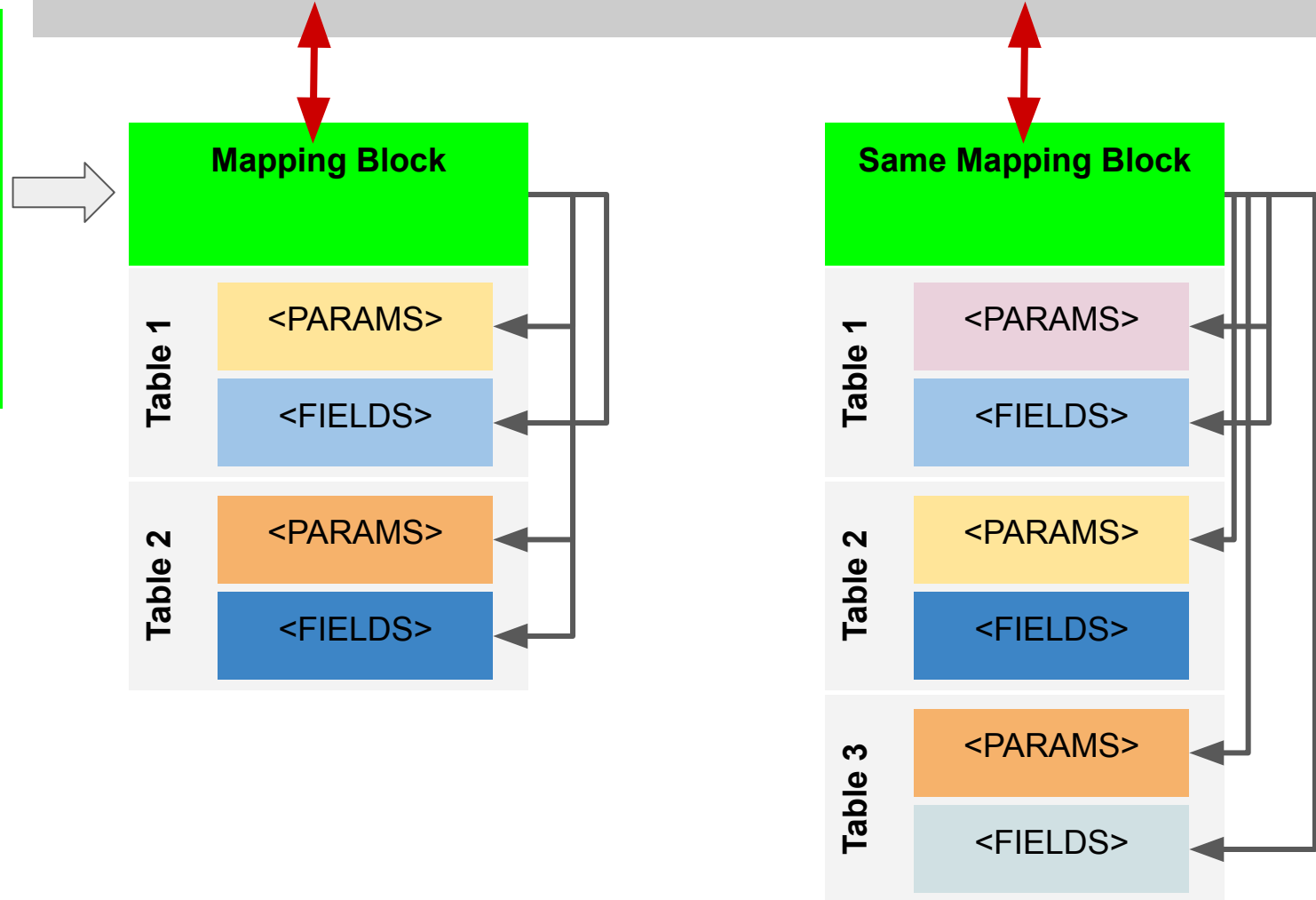
Mapping Process Overhead

- **Not Critical: Mission database**
  - A few number of different products
  - The source model mapping can be done once.

- **Critical: Archival Database (e.g. Vizier)**
  - Huge number of different products
    - Daily updated
  - Mapping a source model comes in addition to usual work
    - Must be done each time a new dataset is published
  - Must be a lightweight task
    - By minimising the amount of meta-data to be mapped
    - By using small **reusable components**

- **Very Critical: TAP services**
  - The possibility of automate the model mapping must be considered
    - This would be very useful for all VO stakeholders
  - There is no concrete proposal yet but (some ideas anyway)
    - Avoid to use show stopper features: **modularity**

Data annotation : "Don't be evil" (Larry Page)

- **Shy Annotations**
  - **#1:** Able to be ignored
    - Do not break working things
    - The parser implementation shouldn't alter the existing code
    - The annotation implementation shouldn't alter the original data

  - **#2:** To provide what is still missing in VOTables
    - A clear indication of the nature of the VOTable content

  - **#3:** Parser helper: Can be used at different levels
    - Just get the type of the VOTable content
    - Just the meta data
    - Just get column mapping
    - Get everything through the model

1. **VO-DML mapping**

   a. Mapping elements grouped outside of the data

   b. Model driven structure

   c. The mapping has its own schema

   d. Very chatty

My Proposal: Simplified syntax presented at College Park

ZTF example

Cone Search Query

Search region
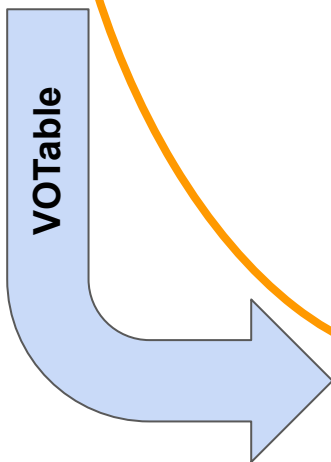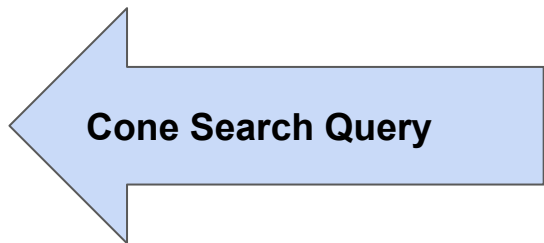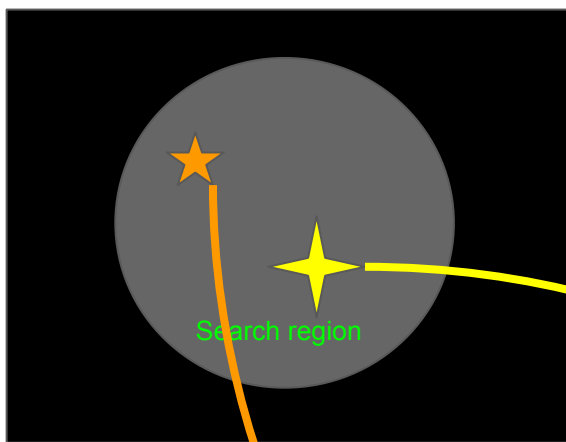
VOTable

Header

Oid ra dec time mag

Table

Challenge

Mapping this VOTable on CABMSD to retrieve the Time Series of the **Orange** star

ZTF Example: instanciated Components

https://irsa.ipac.caltech.edu/docs/program_interface/ztf_lightcurve_api.html
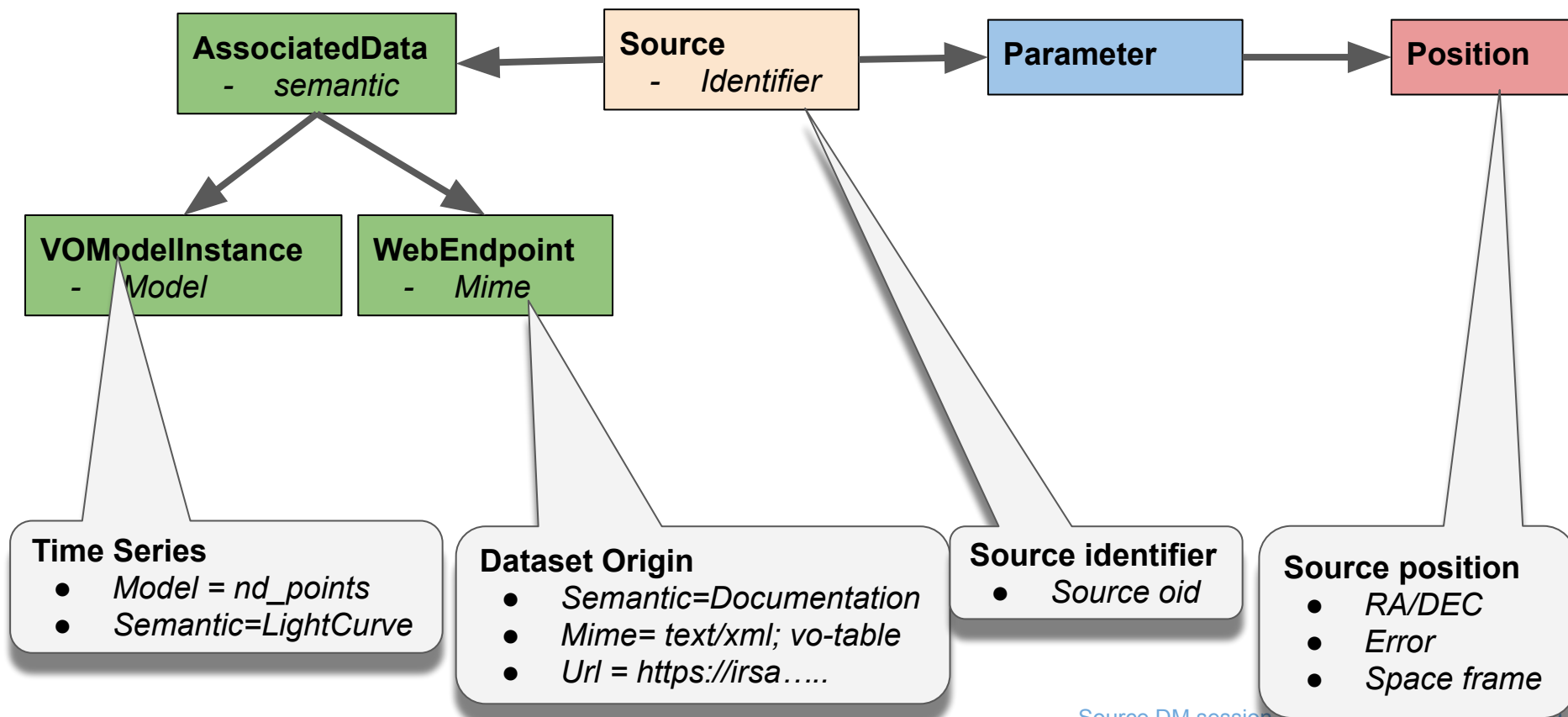
Behind the scenes

Typically there are multiple lightcurves for a given source position. Lightcurves are tagged in the ZTF lightcurve files by object identifier. As mentioned above, lightcurves corresponding to specific identifiers may be retrieved using the ZTF-LC-API via the ID parameter. Alternatively, this API also supports the retrieval of all lightcurves meeting positional and other constraints. When such constraints are supplied, a preliminary TAP query extracts the identifiers of all objects meeting those constraints from the ZTF objects table. Lightcurves corresponding to these identifiers are then retrieved from the ZTF lightcurve collection.

Sample queries

- Obtain the available g-band lightcurves within 5 arcsec of a source position:

  https://irsa.ipac.caltech.edu/cgi-bin/ZTF/nph_light_curves?POS=CIRCLE 298.0025 29.87147 0.0014&BANDNAME=g

---

AssociatedData
- *semantic*

Source
- *Identifier*

Parameter

Position

VOModelInstance
- *Model*

WebEndpoint
- *Mime*

Time Series
- *Model = nd_points*
- *Semantic=LightCurve*

Dataset Origin
- *Semantic=Documentation*
- *Mime= text/xml; vo-table*
- *Url = https://irsa…..*

Source identifier
- *Source oid*

Source position
- *RA/DEC*
- *Error*
- *Space frame*

```xml
<INSTANCE dmrole="nd_point:point" dmtype="nd_point:Point">
    <FILTER key="oid" value="686103400034440" />
    <!--
     Time Stamp : MJD supported CAB-MSD s
    -->
    <INSTANCE dmrole="nd_point:TimeStamp" dmtype="coords:MJD">
        <INSTANCE dmrole="nd_point:TimeStamp.date" dmtype="ivoa:RealQuantity">
            <VALUE dmrole="ivoa:Quantity.unit" dmtype="ivoa:Unit" value="d"/>
            <!-- fields have no ID: let's take their names as identifiers -->
            <VALUE dmrole="ivoa:RealQuantity.value" dmtype="ivoa:real"  ref="mjd" />
            <INSTANCE dmrole="coords:Coordinate.frame" ref="TimeFrame_TT_BARY"/>
        </INSTANCE>
    </INSTANCE>
    <!--
      Mag : The band is given by the filter
    -->
    <INSTANCE dmrole="nd_point:Mag" dmtype="nd_point:Mag">
        <INSTANCE dmrole="nd_point:Mag.coord" dmtype="nd_point:Coord">
            <VALUE dmrole="nd_point:mag.value" dmtype="ivoa:real"  ref="mag" />
            <INSTANCE dmrole="nd_point:Mag.frame" ref="Filter_G"/>
        </INSTANCE>
        <COMPOSITION dmrole="meas:Measure.error" size="1">
            <INSTANCE dmrole="meas:Measure.error"  ref="magerr"/>
        </COMPOSITION>
    </INSTANCE>
</INSTANCE>
```
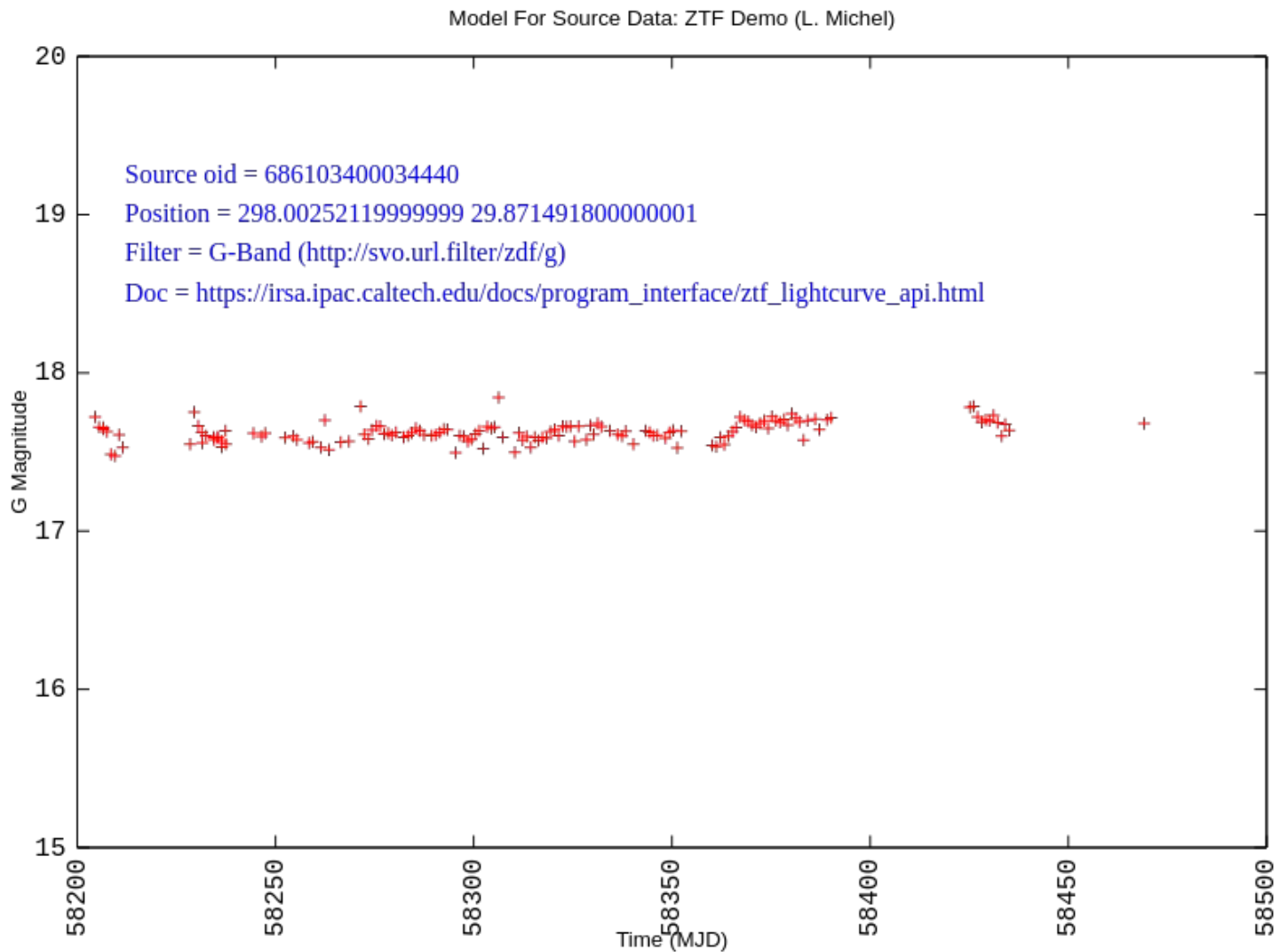
Model For Source Data: ZTF Demo (L. Michel)

Source oid = 686103400034440
Position = 298.00252119999999 29.871491800000001
Filter = G-Band (http://svo.url.filter/zdf/g)
Doc = https://irsa.ipac.caltech.edu/docs/program_interface/ztf_lightcurve_api.html

Moving Ahead: Roadmap

- **Assess the interest of the community for CAB-MSD**
- **Model**
  - Distribute use case requirements over imported model components and CABMSD classes
- **Data annotation**
  - Define the schema of the mapping syntax
  - Assess mapping methods with data providers (including TAP)
- **Client side**
  - Define a language-agnostic API (à la Xpath?)
  - Develop a AstroPy module
  - Develop a Java client
- **Start with a end to end implementation of selected cases**
- **Go in PR**

Pointers

- **Note:**
  - https://github.com/lmichel/Model-For-Source-Data/

- **ZTF Demo**
  - https://github.com/lmichel/vodml-lite-mapping
  - https://github.com/lmichel/vodml-lite-mapping/tree/master/src/test/java/test/xml

- **Wiki Page**
  - https://wiki.ivoa.net/twiki/bin/view/IVOA/SourceCatalogs

Do not read ahead….

Moving Ahead: One Big Open Question

- **Which protocol could carry CAB-MSD instances?**
  - The best (only) candidate is TAP
  - How annotate TAP responses now?

- **Is it possible to do more than a simple demonstrator?**
  - We could start with a few Vizier tables
  - And after?
  - Exercise with a specific TAP service?

PYTHON API Snippet

```
 * Extract the source measurements.
public void exploreDataSet() throws Exception{
    MappingElement dataSet = this.liteMappingParser.getFirstNodeWithRole("cab-msd:Source.identifier");
    this.cabmsdReport.sourceId = dataSet.getStringValue();
    dataSet = this.liteMappingParser.getFirstNodeWithRole("meas:EquatorialPosition.ra");
    this.cabmsdReport.ra = dataSet.getSubelementsByRole(IVOA_REAL_QUANTITY_VALUE).get(0).getStringValue();
    dataSet = this.liteMappingParser.getFirstNodeWithRole("meas:EquatorialPosition.dec");
    this.cabmsdReport.dec = dataSet.getSubelementsByRole(IVOA_REAL_QUANTITY_VALUE).get(0).getStringValue();

    this.cabmsdReport.docURL = this.liteMappingParser.getFirstNodeWithRole("cab-msd:WebUrl.url").getStringValue();
}
 * Extract the magnitude filter.
public void exploreFilters() throws Exception{
    MappingElement dataSet = this.liteMappingParser.getFirstNodeWithRole("cab-msd:FilterUrl");
    this.cabmsdReport.filterUrl = dataSet.getStringValue();
    dataSet = this.liteMappingParser.getFirstNodeWithRole("cab-msd:Filter.name");
    this.cabmsdReport.filterName = dataSet.getStringValue();
}
 * Extract the time series.
public void exploreData() throws Exception{
    MappingElement pointArray = this.liteMappingParser.getFirstNodeWithRole("nd_point:points");
    this.cabmsdReport.nbPoints = pointArray.getLength();
    this.cabmsdReport.columnMapping = pointArray.getColumnRoles();
    for( int i=0 ; i<this.cabmsdReport.nbPoints ; i++ ){
        MappingElement row = pointArray.getContentElement(i);
        this.cabmsdReport.points.add(
                new Point(
                        row.getSubelementsByRole(IVOA_REAL_QUANTITY_VALUE).get(0).getStringValue(),
                        row.getSubelementsByRole("nd_point:mag.value").get(0).getStringValue()
                        )
        );
    }
}
```
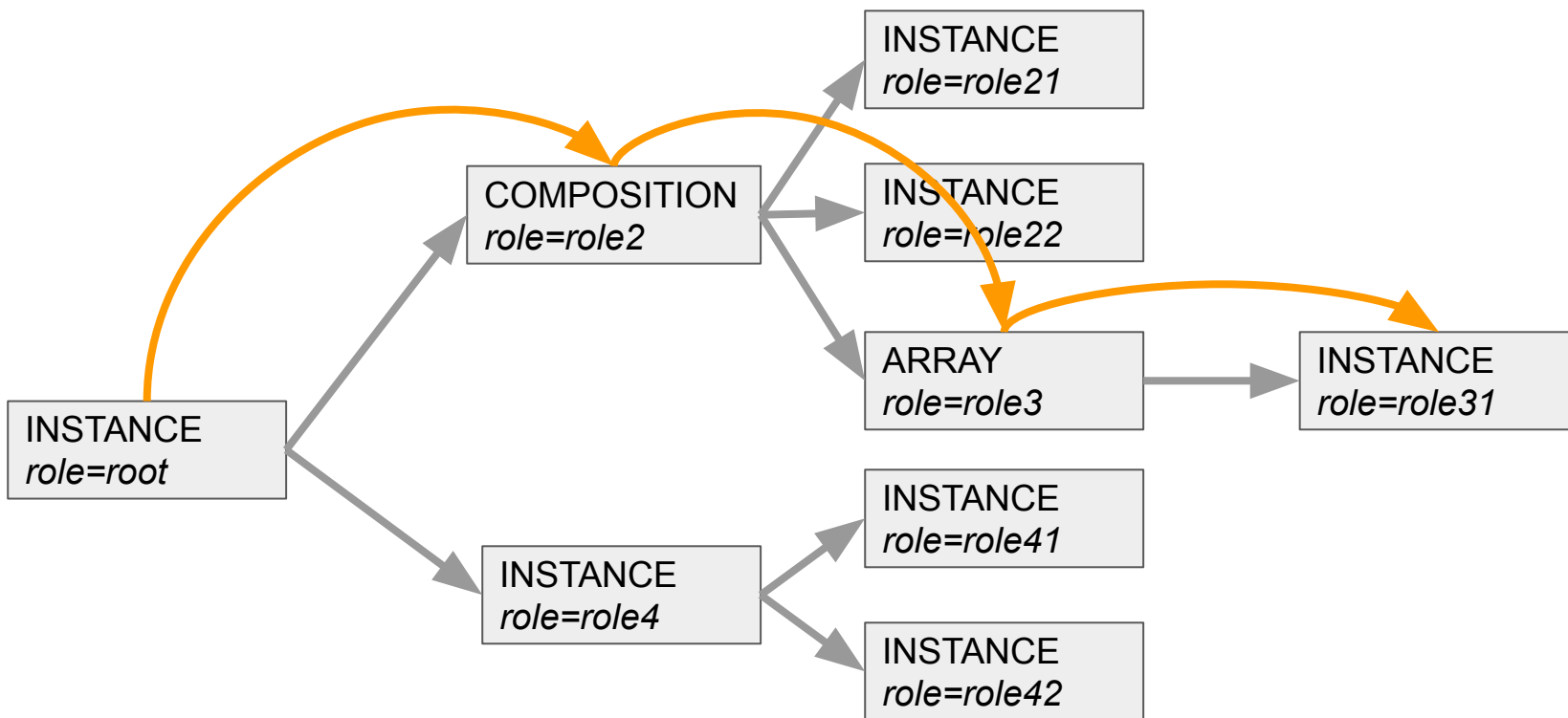
The Mapping DOM

INSTANCE
role=role21

COMPOSITION
role=role2

INSTANCE
role=role22

ARRAY
role=role3

INSTANCE
role=role31

INSTANCE
role=root

INSTANCE
role=role4

INSTANCE
role=role41

INSTANCE
role=role42

Parser dom
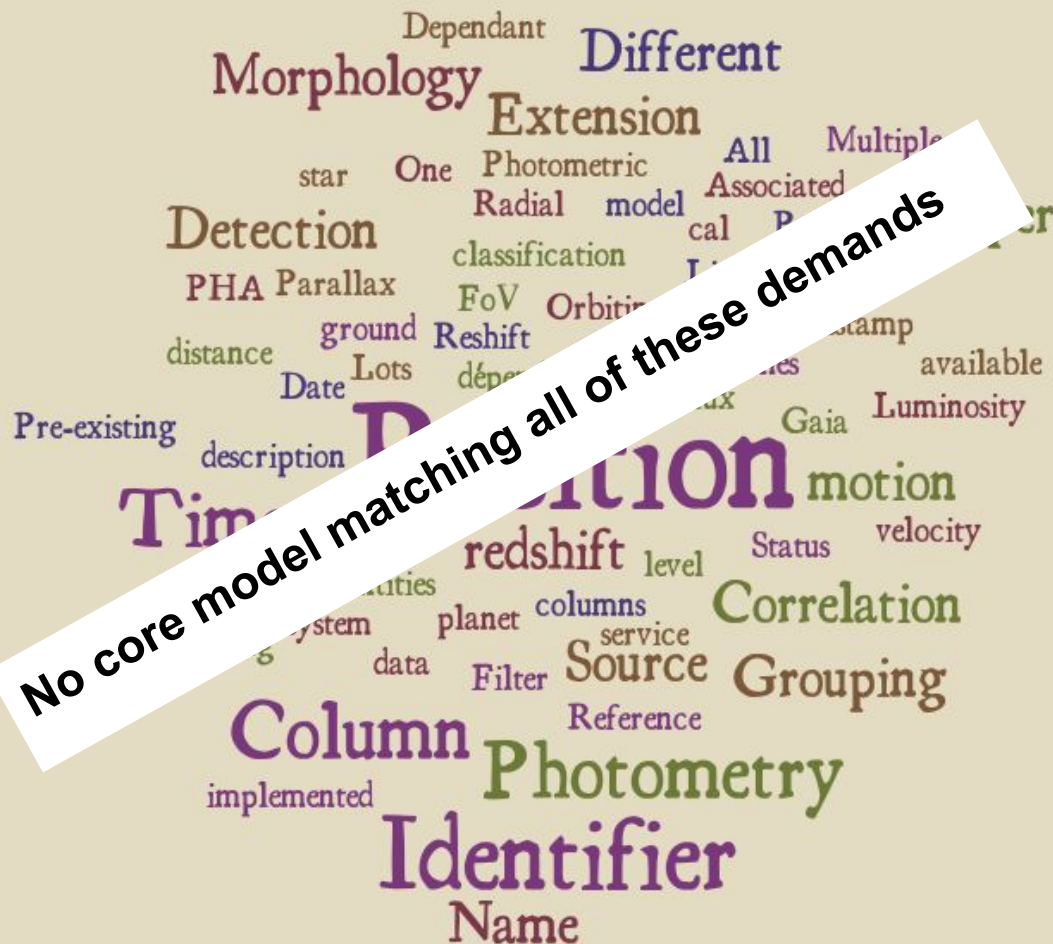
What the client does

```
Node(role=role31) = Node(role=root)
                ->Node(role=role2)
                    ->Node(role=role3)
                        ->Node(role=role31)
```

ZTF Example: Ass. data#2: T.S. origin

```xml
<INSTANCE dmrole="cab-msd:link" dmtype="cab-msd:VoInstance">
    <VALUE dmrole="cab-msd:link.semantic" value="LightCurve"/>
    <VALUE dmrole="cab-msd:VoInstance.model" value="nd_point"/>
    <INSTANCE dmrole="cab-msd:VoInstance.Instance"/>
    <!--
      ARRAY: One instance of  cab-msd:VoInstance per DATATABLE row
      -->
    <ARRAY dmrole="nd_point:TimeSeries.points">
        <!--
          set of measurements defined by the LightCurve model (time-stamp + mag)
          -->
        <INSTANCE dmrole="nd_point:TimeSeries.point" dmtype="nd_point:Point">
            <FILTER key="oid" value="686103400034440" />
            <!--
              Time Stamp : HJD not defined in STC, let's suppose that CAB-MSD supports it
              -->
            <INSTANCE dmrole="nd_point:TimeStamp" dmtype="cab-msd:HJD">
                <INSTANCE dmrole="nd_point:TimeStamp.date" dmtype="ivoa:RealQuantity">
                    <VALUE dmrole="ivoa:Quantity.unit" dmtype="ivoa:Unit" value="d"/>
                    <!-- fields have no ID: let's take their names as identifiers -->
                    <VALUE dmrole="ivoa:RealQuantity.value" dmtype="ivoa:real"  ref="hjd" />
                </INSTANCE>
                <INSTANCE dmrole="meas:Time.coord" dmtype="coords:TimeStamp">
                    <INSTANCE dmrole="coords:Coordinate.frame" ref="TimeFrame_TT"/>
                </INSTANCE>
                <COMPOSITION dmrole="meas:Measure.error" size="1">
                    <INSTANCE dmrole="meas:Measure.error" dmtype="meas:Error" >
                        <INSTANCE dmrole="meas:Error.ranError" dmtype="meas:Symmetrical">
                            <INSTANCE dmrole="meas:Symmetrical.radius" dmtype="ivoa:RealQuantity">
                                <VALUE dmrole="ivoa:Quantity.unit" dmtype="ivoa:Unit" value="sec"/>
                                <VALUE dmrole="ivoa:RealQuantity.value" dmtype="ivoa:real" value="1"/>
                            </INSTANCE>
                        </INSTANCE>
                    </INSTANCE>
                </COMPOSITION>
            </INSTANCE>
            <!--
              Mag : the band is given by the filter
              -->
            <INSTANCE dmrole="root" dmtype="nd_point:Mag">
                <INSTANCE dmrole="nd_point:Mag.coord" dmtype="nd_point:Mag">
                    <VALUE dmrole="nd_point:mag.value" dmtype="ivoa:real"  ref="mag" />
                    <INSTANCE dmrole="nd_point:Mag.frame" ref="Filter_G"/>
                </INSTANCE>
                <COMPOSITION dmrole="meas:Measure.error" size="1">
                    <INSTANCE dmrole="meas:Measure.error"  ref="magerr"/>
                </COMPOSITION>
            </INSTANCE>
        </INSTANCE>
    </ARRAY>
</INSTANCE>
```

```java
/**
 * Parse the file and build the mapping maps
 * @throws Exception
 */
private void initParser() throws Exception {
    URL url = ZdfExplorer.class.getResource(VOTABLE_RESOURCE);
    String sampleName =url.getFile();
    liteMappingParser = new LiteMappingParser(sampleName);
}


/**
 * Extract the source measurements
 * @throws Exception
 */
public void exploreDataSet() throws Exception{
    MappingElement dataSet = this.liteMappingParser.getFirstNodeWithRole("cab-msd:Source.identifier");
    this.zdfTimeSeriesReport.sourceId = dataSet.getStringValue();
    dataSet = this.liteMappingParser.getFirstNodeWithRole("meas:EquatorialPosition.ra");
    this.zdfTimeSeriesReport.ra = dataSet.getSubelementsByRole(REAL_VALUE).get(0).getStringValue();
    dataSet = this.liteMappingParser.getFirstNodeWithRole("meas:EquatorialPosition.dec");
    this.zdfTimeSeriesReport.dec = dataSet.getSubelementsByRole(REAL_VALUE).get(0).getStringValue();
}


/**
 * Extract the magnitude filter
 * @throws Exception
 */
public void exploreFilters() throws Exception{
    MappingElement dataSet = this.liteMappingParser.getFirstNodeWithRole("cab-msd:FilterUrl");
    this.zdfTimeSeriesReport.filterUrl = dataSet.getStringValue();
    dataSet = this.liteMappingParser.getFirstNodeWithRole("cab-msd:Filter.name");
    this.zdfTimeSeriesReport.filterName = dataSet.getStringValue();
}


/**
 * Extract the time series
 * @throws Exception
 */
public void exploreData() throws Exception{
    MappingElement pointArray = this.liteMappingParser.getFirstNodeWithRole("nd_point:TimeSeries.points");
    this.zdfTimeSeriesReport.nbPoints = pointArray.getLength();
    this.zdfTimeSeriesReport.columnMapping = pointArray.getColumnRoles();
    for( int i=0 ; i<this.zdfTimeSeriesReport.nbPoints ; i++ ){
        MappingElement row = pointArray.getContentElement(i);
        this.zdfTimeSeriesReport.points.add(
                new Point(
                        row.getSubelementsByRole(REAL_VALUE).get(0).getStringValue(),
                        row.getSubelementsByRole("nd_point:mag.value").get(0).getStringValue()
                        )
        );
    }
```

The Model: The guideline

- **What we do not need**
  - *"This is a VO compliant source instance, make your business with it."*

- **What do we need**
  - *"Here is a **set of data** attached to this **table row**, **help yourself**"*

> **The definition of what is a source is too much dependant from each use case to build a common model**
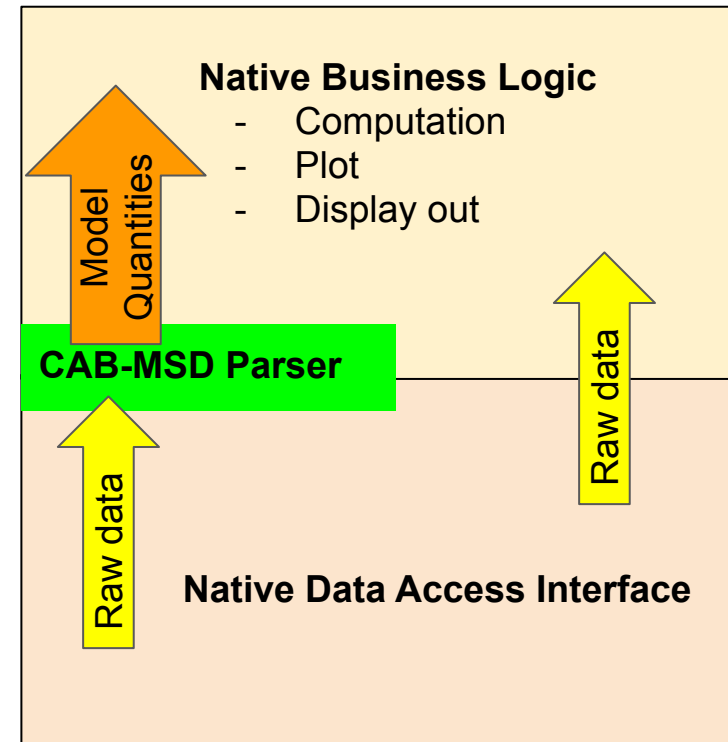
The parser code must be independent of any peculiar model

It must be able to retrieve individual quantities on demand

It must be designed in such a way it can be inserted in existing applications without breaking anything

Native Business Logic
- Computation
- Plot
- Display out

Model Quantities

CAB-MSD Parser

Raw data

Raw data

Native Data Access Interface

Mapping Snippets Process


```xml
<INSTANCE dmrole="@@@@@@" dmtype="meas:EquatorialPosition">
  <COLLECTION size="-1">
    <INSTANCE dmrole="meas:Measure.error" dmtype="meas:Error" ref="SymetricalError"/>
  </COLLECTION>
  <INSTANCE dmrole="meas:EquatorialPosition.dec" dmtype="coords:Latitude">
    <INSTANCE dmrole="coords:Coordinate.frame" ref="SpaceFrame_ICRS"/>
    <INSTANCE dmrole="coords:SpaceCoord.cval" dmtype="ivoa:RealQuantity">
      <VALUE dmrole="ivoa:Quantity.unit" dmtype="ivoa:Unit" ref="@@@@@@"/>
      <VALUE dmrole="ivoa:RealQuantity.value" dmtype="ivoa:real" ref="@@@@@@"/>
    </INSTANCE>
  </INSTANCE>
  <INSTANCE dmrole="meas:EquatorialPosition.ra" dmtype="coords:Longitude">
    <INSTANCE dmrole="coords:Coordinate.frame" ref="SpaceFrame_ICRS"/>
    <INSTANCE dmrole="coords:SpaceCoord.cval" dmtype="ivoa:RealQuantity">
```

mapping_componants
- EquatorialPosition.xml
- GenericCoordFrame.xml
- GenericMeasure.xml
- SpaceFrame_ICRS.xml
- SymetricalError.xml
- Time.xml
- TimeFrame_TT.xml
- zdr_annotated.xml

```xml
<INSTANCE dmrole="root" dmtype="coords:TimeFrame" ID="TimeFrame_TT">
  <INSTANCE dmrole="coords:TimeFrame.refPosition" dmtype="coords:StdRefLocation">
    <VALUE value="HELIOCENTRIC" dmrole="coords:StdRefLocation.position" dmtype="ivoa:string" ref="@@@@@@"/>
  </INSTANCE>
  <VALUE value="TT" dmrole="coords:TimeFrame.timescale" dmtype="ivoa:string" ref="@@@@@@"/>
</INSTANCE>
```

```xml
<INSTANCE dmrole="root" dmtype="meas:Time">
  <INSTANCE abstr="true" dmrole="meas:Time.coord" dmtype="coords:TimeStamp">
    <INSTANCE dmrole="coords:Coordinate.frame" ref="TimeFrame_TT"/>
  </INSTANCE>
  <COLLECTION size="-1">
    <INSTANCE dmrole="meas:Measure.error" ref="SymetricalError"/>
  </COLLECTION>
</INSTANCE>
```

```xml
<INSTANCE dmrole="root" dmtype="coords:SpaceFrame" ID="SpaceFrame_ICRS">
  <VALUE value="J2000" dmrole="coords:SpaceFrame.equinox" dmtype="coords:Epoch" ref="@@@@@@"/>
  <VALUE dmrole="coords:SpaceFrame.planetaryEphem" dmtype="ivoa:string" ref="@@@@@@"/>
  <VALUE value="ICRS" dmrole="coords:SpaceFrame.spaceRefFrame" dmtype="ivoa:string" ref="@@@@@@"/>
  <INSTANCE dmrole="coords:SpaceFrame.refPosition" dmtype="coords:StdRefLocation">
    <VALUE value="GEOCENTER" dmrole="coords:StdRefLocation.position" dmtype="ivoa:string" ref="@@@@@@"/>
  </INSTANCE>
</INSTANCE>
```

To be Checked and Completed

1. **Just saying what data are**

   a. This is a symmetrical error

2. **Saying how to use these data**

   a. This is a symmetrical error defined at 3sigma

3. **Giving information about the data provenance**

   a. This symmetrical error defined at 3sigma is function of (position, pm, RV mag)

Motivation for a Source Model

start AladinLite   plot the output   query using TAP/SQL

Full	RAJ2000 "h:m:s"	DEJ2000 "d:m:s"	Mm	Name	Zmag mag	Ymag mag	Jmag mag	Hmag mag	Kmag mag	pmRA mas/yr	pmDE mas/yr	logL [Lsun]	Mass Msun	GCS	Simbad	RAJ2000 "h:m:s"	DEJ2000 "d:m:s"
1	17 42 05.9300	+05 24 13.900	C	174205.93+052413.9	15.113	14.535	13.849	13.200	12.831	-1.80	-19.56	-1.29	0.448	GCS	Simbad	17 42 05.93	+05 24 13.9
2	17 42 06.0200	+05 14 17.900	C	174206.02+051417.9	15.720	15.304	14.737	14.173	13.850	-56.48	24.63	-1.67	0.241	GCS	Simbad	17 42 06.02	+05 14 17.9
3	17 42 09.5800	+05 21 12.600	C	174209.58+052112.6	15.693	15.200	14.587	13.944	13.615	-36.93	-34.57	-1.61	0.267	GCS	Simbad	17 42 09.58	+05 21 12.6
4	17 42 16.9500	+05 26 51.300	C	174216.95+052651.3	15.670	15.269	14.676	14.082	13.761	-3.44	3.13	-1.65	0.251	GCS	Simbad	17 42 16.95	+05 26 51.3
5	17 42 17.7800	+05 56 26.200	C	174217.78+055626.2	16.616	16.024	15.344	14.767	14.377	-25.95	-12.29	-1.93	0.160	GCS	Simbad	17 42 17.78	+05 56 26.2
													0.412	GCS	Simbad	17 42 18.00	+05 49 25.5
													0.115	GCS	Simbad	17 42 18.19	+05 53 53.3
													0.508	GCS	Simbad	17 42 20.29	+05 55 56.5
													0.293	GCS	Simbad	17 42 20.79	+05 46 35.6
													0.104	GCS	Simbad	17 42 21.08	+05 43 13.9
													0.328	GCS	Simbad	17 42 23.55	+05 38 23.5
													0.181	GCS	Simbad	17 42 24.89	+05 06 06.1
													0.453	GCS	Simbad	17 42 25.41	+06 21 05.3
													0.114	GCS	Simbad	17 42 25.69	+05 29 47.2
													0.440	GCS	Simbad	17 42 26.60	+06 22 19.8
													0.132	GCS	Simbad	17 42 26.93	+06 20 14.6
													0.481	GCS	Simbad	17 42 28.03	+05 26 40.7
													0.078	GCS	Simbad	17 42 28.93	+05 54 53.8
													0.336	GCS	Simbad	17 42 28.94	+06 20 28.0
													0.524	GCS	Simbad	17 42 31.91	+06 18 49.5

```
SOURCE  : 174325.35+052128.5
POSITION: 17 43 25.35 +05 21 28.5
P.MOTION: -50.22 -46.39
MAGS    : I=18.264 z=17.664 Z=17.763 Y=17.058 J=16.386 H=15.803 K=15.392
SOURCE  : 174330.63+050217.5
POSITION: 17 43 30.63 +05 02 17.5
P.MOTION: -19.53 -10.09
MAGS    : I=18.312 z=17.641 Z=17.760 Y=17.144 J=16.457 H=15.895 K=15.563
SOURCE  : 174331.38+052723.3
POSITION: 17 43 31.38 +05 27 23.3
P.MOTION: -1.24 7.69
MAGS    : I=17.350 z=16.854 Z=16.912 Y=16.395 J=15.779 H=15.218 K=14.865
SOURCE  : 174331.39+053403.6
POSITION: 17 43 31.39 +05 34 03.6
P.MOTION: -904.33 -1413.85
MAGS    : I=18.705 z=18.013 Z=18.107 Y=17.431 J=16.745 H=16.184 K=15.753
SOURCE  : 174331.77+063945.3
POSITION: 17 43 31.77 +06 39 45.3
P.MOTION: -2.70 -9.07
MAGS    : I=16.149 z=15.739 Z=15.778 Y=15.317 J=14.662 H=13.971 K=13.697
```