

# Handling Domain Overlap in our models

Follow-up from Joint Session in Sydney

# Primer

## Getting started

- Our early models (eg: ObsCore, Spectrum, Char, STC) 'reused' elements from each other
  - This generally meant re-defining the content with summarized descriptions, referring to the parent model 'for more details'.
  - The models do not share schema, so the shared content is independently defined in each model's schema.
  - Which inevitably leads to inconsistencies in the representations of the same concepts.
  - Even when the definitions are exactly the same, as far as software is concerned, they are different elements. This makes it very hard to make software which is reusable in other contexts.

# Primer

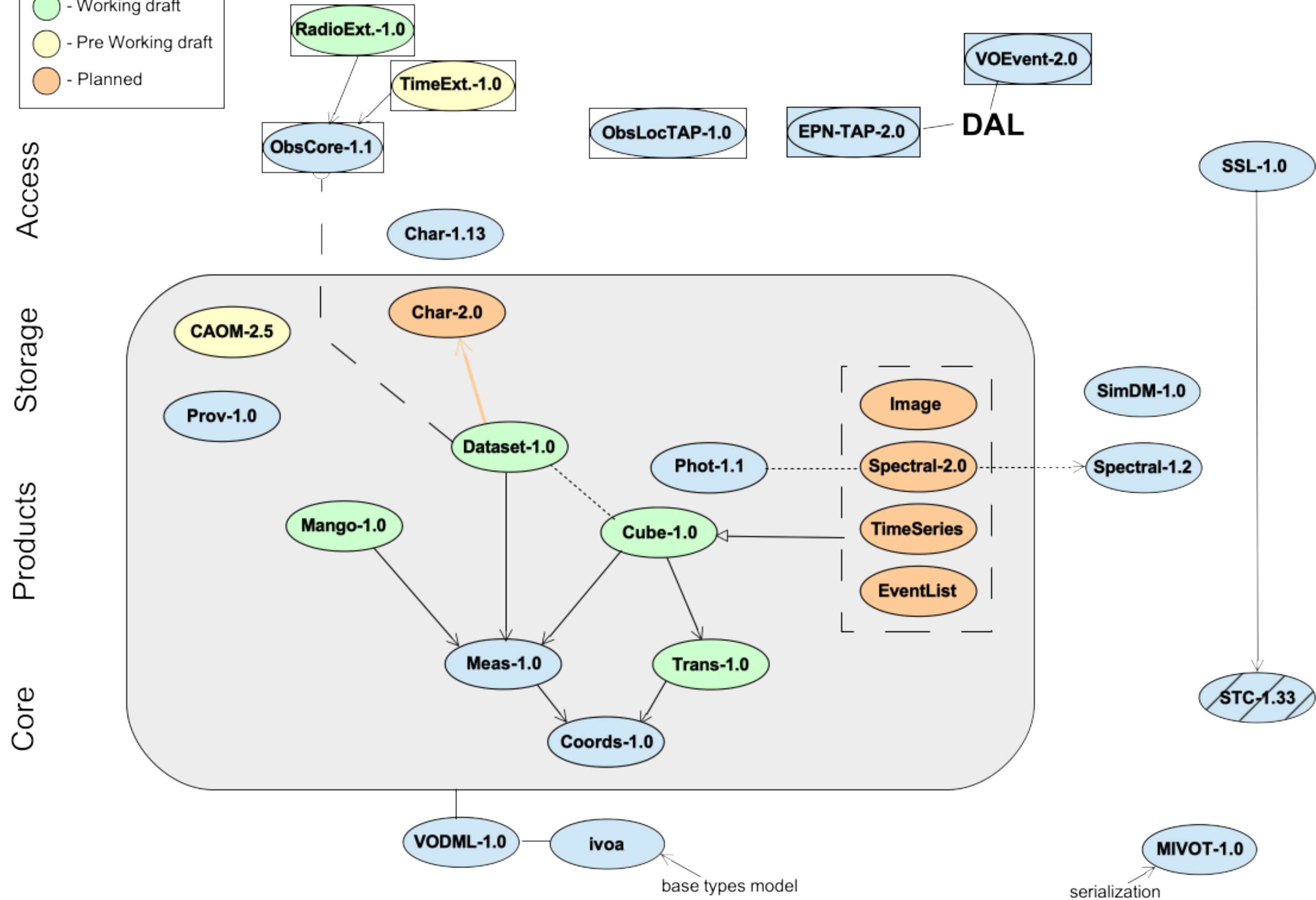
## A new paradigm

- The migration to VO-DML enabled us to truly ‘reuse’ content from other models.
  - Models ‘import’ content of models they wish to extend or use
  - While the documentation may still summarize the imported content to clarify the context in which it is being used, there is a single definition of the concepts in each model’s vodml/XML specification.
- This has enabled us to create a family of models which focus on specific areas of the domain space.

Key	
<span style="color: blue;">●</span>	- Recommendation
<span style="color: green;">●</span>	- Working draft
<span style="color: yellow;">●</span>	- Pre Working draft
<span style="color: orange;">●</span>	- Planned

## Data Models

Updated: Oct. 2024



# Primer

## Growing Pains

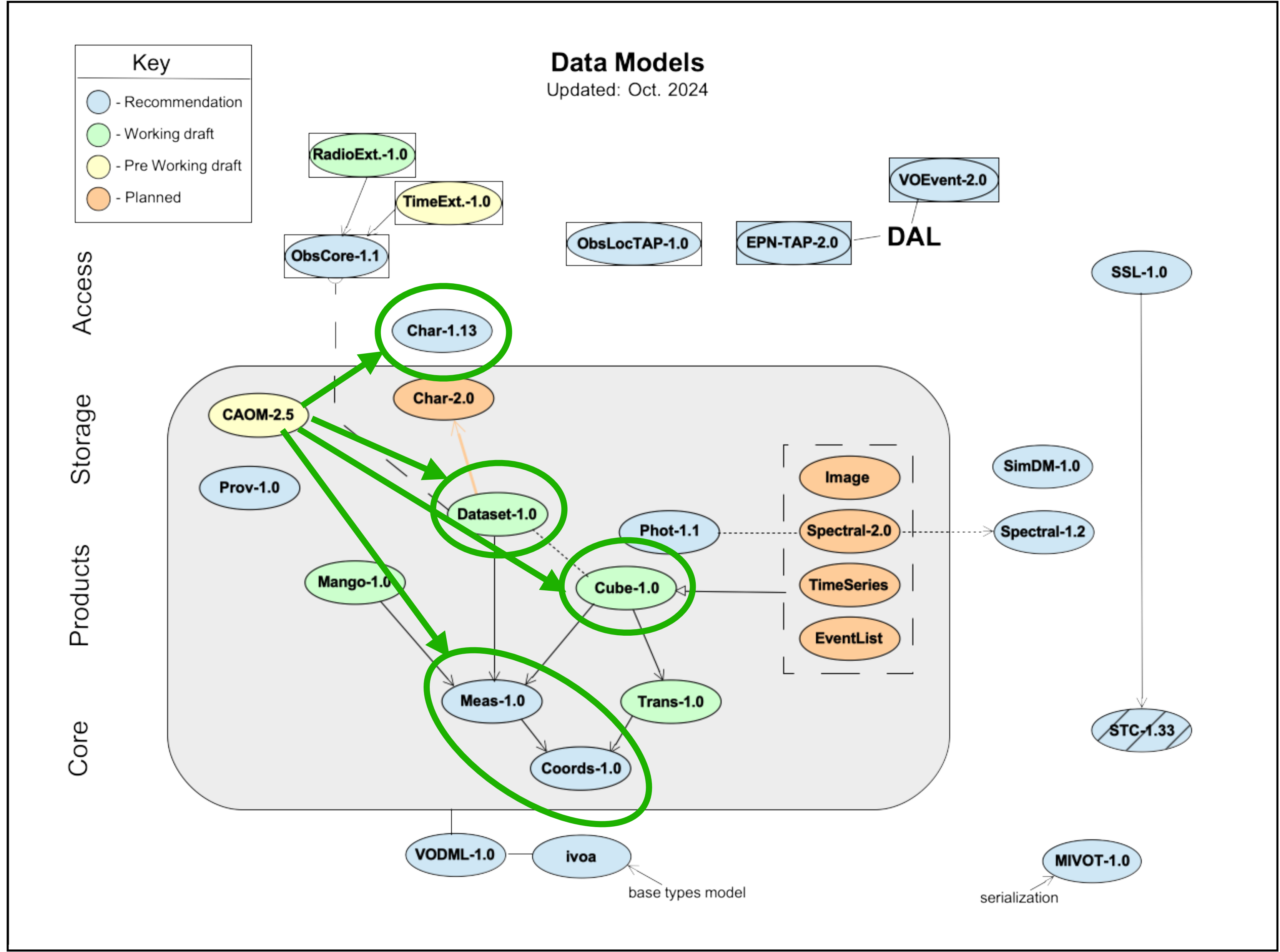
Recent projects involve models covering the same domain space as other models, but at a level suitable for their target users.

- The interest in folding CAOM into the IVOA Data Model landscape (PD: session 8)
- The interest in extending ObsCore to improve discovery of certain kinds of data; Radio, TimeDomain, High Energy ( Discussion: session 9 )
- Mango model support for the Epoch Propagation thread ( LM: session 4 )
- Ongoing work with ‘One-step Provenance’ to define a representation of Provenance suitable for usage within a dataset.
- Interest in reconciling the Shape concept which exists in STC-1.33 (Region), DALI-1.1 (base types), FOV-0.0 (Shape)



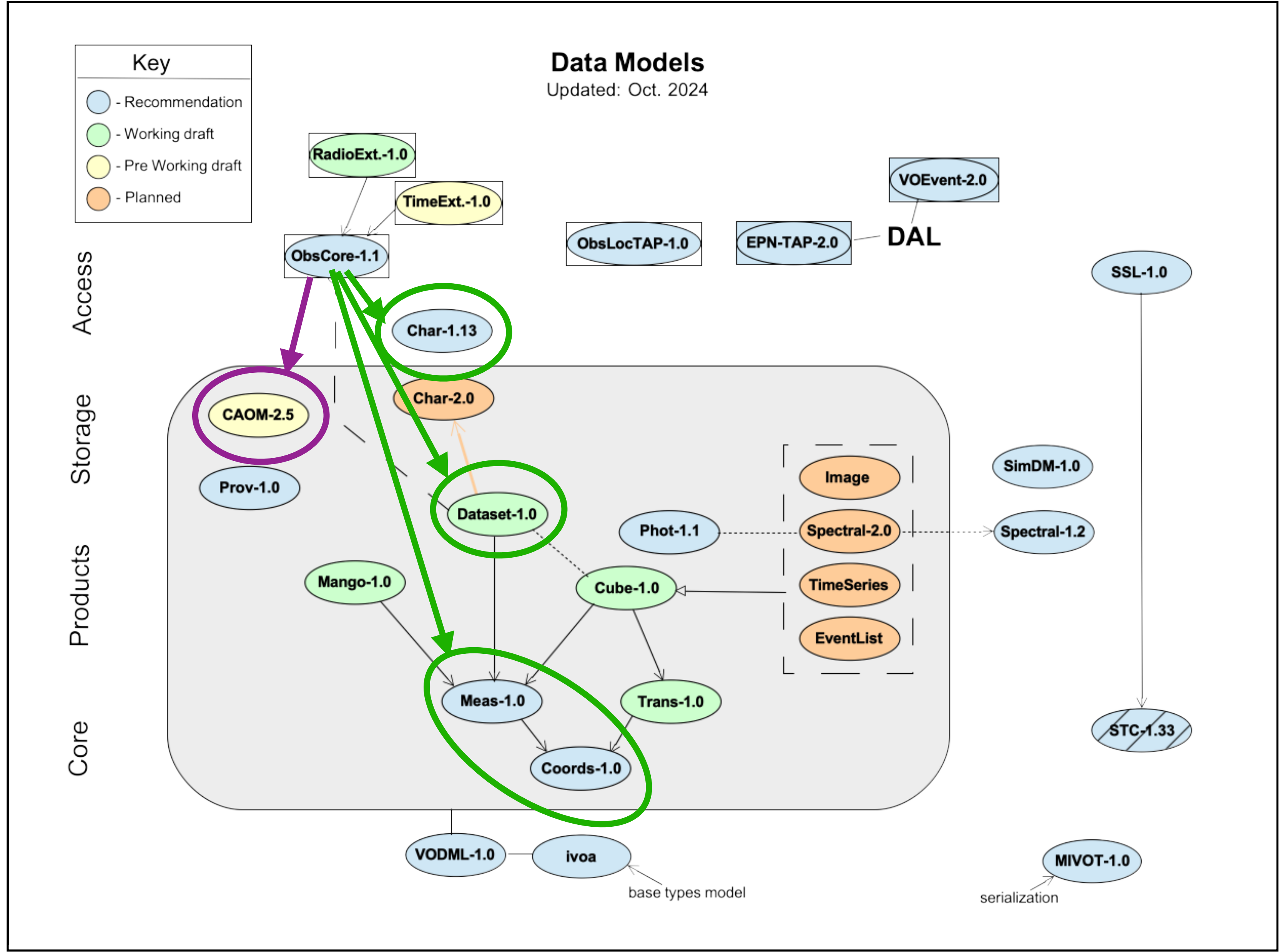
# Model Domain Overlap

- CAOM: model has conceptual overlap with several models at a level somewhere between data products and data discovery.



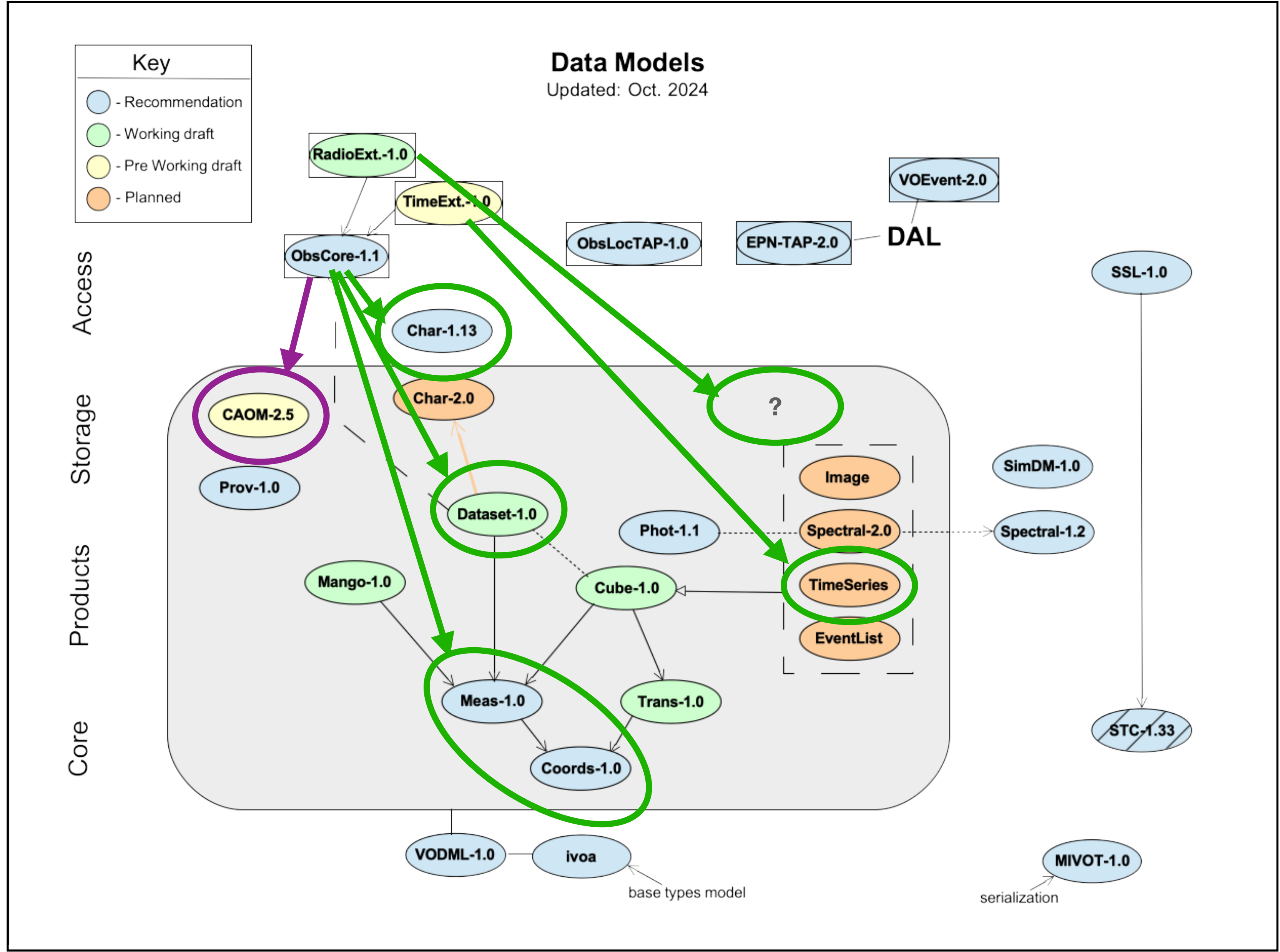
# Model Domain Overlap

- CAOM: model has conceptual overlap with several models at a level somewhere between data products and data discovery.
- ObsCore: has conceptual overlap with CAOM and several vo-dml models, but simplified to a level suitable for data discovery



# Model Domain Overlap

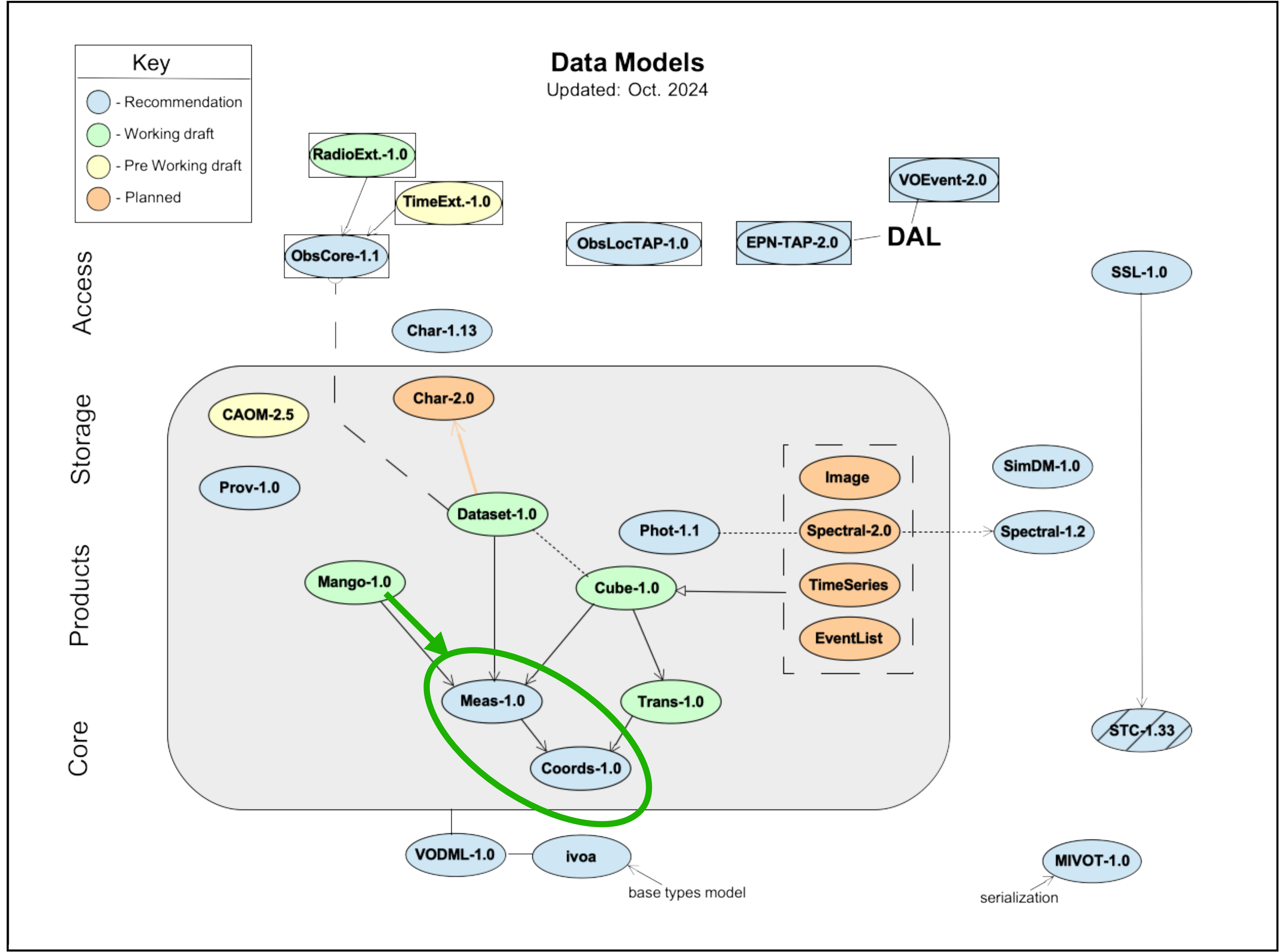
- CAOM: model has conceptual overlap with several models at a level somewhere between data products and data discovery.
- ObsCore: has conceptual overlap with CAOM and several vo-dml models, but simplified to a level suitable for data discovery
- ObsCore Extensions: presumably continue this trend





# Model Domain Overlap

- CAOM: model has conceptual overlap with several models at a level somewhere between data products and data discovery.
- ObsCore: has conceptual overlap with CAOM and several vo-dml models, but simplified to a level suitable for data discovery
- ObsCore Extensions: presumably continue this trend
- EpochPosition: consolidates content from core models to create a simpler object for an application thread.



# Statement of Purpose

## Why are we here?

- To continue a discussion about the PROCESS by which the IVOA data models serve their users (primarily DAL and APPS).
  - Satisfying the most sophisticated threads (eg: representing data products and catalogs), requires detailed and flexible data models.
  - However, MANY usage threads have much lighter requirements and those users want to deal with objects at the same level.
  - This is increasingly leading to models which describe the same concept domain at different levels. Without a formal relation of these models to their parent models, this will inevitably lead to inconsistencies. I believe this would be a serious hinderance for interoperability.
  - What strategies do we want to employ to enable these usage threads, but ensure model consistency?
  - How do we make it clear to users which models apply to their needs?

# Recent Work

- Updated Data Model Working group twiki page
  - In particular, updated the model ecosystem diagram to move toward making it easier to determine which models are targeted for which users
- Created a twiki page for this discussion topic:
  - <https://wiki.ivoa.net/twiki/bin/view/IVOA/DataModelConsistency>
  - Outlines the project, background information, use cases, examples.
  - Will build this into defining a set of requirements for any system we employ to relate elements derived from other models
- Working Epoch Propagation example (relatively small)
  - Manually mapping the Mango model element to corresponding Meas/Coords elements.

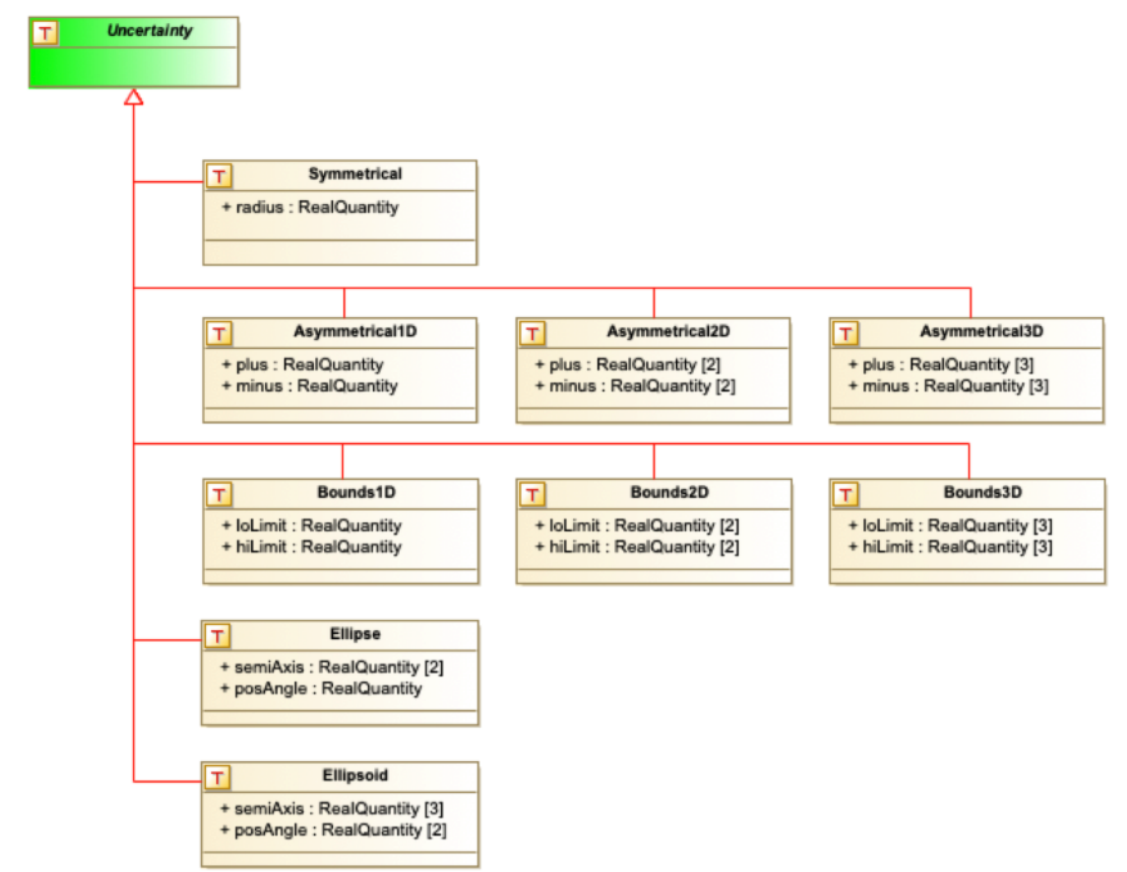
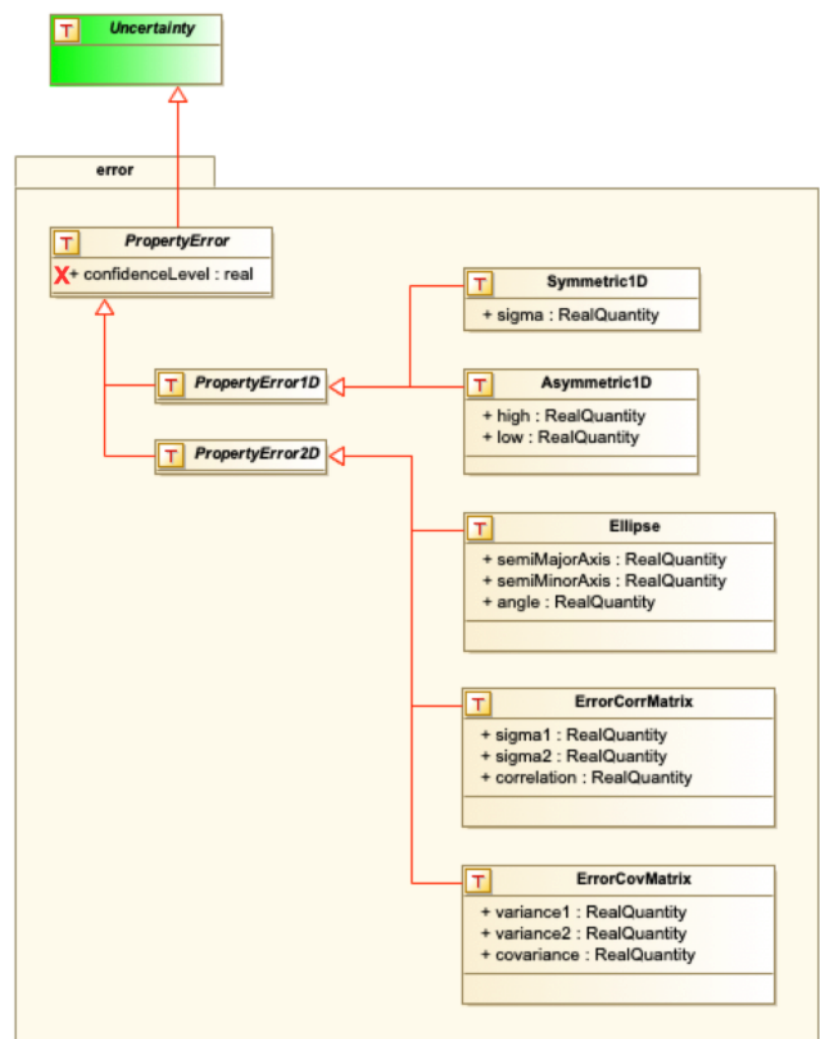
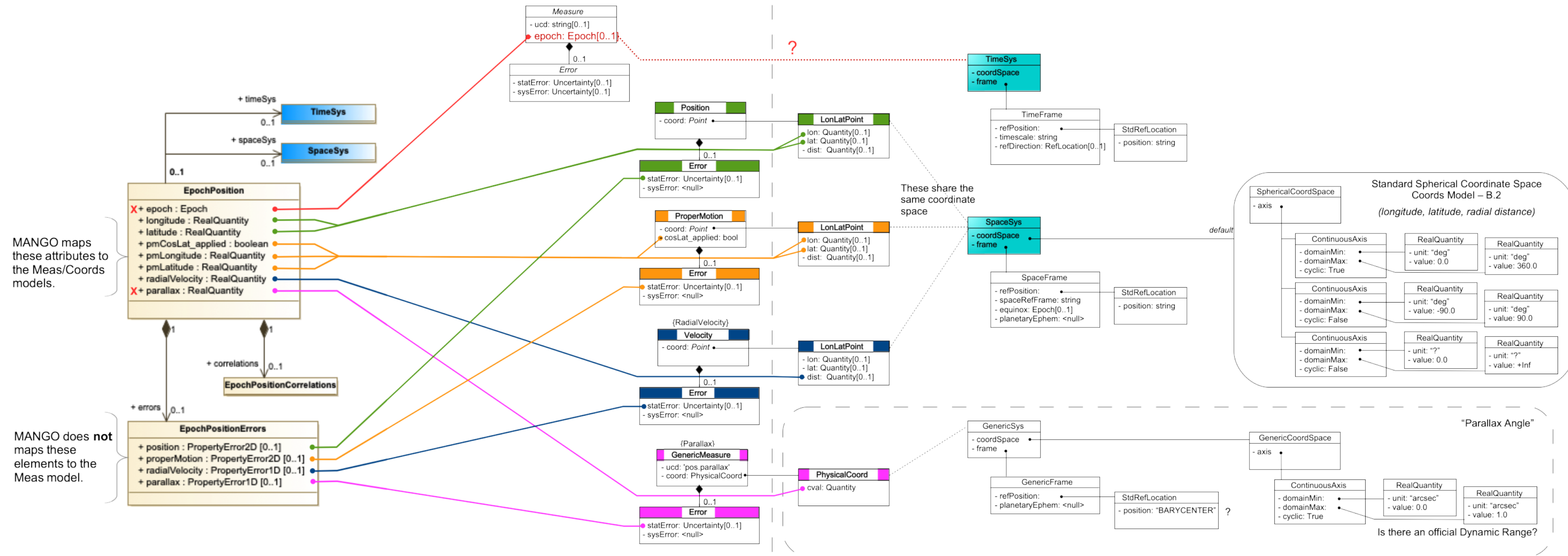


# Mango

# Meas

# Coords

GAIA Source database table  
Possible representation with current (V1.0) Meas and Coords models  
MCD – Dec 2023

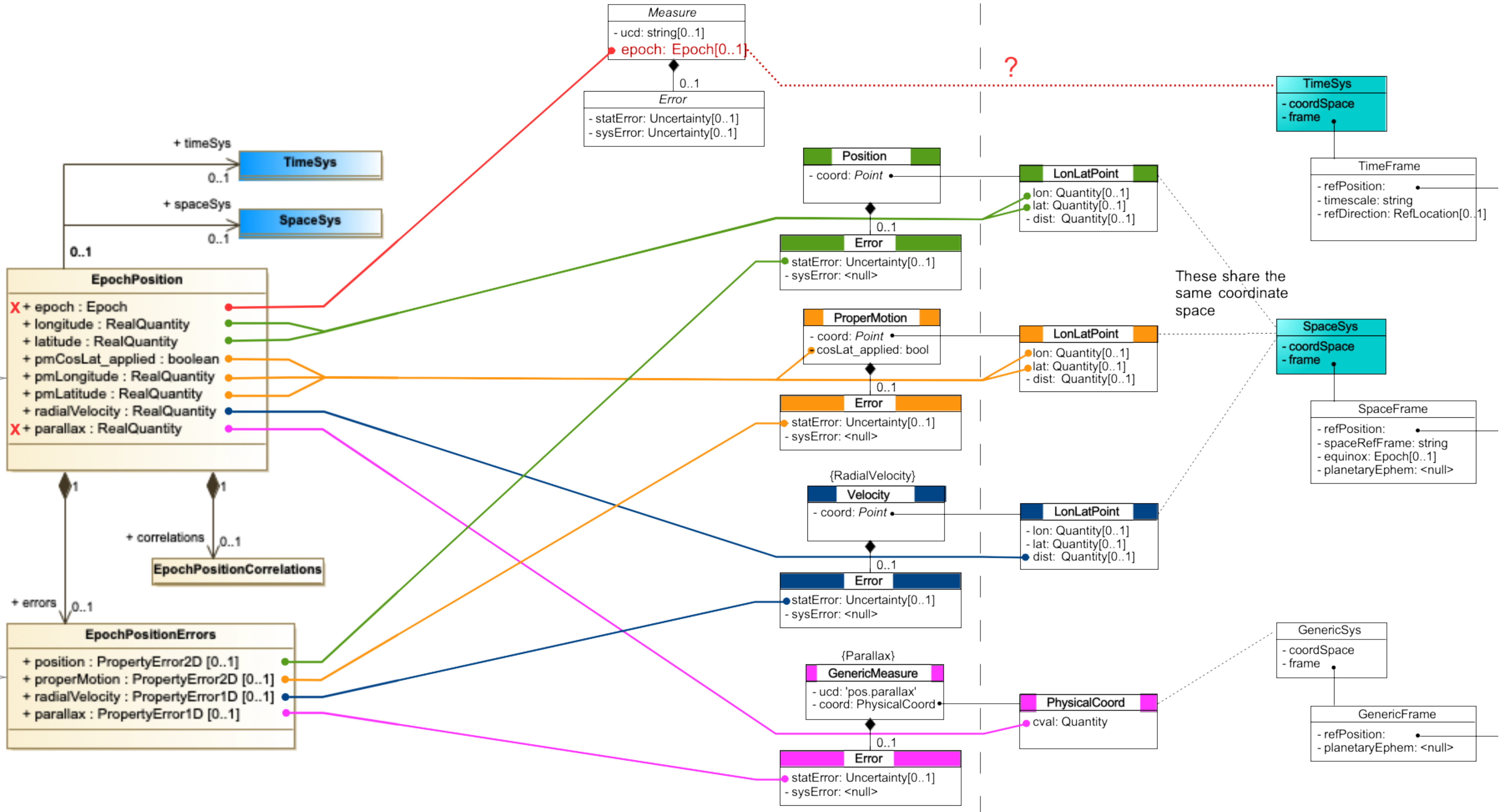




# Mango

# Meas

# Coords



MANGO maps these attributes to the Meas/Coords models.

MANGO does **not** map these elements to the Meas model.

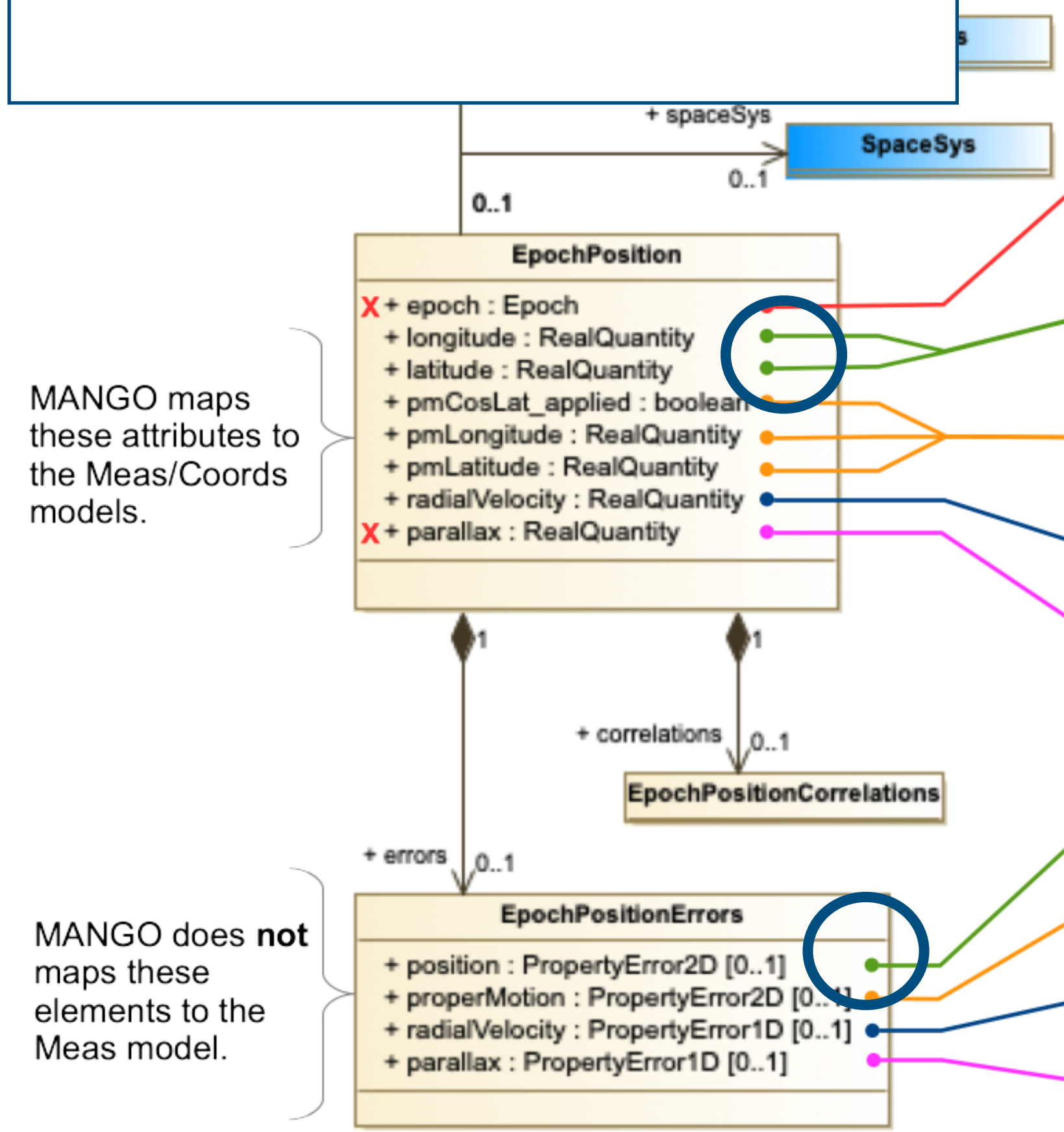
These share the same coordinate space

# Mango

# Meas

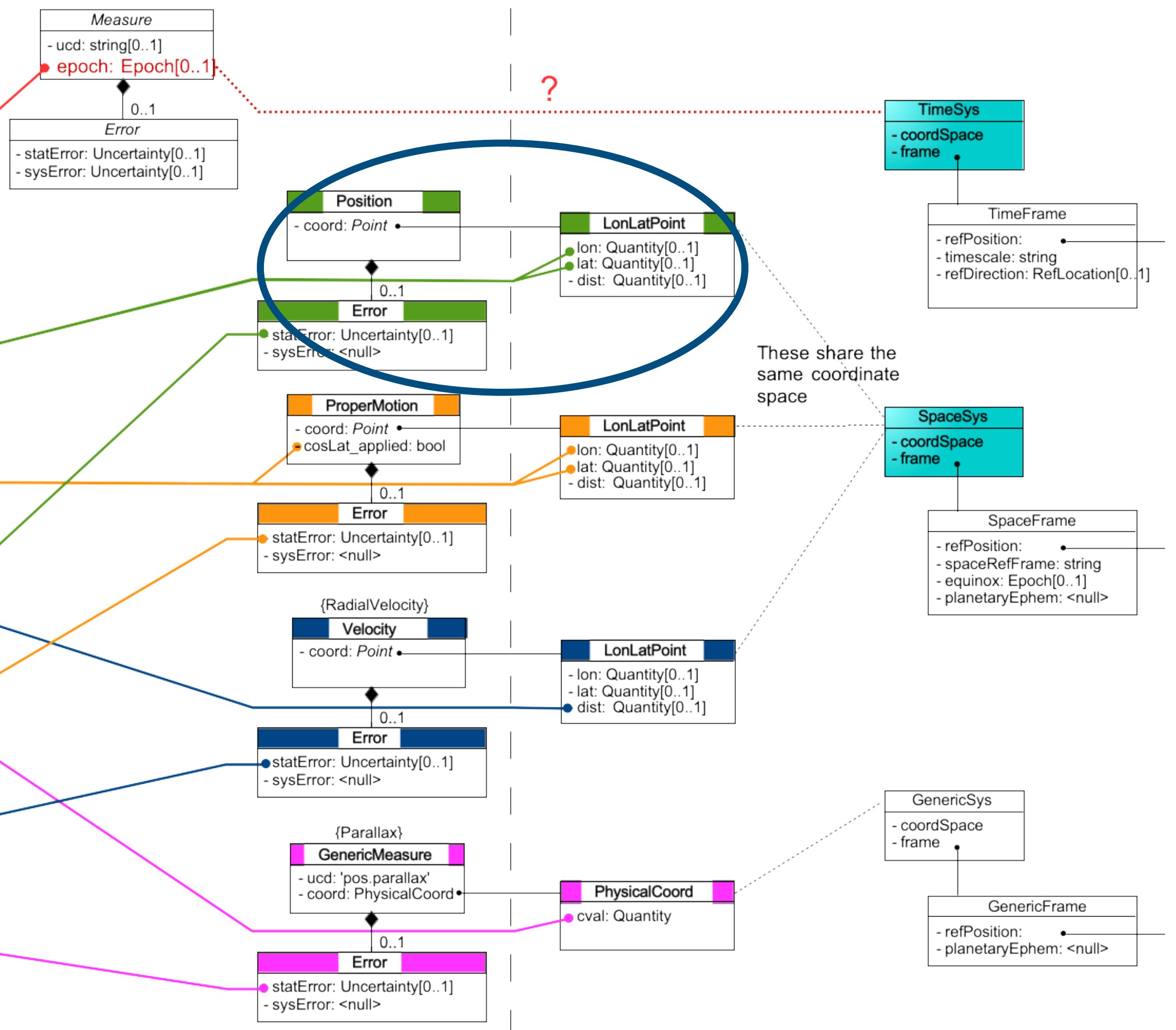
# Coords

**Derived Requirements**  
 \* group leaves contributing to parent object  
 \* may come from multiple child objects



MANGO maps these attributes to the Meas/Coords models.

MANGO does **not** map these elements to the Meas model.



These share the same coordinate space

?



# Mango

# Meas

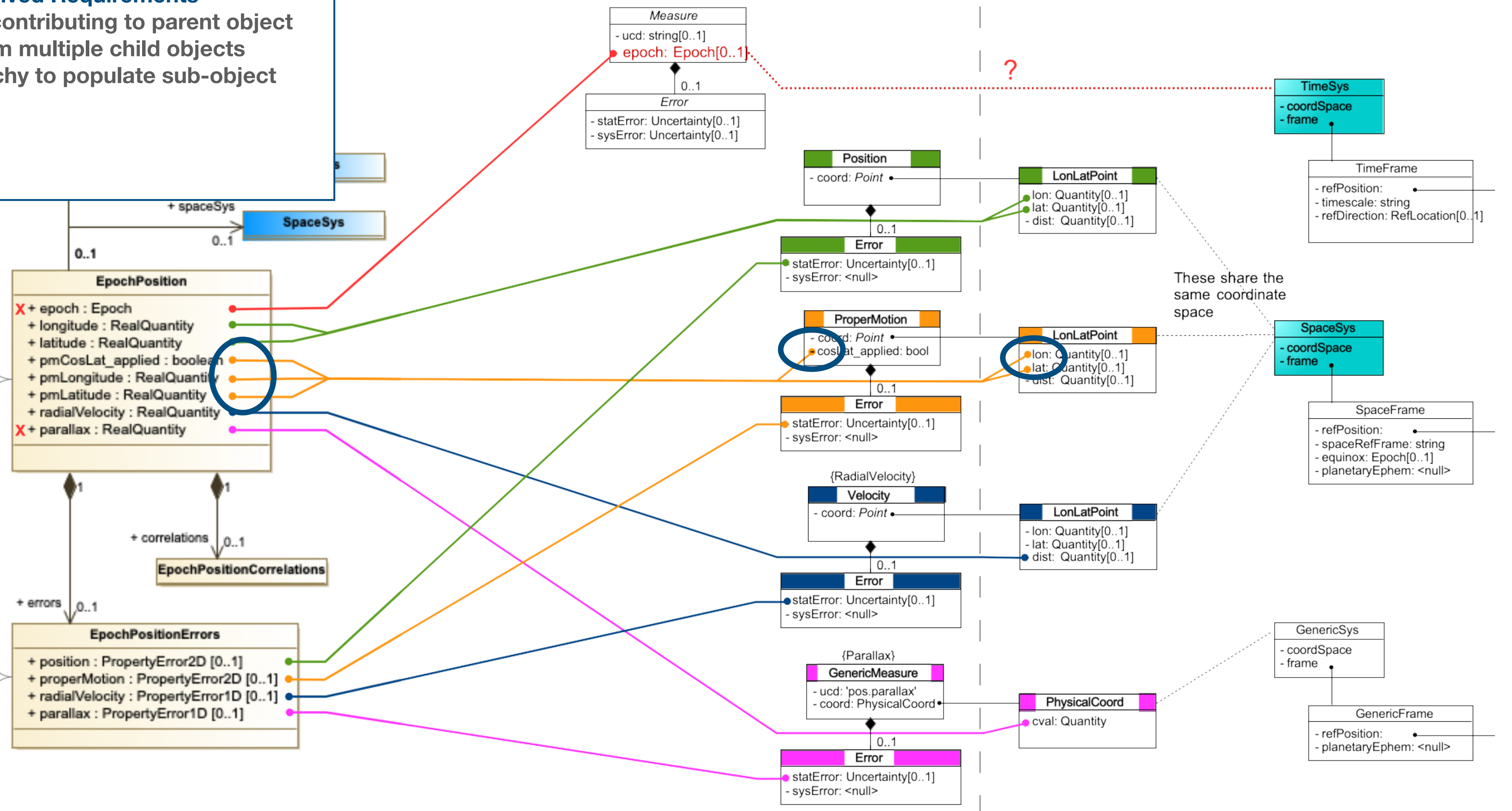
# Coords

**Derived Requirements**

- \* group leaves contributing to parent object
- \* may come from multiple child objects
- \* include hierarchy to populate sub-object

MANGO maps these attributes to the Meas/Coords models.

MANGO does **not** map these elements to the Meas model.



# Mango

# Meas

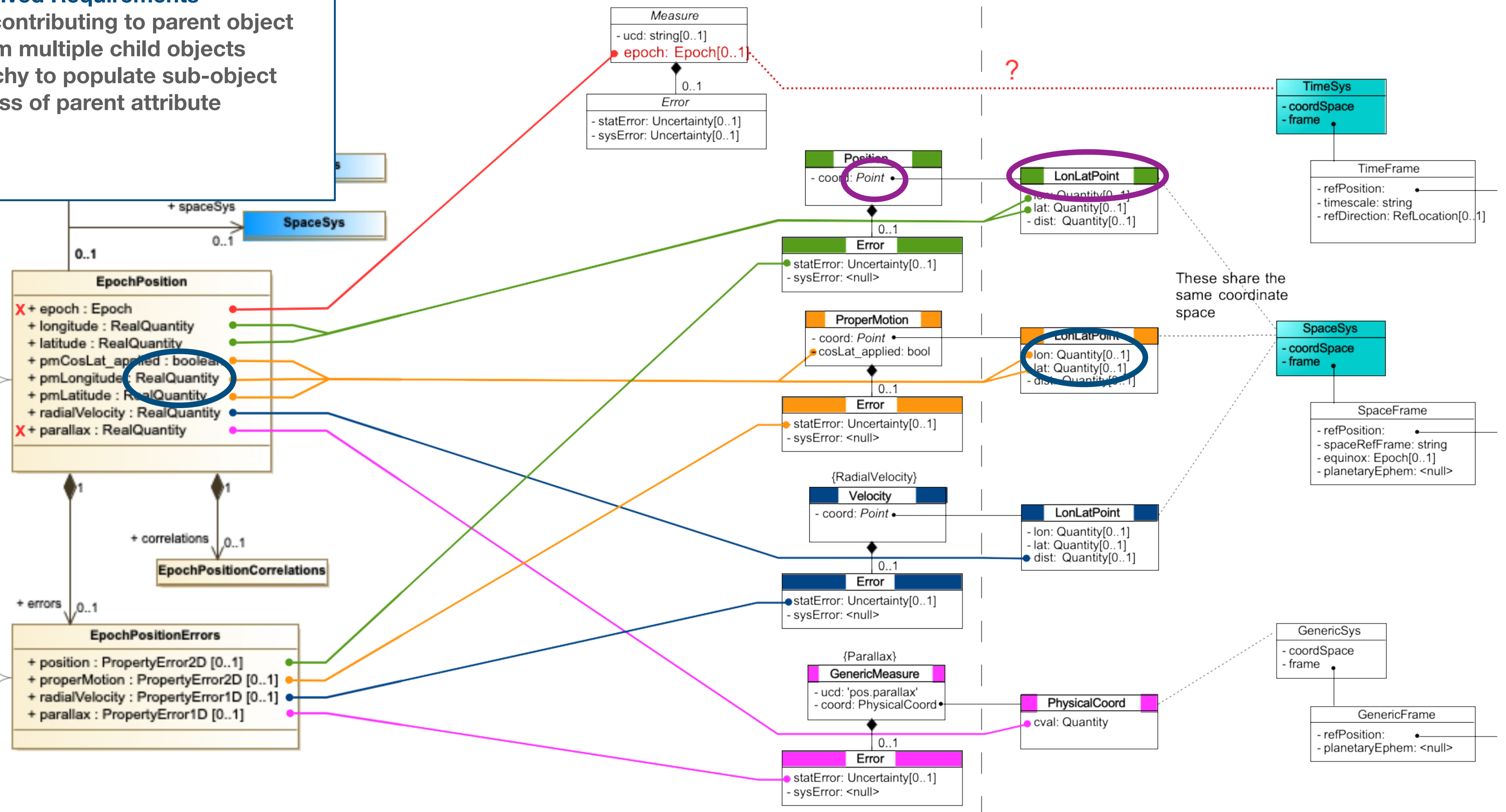
# Coords

**Derived Requirements**

- \* group leaves contributing to parent object
- \* may come from multiple child objects
- \* include hierarchy to populate sub-object
- \* restrict subclass of parent attribute

MANGO maps these attributes to the Meas/Coords models.

MANGO does **not** map these elements to the Meas model.





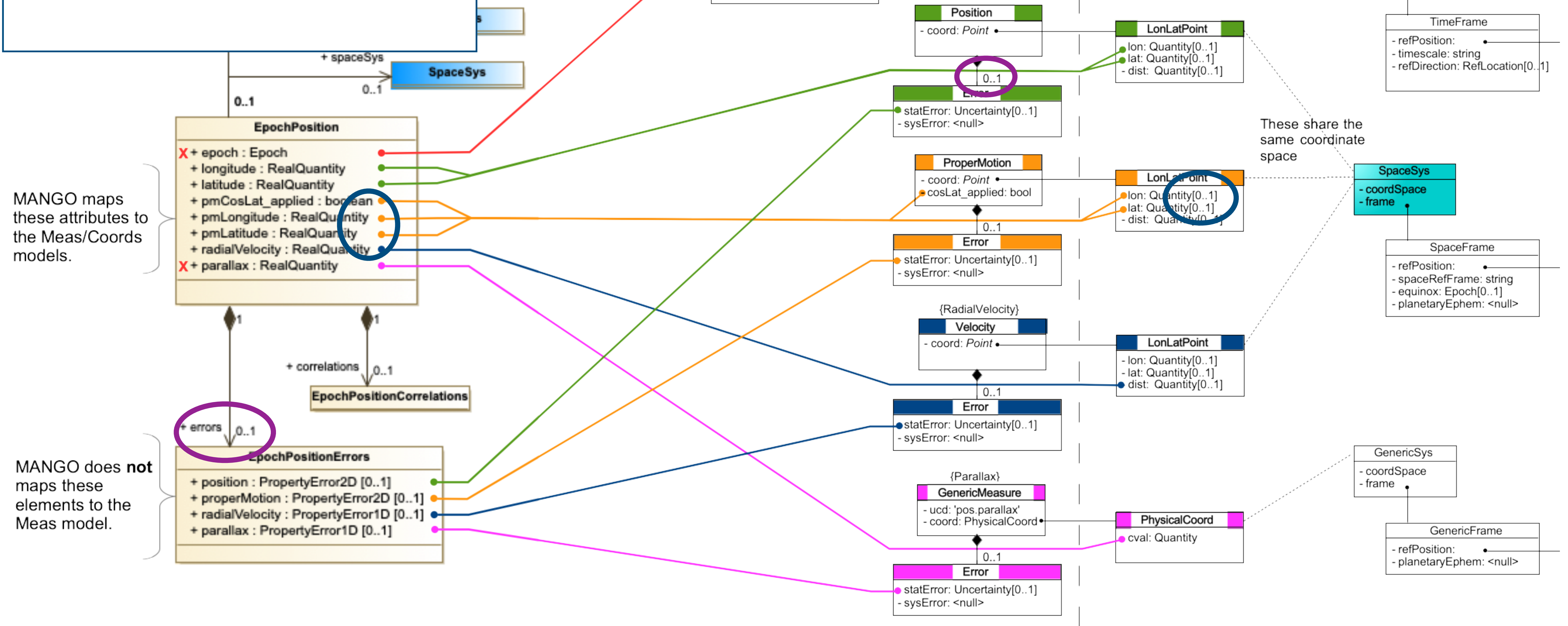
# Mango

# Meas

# Coords

## Derived Requirements

- \* group leaves contributing to parent object
- \* may come from multiple child objects
- \* include hierarchy to populate sub-object
- \* restrict subclass of parent attribute
- \* multiplicity must be compatible with parent



MANGO maps these attributes to the Meas/Coords models.

MANGO does not map these elements to the Meas model.

# Summary

- We NEED to serve models at levels appropriate for the targeted user
- Which opens the door for inconsistency.. which we REALLY don't want!
- Manually mapping the relations between models will be hard/time-consuming work (at least at first).
- A vo-dml like meta model for describing these would be helpful by making it a machine readable relation.. BUT
- There may already be something out there which does this.. (??).
- We don't want to re-invent the wheel.

# Moving Forward

- Please attend the upcoming sessions
  - CAOM Session: today at 16:00
  - ObsCore Radio Extension: tomorrow at 9:00
- Participate in the discussion.. this will greatly inform the path forward.
- Will be continuing the discussion on the DM mail list.