

Attendees: ranged from 46 to 56

Tom Donaldson announces splinter session on Friday after GS. 14:40UTC

## **Markus Demleitner - Getting COOSYS ready for 2025**

Alberto Micol - Utypes are invited for this proposal?

Alberto Micol: Illustrate how the proposal solves the problem:

Problem is the epoch slider.

Current COOSYS will mix different references of position and proper motion.

Lauren Michel: Doesn't understand the reason

MD- there should be a way in VOTable to address this in 90% of cases

Data provider to supply both annotations

Tom Donaldson: fundamental question - should the VOTable contain description of the notions or just the tabular data.

In the interest of time we should move to the next presentation but continue with this with a meeting in Gather Town

Zoom Chat:

Alberto Micol to Everyone (6:42 AM)

Is it not possible to add a timesys attribute to the COOSYS?

gilles landais to Everyone (6:43 AM)

reference is a good idea, there are tables (in Vizier) where the epochs are defined in a column

Alberto Micol to Everyone (6:44 AM)

I see, thanks Gilles

Dave Morris to Everyone (6:47 AM)

Not sharing the right window ?

We see the LaTeX presentation not Aladin

gilles landais to Everyone (6:49 AM)

what about protocols - like conesearch - could it be possible to query a table using an epoch ?

Pierre Fernique (CDS) to Everyone (6:50 AM)

Just keep in mind that is already possible to repeat COOSYS definition for handling multiple coordinates column with various attributes

François Bonnarel to Everyone (6:54 AM)

Why not using GROUPS with FIELDref to FIELDS and PARAM with ref to COOSYS / TIMESYS instead of Changing COOSYS ?

Markus Demleitner to Everyone (6:55 AM)

gilles: this is independent of any concrete protocol -- and I personally wouldn't add EPOCH to SCS (but see my DAL talk on what I'd do about epoch propagation in TAP).

Pierre: Well, yes, but as long as the referencing is in the wrong direction, it's all gambling (and the role assignment would also need a solution).

Markus Demleitner to Everyone (6:56 AM)

François: COOSYS is already there -- and I'd say it's really the obvious choice of a thing like that.

s/of/for/

Pierre Fernique (CDS) to Everyone (6:57 AM)

but the role assignment (utype) is already present in the FIELD.

Markus Demleitner to Everyone (6:58 AM)

yeah, but that breaks when a single FIELD is referenced in two roles (e.g. as a value in a TIMESYS and an epoch in a COOSYS)

gilles landais to Everyone (7:01 AM)

good point!

Pierre Fernique (CDS) to Everyone (7:03 AM)

That's right, but it's more like it's well within the 10% of complicated cases, and perhaps a more general solution based on the efforts of the DM Working Group will be more appropriate. To be discussed

Markus Demleitner to Everyone (7:06 AM)

(of course, it's also much easier to implement if you reference from COOSYS to FIELD-s, independent of 90/10 considerations). But yes, let's discuss this -- specifically, I won't start walking without your blessing, Pierre.

Pierre Fernique (CDS) to Everyone (7:09 AM)

I am neither for nor against, and the proposed solution seems to me to be both light and compatible with the existing system. But it will require a major effort in the choice of utypes and I fear that a third way will be attempted when the second solution (DM) is no longer very far from being accepted.

Markus Demleitner to Everyone (7:12 AM)

Well, my hope is that there's a synergy when people figuring out their STC metadata for Coord/Meas can add this for free and make it simple and robust to do epoch propagation.

## **Francois-Xavier Pineau - MOC lib Rust and its derivatives: MOCPy, MOCWasm, MOCCLI and MOCSet**

Tom Donaldson - I like the way RUST has entry points for javascript, python

No questions

Zoom Chat:

Jon Juaristi to Everyone (7:13 AM)

As for MOC support on C/PostgreSQL, recent pgsphere versions ought to have it: <https://github.com/credativ/pgsphere> ("moc" branch)

gilles landais to Everyone (7:14 AM)

just a note on Francois-xavier implementation.

This architecture manages well memory - memorymap enables more freedom to share memory and is well adapted for server being limited in memory!

Markus Demleitner to Everyone (7:16 AM)

Update: MOC support in pg\_sphere is actually merged upstream

(<https://github.com/postgrespro/pgsphere>), and it's in the Debian package.

## Omar Laurino - Supporting Prototype Implementations in PyVO

MD - I'd like to have an opt out option as well

Developers went back and forth about opt in or opt out model.

With deprecation there's a warning and could present an alternative implementation

Discussion on the chat. PR could be used to express further comments.

Omar Laurino - we could go in phases and be able to turn on as soon there's a draft for example.

Based on the chat, I'm trying to incline towards to the optin model multitier approach might be a good way

Zoom Chat:

Xiuqin Wu (she/her) to Everyone (7:22 AM)

I prefer opt-in so I am not surprised by unknown features in prototype.

Tim Jenness to Everyone (7:23 AM)

I also prefer opt in

Markus Demleitner to Everyone (7:23 AM)

Well, you're all more the "library authors", not the end users...

Tim Jenness to Everyone (7:23 AM)

You want people to acknowledge that they are using features that may change at any point because the standard is not complete

Xiuqin Wu (she/her) to Everyone (7:24 AM)

Tim Jenness +1

Mark Taylor to Everyone (7:24 AM)

End users might not mind a new feature appearing, but if the feature changes its behaviour or calling conventions unexpectedly that could be more of a nasty surprise that users might not want to encounter.

Tim Jenness to Everyone (7:24 AM)

If an application is using a prototype feature the application has to be prepared to catch the exceptions if things change and say that something changed. The server end might be changing a lot.

Markus Demleitner to Everyone (7:25 AM)

In my experience, it's rather rare for things to change significantly once they hit pyvo. And I'd highly prefer if as many people tried out standards in WD or PR as possible.

Hence, I'd like to make that as simple as possible.

Tim Jenness to Everyone (7:25 AM)

But isn't the point of this discussion that people want to start adding these prototypes to Pyvo sooner than before

Xiuqin Wu (she/her) to Everyone (7:26 AM)

for operations, I like a stable version

Pierre Fernique (CDS) to Everyone (7:26 AM)

+1 Markus comment.

Dave Morris to Everyone (7:29 AM)

+1 Markus compromise, different levels

Tim Jenness to Everyone (7:30 AM)

If something is automatically opted in it should issue prototype warning

Xiuqin Wu (she/her) to Everyone (7:31 AM)

I can support the compromise approach

Omar Laurino (he/him) to Everyone (7:31 AM)

Indeed, Tim, that's what my first implementation did.