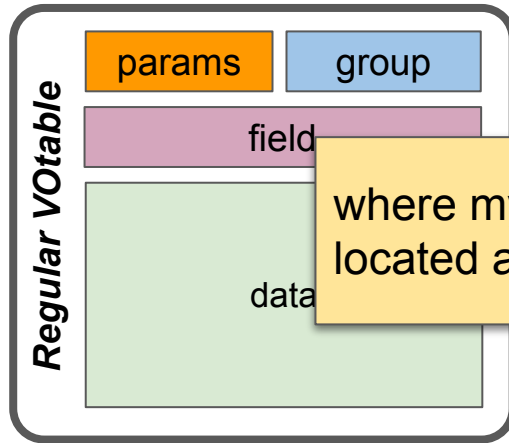


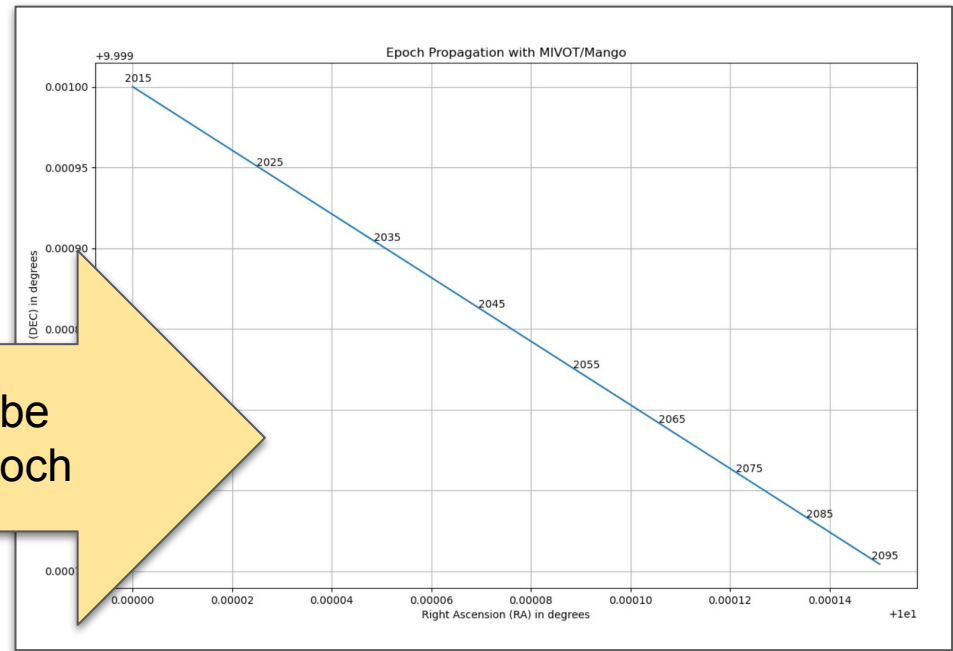
# SOLVING THE EPOCH PROPAGATION CASE WITH MIVOT AND THE MODELS

Laurent MICHEL on the behalf of all DMers

# The Challenge



where my object will be located at a given epoch



**In this VOTABLE, I have somewhere:**

- Position
- Proper motion
- Parallax
- Radial velocity
- Desired metadata

**The challenge: find an appropriate way to do it**

# The Challenge

## 2 approaches

- Use legacy VOTable features
  - PARAM, FIELD, GROUP, COOSYS
- Use all available VOTable features
  - PARAM, FIELD
- **Model Instance in VOTable** (MIVOT + models)

## 2 aspects to tackle with anyway

- SERVER: Arrange VOTables elements to facilitate the processing automation
- CLIENT: Implement the processing automation

# Legacy: Position + Proper Motion

```
<COOSYS ID="J2000" equinox="J2000" epoch="J2000" system="eq_J2000"/>
<FIELD name="pos_RA" ucd="pos.eq.ra;meta.main" datatype="double" unit="deg" ref="J2000"/>
<FIELD name="pos_DEC" ucd="pos.eq.dec;meta.main" datatype="double" unit="deg" ref="J2000"/>
<FIELD name="pm_RA" ucd="pos.pm.ra;meta.main" datatype="double" unit="mas/y" ref="J2000"/>
<FIELD name="pm_DEC" ucd="pos.pm.dec;meta.main" datatype="double" unit="mas/year" ref="J2000"/>
```



- Can see with UCDs that **pos\_RA** and **pos\_DEC** do work together
- Can see with UCDs that **pm\_RA** and **pm\_DEC** do work together
- The 4 columns refer to the **COOSYS** element



- The role of the **@ref->@ID** link is implicit (**@ref** to what?)
- No clear way to see that **pos\_RA/pos\_DEC** and **pm\_RA/pm\_DEC** relate to the same quantity
- No clear way to list all **FIELDS** that are connected to **COOSYS**

# legacy : Solution 1: Add FIELDREF into COOSYS On

behalf of Markus D.

- The Epoch propagation components are **grouped into the COOSYS** element

- They map an **ad-hoc model** based on utypes reusing names of VO model elements

- **Small changes** in the VOTable **schema**
- **Easy to implement** in actual parsers

- Introduce **bi-directional links** which is a **new VOTable feature**
- EPOCH as a literal value
- Complex errors **not supported**
- **Short term** solution



the future in favor of a more generic way of describing the conventions used to define the positions of the objects studied in the enclosed tables.

```
<VOTABLE
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="1.4" xsi:schemaLocation="http://
    /xml/VOTable/v1.3 http://vo.ari.uni-heidelberg.de/docs/schemata/VOTable-1.4.xsd">
  <RESOURCE type="results">
    <TABLE>
      <COOSYS ID="system" epoch="J2015.0" refposition="BARYCENTER" system="ICRS">
        <FIELDref ref="dist" utype="votable:LonLatPoint-dist"/>
        <FIELDref ref="delta" utype="votable:LonLatPoint-lat"/>
        <FIELDref ref="alpha" utype="votable:LonLatPoint-lon"/>
        <FIELDref ref="pmdec" utype="votable:ProperMotion-lat"/>
        <FIELDref ref="pmra" utype="votable:ProperMotion-lon"/>
        <FIELDref ref="rv" utype="votable:ProperMotion-rv"/>
      </COOSYS>
      <TIMESYS ID="ts" refposition="GEOCENTER" timescale="TT"/>
      <FIELD ID="alpha" datatype="float" name="alpha" ref="system"/>
      <FIELD ID="delta" datatype="float" name="delta" ref="system"/>
      <FIELD ID="pmra" datatype="float" name="pmra" ref="system"/>
      <FIELD ID="pmdec" datatype="float" name="pmdec" ref="system"/>
      <FIELD ID="dist" datatype="float" name="dist" ref="system"/>
      <FIELD ID="rv" datatype="float" name="rv" ref="system"/>
      <FIELD ID="tca" datatype="float" name="tca" ref="ts"/>
    <DATA>
      <BINARY>
        <STREAM encoding="base64">QSAAAMEgAAA6gxJvuNG3F0I0AADAgAAAR1X/4A==</STREAM>
      </BINARY>
    </DATA>
  </TABLE>
</RESOURCE>
</VOTABLE>
```

Note that the **COOSYS** may be deprecated in

# Legacy: Solution 2: add FIELDREF into COOSYS

On behalf of François B.

- The Epoch propagation components are packed **into a GROUP** element
- The **GROUP** role is set with a UType reusing names of VO model elements
- The **GROUP** refers to the **COOSYS**

- **No change** in the VOTable **schema**
- **No bi-directional** links
- Can have **multiple GROUPS** for multiple parameter sets
- **Easy to implement** in actual parsers

- **Alternative** model mapping syntax **less featured** than MIVOT
- Do not rely on any documented data model
- Complex errors **not supported**
  - Covariance, correlation
- **Short term** solution

```
<COOSYS ID="J2000" equinox="J2000" epoch="J2000" system="eq_J2000"/>
<GROUP utype="demo:epoch.propagation" ref="J2000">
  <FIELDREF ref="pos_RA" ucd="pos.eq.ra;meta.main" />
  <FIELDREF ref="pos_DEC" ucd="pos.eq.dec;meta.main" />
  <FIELDREF ref="pm_RA" ucd="pos.pm.ra;meta.main" />
  <FIELDREF ref="pm_DEC" ucd="pos.pm.dec;meta.main" />
  <FIELDREF ref="RV" ucd="spect.dopplerVeloc;pos.heliocentric" />
  <FIELDREF ref="PARALLAX" ucd="pos.parallax" />
</GROUP>
<FIELD name="pos_RA" ucd="pos.eq.ra;meta.main" datatype="double" unit="deg" />
<FIELD name="pos_DEC" ucd="pos.eq.dec;meta.main" datatype="double" unit="deg" />
<FIELD name="pm_RA" ucd="pos.pm.ra;meta.main" datatype="double" unit="mas/y" />
<FIELD name="pm_DEC" ucd="pos.pm.dec;meta.main" datatype="double" unit="mas/year" />
<FIELD name="RV" ucd="spect.dopplerVeloc;pos.heliocentric" datatype="double" unit="km/s" />
<FIELD name="PARALLAX" ucd="pos.parallax" datatype="double" unit="mas" ref="J2000" />
```

the future in favor of a more generic way of describing the conventions used to define the positions of the objects studied in the enclosed tables.

Note that the **COOSYS** may be deprecated in

# Legacy: Limitation of the COOSYS solutions



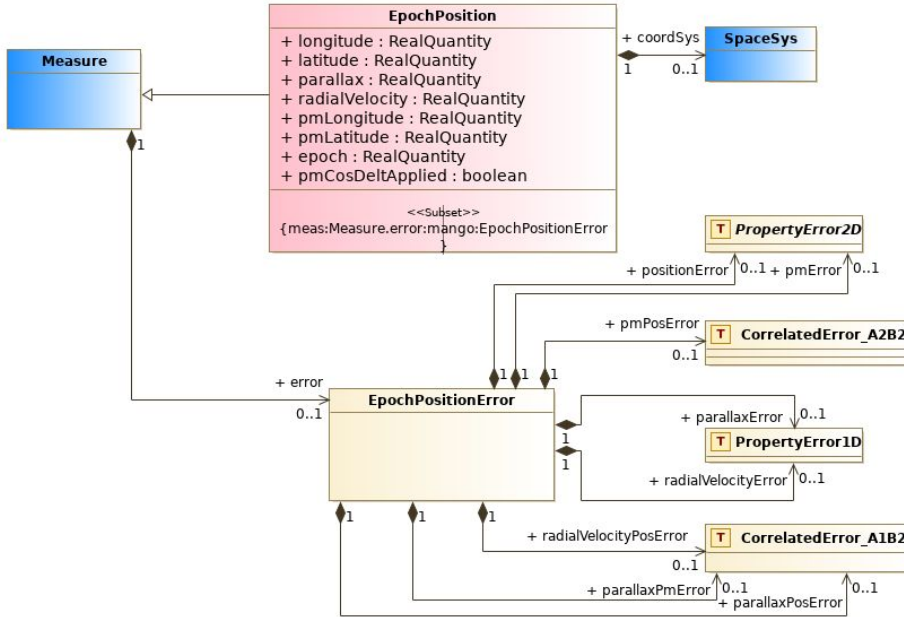
- The `@ref` semantic or **role is not well defined**
- The way fields connected to `coosys` **do interact** together is **poorly defined**
- **No** underlying **data model** (common formal description of the quantity)
- According the the VOTable document, solution based on `coosys` are **short term solution**

# MIVOT: The Model Based Alternative

- Keep the VOTable as delivered by the ADQL engine
  - No FIELDREF tuning
  - No GROUP to create
- Take a VO model describing the EPOCH propagation
  - 6 parameters
  - Errors
  - Space Frame
- Write an XML serialization of that MODEL in the VOTable
  - MIVOT syntax
  - Put the proper FIELD references into model leaves
- Insert that XML piece above the TABLE



# MIVOT: The Daunting Step: Build a Model



- The role of all model components is perfectly defined by the model

- The class is part of MANGO draft
  - The MANGO overview is not shown here
- Use of Meas/Coord
  - Extends *Measure* classes from the Measure Model
  - Use *Coords* classes to describe coordinate systems
- Support complex errors
  - Per parameter errors
  - Covariance errors
  - Correlated errors
- No need to use all the features proposed by the model
  - only use model elements that match data

# MIVOT: Add a Mapping Block above the Data Table

- The space coordinate system is a **GLOBAL** object that can be referenced by any other MIVOT element

- Each table row can be interpreted as an instance of the class **EpochPosition** of the MANGO model

```
<MODEL name="coords" url="https://www.ivoa.net/xml/STC/20200908/Coords-v1.0.vo-dml.xml" />
<MODEL name="mango" />
<MODEL name="ivoa" url="https://www.ivoa.net/xml/VODML/IVOA-v1.vo-dml.xml" />
<GLOBALS>
  <INSTANCE dmid="SpaceFrame_ICRS" dmtpe="coords:SpaceSys">
    <INSTANCE dmrole="coords:PhysicalCoordSys.frame" dmtpe="coords:SpaceFrame">
      <INSTANCE dmrole="coords:SpaceFrame.refPosition" dmtpe="coords:StdRefLocation">
        <ATTRIBUTE dmrole="coords:StdRefLocation.position" dmtpe="ivoa:string" value="NoSet" />
      </INSTANCE>
      <ATTRIBUTE dmrole="coords:SpaceFrame.spaceRefFrame" dmtpe="ivoa:string" value="ICRS" />
    </INSTANCE>
  </INSTANCE>
</GLOBALS>
<TEMPLATES>
  <INSTANCE dmtpe="mango:EpochPosition">
    <REFERENCE dmrole="coords:Coordinate.coosys" dmref="SpaceFrame_ICRS"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.longitude" dmtpe="ivoa:RealQuantity" ref="pos_RA"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.latitude" dmtpe="ivoa:RealQuantity" ref="pos_DEC" />
    <ATTRIBUTE dmrole="mango:EpochPosition.pmLongitude" dmtpe="ivoa:RealQuantity" ref="pm_RA" />
    <ATTRIBUTE dmrole="mango:EpochPosition.pmLatitude" dmtpe="ivoa:RealQuantity" ref="pm_DEC"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.pmCosDeltApplied" dmtpe="ivoa:boolean" value="true"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.radialVelocity" dmtpe="ivoa:RealQuantity" ref="RV"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.parallax" dmtpe="ivoa:RealQuantity" ref="PARALLAX" />
    <ATTRIBUTE dmrole="mango:EpochPosition.epoch" dmtpe="coords:Epoch" value="J2016.0" unit="year"/>
  </INSTANCE>
</TEMPLATES>
</VODML>
```

# MIVOT: Add a Mapping Block above the Data Table

- The space coordinate system is a **GLOBAL** object that can be referenced by any other MIVOT element

- Each table row can be interpreted as an instance of the class **EpochPosition** of the MANGO model

```
<MODEL name="coords" url="https://www.ivoa.net/xml/STC/20200908/Coords-v1.0.vo-dml.xml" />
<MODEL name="mango" />
<MODEL name="ivoa" url="https://www.ivoa.net/xml/VODML/IVOA-v1.vo-dml.xml" />
<GLOBALS>
  <INSTANCE dmid="SpaceFrame_ICRS" dmtpe="coords:SpaceSys">
    <INSTANCE dmrole="coords:PhysicalCoordSys.frame" dmtpe="coords:SpaceFrame">
      <INSTANCE dmrole="coords:SpaceFrame.refPosition" dmtpe="coords:StdRefLocation">
        <ATTRIBUTE dmrole="coords:StdRefLocation.position" dmtpe="ivoa:string" value="NoSet" />
      </INSTANCE>
      <ATTRIBUTE dmrole="coords:SpaceFrame.spaceRefFrame" dmtpe="ivoa:string" value="ICRS" />
    </INSTANCE>
  </INSTANCE>
</GLOBALS>
<TEMPLATES>
  <INSTANCE dmtpe="mango:EpochPosition">
    <REFERENCE dmrole="coords:Coordinate.coosys" dmref="SpaceFrame_ICRS"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.longitude" dmtpe="ivoa:RealQuantity" ref="pos_RA"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.latitude" dmtpe="ivoa:RealQuantity" ref="pos_DEC" />
    <ATTRIBUTE dmrole="mango:EpochPosition.pmLongitude" dmtpe="ivoa:RealQuantity" ref="pm_RA" />
    <ATTRIBUTE dmrole="mango:EpochPosition.pmLatitude" dmtpe="ivoa:RealQuantity" ref="pm_DEC"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.pmCosDeltApplied" dmtpe="ivoa:boolean" value="true"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.radialVelocity" dmtpe="ivoa:RealQuantity" ref="RV"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.parallax" dmtpe="ivoa:RealQuantity" ref="PARALLAX" />
    <ATTRIBUTE dmrole="mango:EpochPosition.epoch" dmtpe="coords:Epoch" value="J2016.0" unit="year"/>
  </INSTANCE>
</TEMPLATES>
</VODML>
```

# MIVOT: Add a Mapping Block above the Data Table

- The space coordinate system is a **GLOBAL** object that can be referenced by any other MIVOT element

- Each table row can be interpreted as an instance of the class **EpochPosition** of the MANGO model

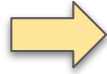
- Class attributes refer to the columns that are used to set their values

- Some class attributes can have fixed values, they don't hold column references

```
<MODEL name="coords" url="https://www.ivoa.net/xml/STC/20200908/Coords-v1.0.vo-dml.xml" />
<MODEL name="mango" />
<MODEL name="ivoa" url="https://www.ivoa.net/xml/VODML/IVOA-v1.vo-dml.xml" />
<GLOBALS>
  <INSTANCE dmid="SpaceFrame_ICRS" dmtpe="coords:SpaceSys">
    <INSTANCE dmrole="coords:PhysicalCoordSys.frame" dmtpe="coords:SpaceFrame">
      <INSTANCE dmrole="coords:SpaceFrame.refPosition" dmtpe="coords:StdRefLocation">
        <ATTRIBUTE dmrole="coords:StdRefLocation.position" dmtpe="ivoa:string" value="NoSet" />
      </INSTANCE>
      <ATTRIBUTE dmrole="coords:SpaceFrame.spaceRefFrame" dmtpe="ivoa:string" value="ICRS" />
    </INSTANCE>
  </INSTANCE>
</GLOBALS>
<TEMPLATES>
  <INSTANCE dmtpe="mango:EpochPosition">
    <REFERENCE dmrole="coords:Coordinate.coosys" dmref="SpaceFrame_ICRS"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.longitude" dmtpe="ivoa:RealQuantity" ref="pos_RA"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.latitude" dmtpe="ivoa:RealQuantity" ref="pos_DEC" />
    <ATTRIBUTE dmrole="mango:EpochPosition.pmLongitude" dmtpe="ivoa:RealQuantity" ref="pm_RA" />
    <ATTRIBUTE dmrole="mango:EpochPosition.pmLatitude" dmtpe="ivoa:RealQuantity" ref="pm_DEC"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.pmCosDeltaApplied" dmtpe="ivoa:boolean" value="true"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.radialVelocity" dmtpe="ivoa:RealQuantity" ref="RV"/>
    <ATTRIBUTE dmrole="mango:EpochPosition.parallax" dmtpe="ivoa:RealQuantity" ref="PARALLAX" />
    <ATTRIBUTE dmrole="mango:EpochPosition.epoch" dmtpe="coords:Epoch" value="J2016.0" unit="year"/>
  </INSTANCE>
</TEMPLATES>
</VODML>
```

# MIVOT: Flexibility

EPOCH defined in a **<FIELD>**  
@ref to the FIELD identifier



```
<ATTRIBUTE dmrole="tucson:Position.epoch"  
           dmtype="coords:Epoch"  
           unit="year"  
           ref="_EPOCH_FIELD"/>
```

EPOCH defined in a **<PARAM>**  
@ref to the PARAM identifier



```
<ATTRIBUTE dmrole="tucson:Position.epoch"  
           dmtype="coords:Epoch"  
           unit="year"  
           ref="_EPOCH_PARAM"/>
```

EPOCH defined in a non  
machine-readable element  
No @ref but a fixed @value

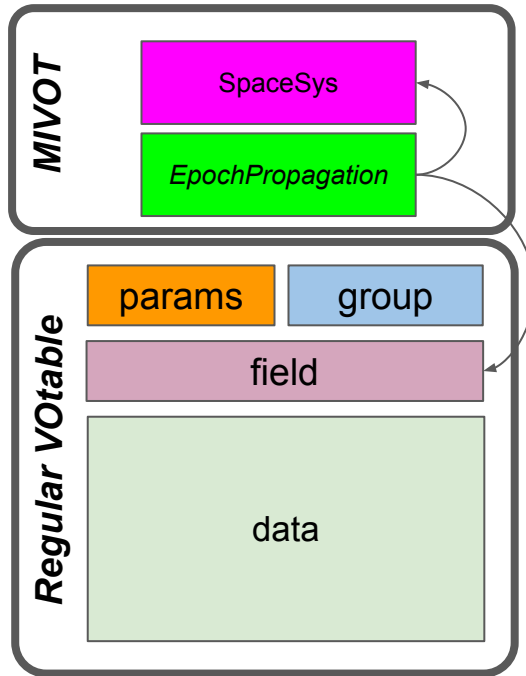


```
<ATTRIBUTE dmrole="tucson:Position.epoch"  
           dmtype="coords:Epoch"  
           unit="year"  
           value="J2023.88"/>
```

The structure of MIVOT block is not altered by the way the EPOCH is set in the VOTable

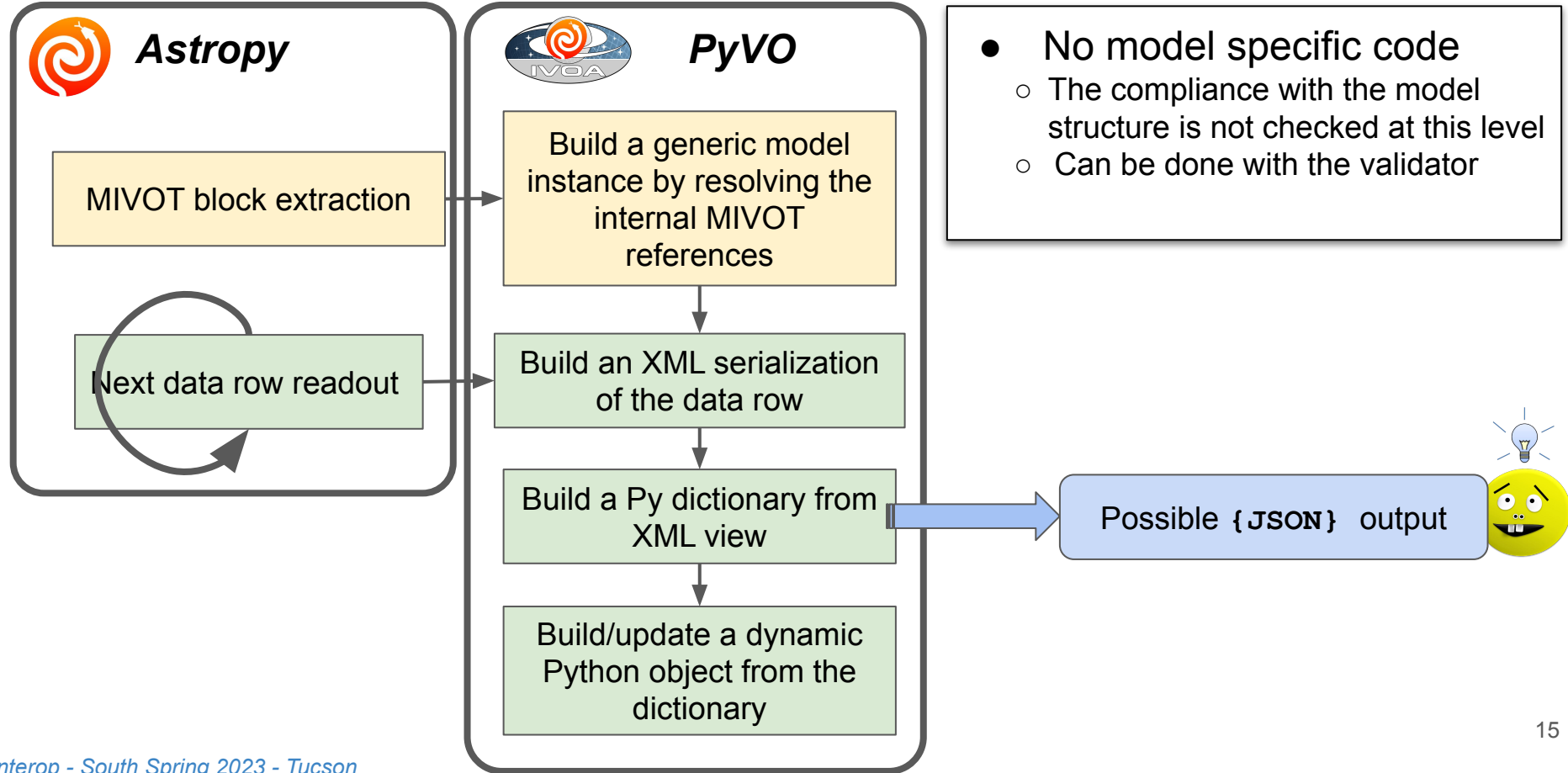
- Allow server code to be versatile
- Allow a same client code to process many different VOTable

# MIVOT: MIVOT annotations



- **MIVOT block: an XML model view**
  - Above the data table
  - The hierarchy of the XML elements denotes the model structure
  - References to the appropriate columns
  - Syntax controlled by the MIVOT XML schema
- **The client can easily get model instances**
  - Read the MIVOT block
  - Resolve the reference to the FIELDS
  - Set the attribute values with the row data

# MIVOT: PyVO implementation,



# MIVOT: PyVO implementation: user point of view

- Go through the model view with Python object fields
- Field names match the model roles
  - Escape rules
- Based on public and documented models

```
from pyvo.mivot.interpreter.model_view import ModelViewer

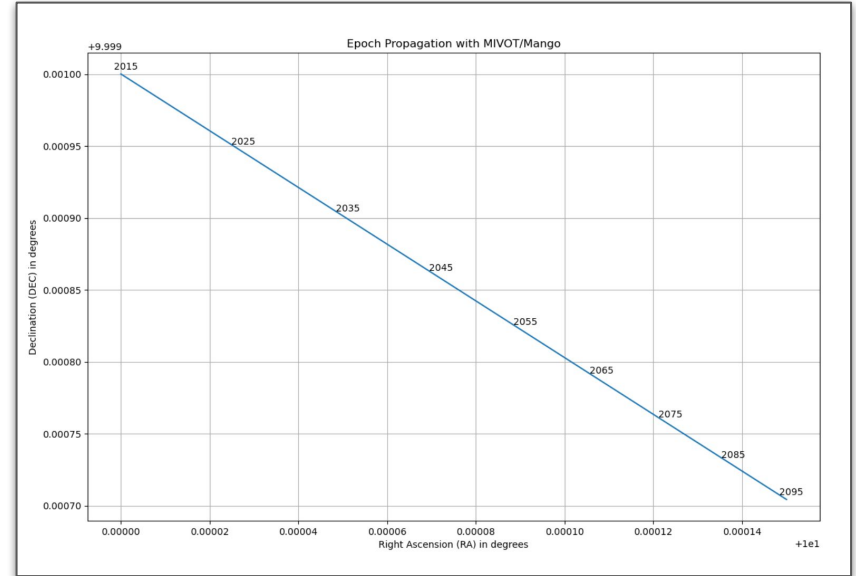
with ModelViewer("whatever-votable.xml") as m_viewer
    while (row_view := m_viewer.get_next_row_view):
        ra = row_view.EpochPostion.longitude.value
        dec = row_view.EpochPostion.latitude.value
        pm_ra = row_view.EpochPostion.longitude.value
        pm_dec = row_view.EpochPostion.latitude.value
        radial_velocity = row_view.EpochPostion.radialVelocity.value
        parallax = row_view.EpochPostion.parallax.value
        # Do whatever you want with those values
```



# MIVOT: PyVO: Epoch Propagation Implementation

```
import matplotlib.pyplot as plt
from pyvo.mivot.viewer.model_viewer import ModelViewer

years = np.arange(2015, 2030, 1)
with ModelViewer("path_to_my_votable") as m_viewer:
    # get the model view on the current data row
    row_view = m_viewer.get_next_row_view()
    # store the every year positions
    positions = [c.apply_space_motion(dt=year * u.year) for year in years]
    ra = [pos.icrs.ra.deg for pos in positions]
    dec = [pos.icrs.dec.deg for pos in positions]
    # Plot the object position over the years
    plt.figure(figsize=(14, 9))
    plt.plot(ra, dec)
    # do some plot polishing and exit
    break
```



# Aladin Desktop Prototype

Aladin v12.1 \*\*\* BETA VERSION (based on v12.107) \*\*\*

Available data → 34241  
● in view ● out view

Command [ ] Frame ICRS Projection Aitoff

DSS PanSTARRS SDSS 2MASS GALEX Gaia Simbad NED +

DSS2 color

select pan dist phot draw tag

Welcome to Aladin,  
your professional sky atlas.

- Discover all astronomical data available over the net!
- Compare them with your own data.
- Prepare your observation missions.

To start, type any object name, press ENTER...

filter cross x-y rgb

Catania

11851 size dens. opac. zoom

00:05:08.84 +07:50:24.6'

180 +18

select

Many stars do move

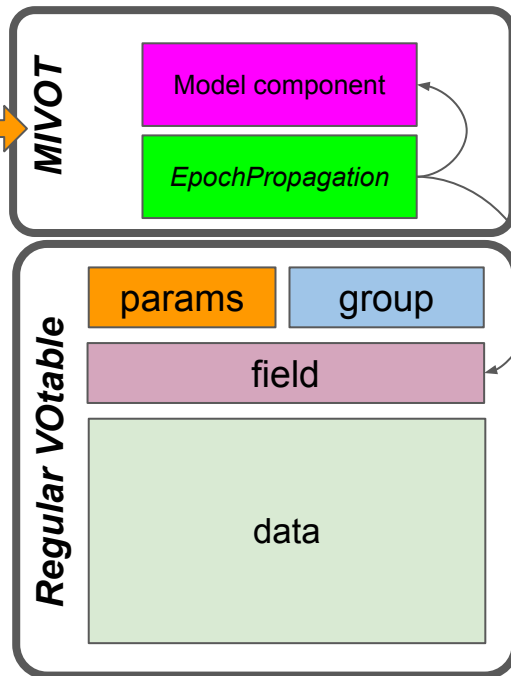
If you move that cursor

Believe me!

# Server Side Implementation

```
<VODML
xmlns:dm:napping="http://www.lvoa.net/xml/mivot">
  <!-- MIVOT block template -->
  <MODEL name="mango" url="https://www.lvoa.net/xml/v1.0.vo-dnl.xml" />
  <MODEL name="lvoa" url="https://www.lvoa.net/xml/VODML/lvoa-v1.0.vo-dnl.xml" />
  <GLOBALS>
    <INSTANCE dnId="SpaceFrame_ICRS" dnType="coords:SpaceSys">
      <INSTANCE dnRole="coords:PhysicalCoordsSys.frame" dnType="coords:SpaceFrame">
        <ATTRIBUTE dnRole="coords:SpaceFrame.spaceRefFrame" dnType="lvoa:string" value="ICRS" />
      </INSTANCE>
    </INSTANCE>
  </GLOBALS>
  <TEMPLATES>
    <!-- One meas:Measure per row, no error -->
    <TEMPLATE dnType="mango:EpochPosition">
      <ATTRIBUTE dnRole="mango:EpochPosition.longitude" dnType="lvoa:RealQuantity" ref="pos_RA"/>
      <ATTRIBUTE dnRole="mango:EpochPosition.latitude" dnType="lvoa:RealQuantity" ref="pos_DEC"/>
      <ATTRIBUTE dnRole="mango:EpochPosition.pmLongitude" dnType="lvoa:RealQuantity" ref="pm_RA"/>
      <ATTRIBUTE dnRole="mango:EpochPosition.pmLatitude" dnType="lvoa:RealQuantity" ref="pm_DEC"/>
      <ATTRIBUTE dnRole="mango:EpochPosition.pmCosDelApplied" dnType="lvoa:boolean" value="true"/>
      <ATTRIBUTE dnRole="mango:EpochPosition.radialVelocity" dnType="lvoa:RealQuantity" ref="RV"/>
      <ATTRIBUTE dnRole="mango:EpochPosition.parallax" dnType="lvoa:RealQuantity" ref="PARALLAX"/>
      <ATTRIBUTE dnRole="mango:EpochPosition.epoch" dnType="lvoa:RealQuantity" value="2015.0"/>
      <REFERENCE dnRole="coords:Coordinate.coosys" dnRef="SpaceFrame_ICRS"/>
    </TEMPLATE>
  </TEMPLATES>
</VODML>
```

## MIVOT block template



## Mapping rules

- Give the binding between table columns and model leaves
- Can be stored in the TAP\_SCHEMA

- **The implementation is quite simple**
  - Do not alter the VOTABLE content
  - Just add a MIVOT block at the right place
- **Can easily extend existing services**
  - Can be done as a TAP query response post processing

# Status

- **Models**

- Meas/Coords/PhotDM: REC
- MANGO: Currently a pre-draft (need to be validated against VOT 1.5 requirements)

- **Client side**

- AstroPy: Merged (#15390 see V6.0.0.dr1)
- PyVO: Draft PR (#497)
- RUST: Under development at the CDS with the goal of improving the cross-match

- **Server Side**

- A TAP service hosting various datasets must be deployed by the beginning of 2024
- We are working on a VOLLT (CDS TAP lib) extension for annotating query responses

- **IVOA**

- Discussion should start to define a way to register the MIVOT capability
  - Concerned: TAP/UWS/Registry/DALI/...
- Looking for others implementers

# Conclusions

## **MIVOT+Mango: a seamless solution for the Epoch propagation**

- Model supporting the complex errors
- Astropy/PyVO API
- Server side implementation preserving the original VOTable
- No change in the VOTable schema

## **The same mechanism can be used for many others quantities**

- Versatile mapping syntax
- Photometric data
- Dataset meta data
- ...

"eq\_FK4"), and `epoch` specifies the epoch of the positions if necessary. Note that the `COOSYS` may be deprecated in the future in favor of a more generic way of describing the conventions used to define the positions of the objects studied in the enclosed tables.

# BACK-UP

# Legacy: Connect Sky Position with a Space Frame

```
<COOSYS ID="J2000" epoch="J2000" system="eq_J2000"/>  
<FIELD name="pos_RA" ucd="pos.eq.ra;meta.main" datatype="double" unit="deg" ref="J2000"/>  
<FIELD name="pos_DEC" ucd="pos.eq.dec;meta.main" datatype="double" unit="deg" ref="J2000"/>
```



- Can see with the UCDs that **RA** and **Dec** do work together
- Both columns refer to the COOSYS element

- The **ref** attribute is used to quote another element of the document in the definition of a **FIELD** or **PARAM**. It is used in the example of section 3.1 to indicate the coordinate system in which the coordinates are expressed (reference to the **COOSYS** element which specifies the coordinate frame).



- The role of the **@ref->@ID** link is implicit
- **@ref** to what?

# In the Hood

```
<instance dmtype="tucson:Position">
  <attribute dmrole="tucson:Position.ra"
    dmtype="ivoa:RealQuantity" value="10.876"
    ref="_RA2000"/>
  <attribute dmrole="tucson:Position.dec"
    dmtype="ivoa:RealQuantity" value="-45.765"
    ref="_DEC2000"/>
</instance>
```

XML view transformed as a Py dict  
MIVOT element names are removed  
Only keep labels and values of interest

```
# class naming not defined yet
class MivotAttribute:
    def __init__(self, value, ref, unit)
        self.value = value
        self.ref = ref
        self.unit = unit

class Position:
    def __init__(self, ra, dec):
        self.ra = ra
        self.dec = dec
```

```
{
  "@dmtype": "tucson:Position",
  "tucson:Position.ra" : {
    "@dmtype": "ivoa:RealQuantity",
    "value": 10.876,
    "unit": "deg",
    "ref": "_RA2000"
  },
  "tucson:Position.dec" : {
    "@dmtype": "ivoa:RealQuantity",
    "value": -45.765,
    "unit": "deg",
    "ref": "_DEC2000"
  }
}
```

Py dict transformed as a dynamic Py object  
Can be incorporated to the application logic



# Work with the Column References

```
#get the model view
m_view m_viewer.get_next_row_view()

# Get the position from the model view
ra = m_view.EpochPropagation.longitude.value
dec = m_view.EpochPropagation.latitude.value

# get the column attached to a model leaf
column_hosting_ra = m_view.EpochPropagation.longitude.ref
column_hosting_dec = m_view.EpochPropagation.latitude.ref
```

The Python API give access to the reference of the columns that have been used to set attributes

```
# update the model view without redoing the parsing
table = votable.to_table()
for row in table:
    m_viewer.EpochPropagation.longitude.value
    = row[m_view.EpochPropagation.longitude.ref]
    m_viewer.EpochPropagation.latitude.value
    = row[m_view.EpochPropagation.latitude.ref]
```

This can be used to update the Python object by skipping the parsing step

```
# get the next RA directly from the data row
table = votable.to_table()
for row in table:
    ra_value = row[m_view.EpochPropagation.longitude.ref]
    dec_value = row[m_view.EpochPropagation.latitude.ref]
```

This can be used to get attribute values without using the model view