International

Virtual

Observatory

Alliance

STC-X:
Space-Time Coordinate (STC) Metadata XML Implementation

Version 1.00

IVOA Note 15 March 2005

This version:
http://www.ivoa.net/Documents/Notes/STC-X/STC-X-20050315.html
Latest version:
http://www.ivoa.net/Documents/latest/STC-X.html
Previous version(s):
None

Author:
A. H. Rots

Abstract

The XML implementation of the Space-Time Coordinate (STC) metadata is described through UML-like diagrams.

Status of This Document

This is a Note. The first release of this document was 15 March 2005.

This is an IVOA Note expressing suggestions from and opinions of the authors. It is intended to share best practices, possible approaches, or other perspectives on interoperability with the Virtual Observatory. It should not be referenced or otherwise interpreted as a standard specification.

A list of current IVOA Recommendations and other technical documents can be found at http://www.ivoa.net/Documents/.

Acknowledgements

The diagrams were created using Altova's XMLSpy.

Contents

1 Introduction

In this Note the XML implementation of the Space-Time Coordinate metadata (STC) is illustrated through the UML-like diagrams created by XMLSpy.

The corresponding XML schemata may be found at:

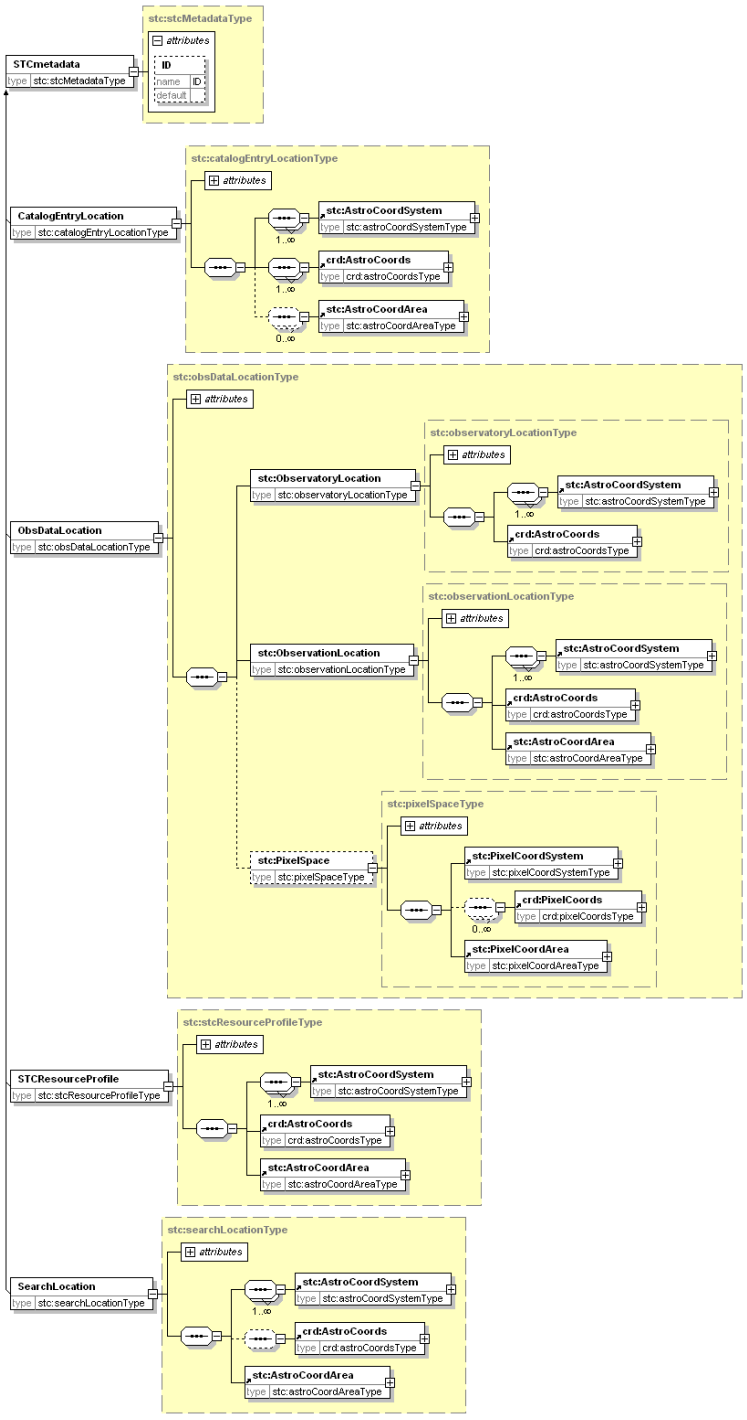
- http://www.ivoa.net/xml/STC/v1.20
  http://hea-www.harvard.edu/~arots/nvometa/v1.2/stc-v1.20.xsd
  is the top-level schema of the XML implementation STC-X, containing the coordinate system and coordinate area definitions.
- http://www.ivoa.net/xml/STC/STCcoords/v1.20
  http://hea-www.harvard.edu/~arots/nvometa/v1.2/coords-v1.20.xsd
  is the STC-X schema with the coordinate value definitions.
- http://www.ivoa.net/xml/STC/STCregion/v1.20
  http://hea-www.harvard.edu/~arots/nvometa/v1.2/region-v1.20.xsd
  is the STC-X schema for the *region* specification.

The definition of STC and the linear string implementation STC-S (based on the current document) are referenced at the end of this document.

2 Top-level Metadata Elements

2.1 STCmetadata

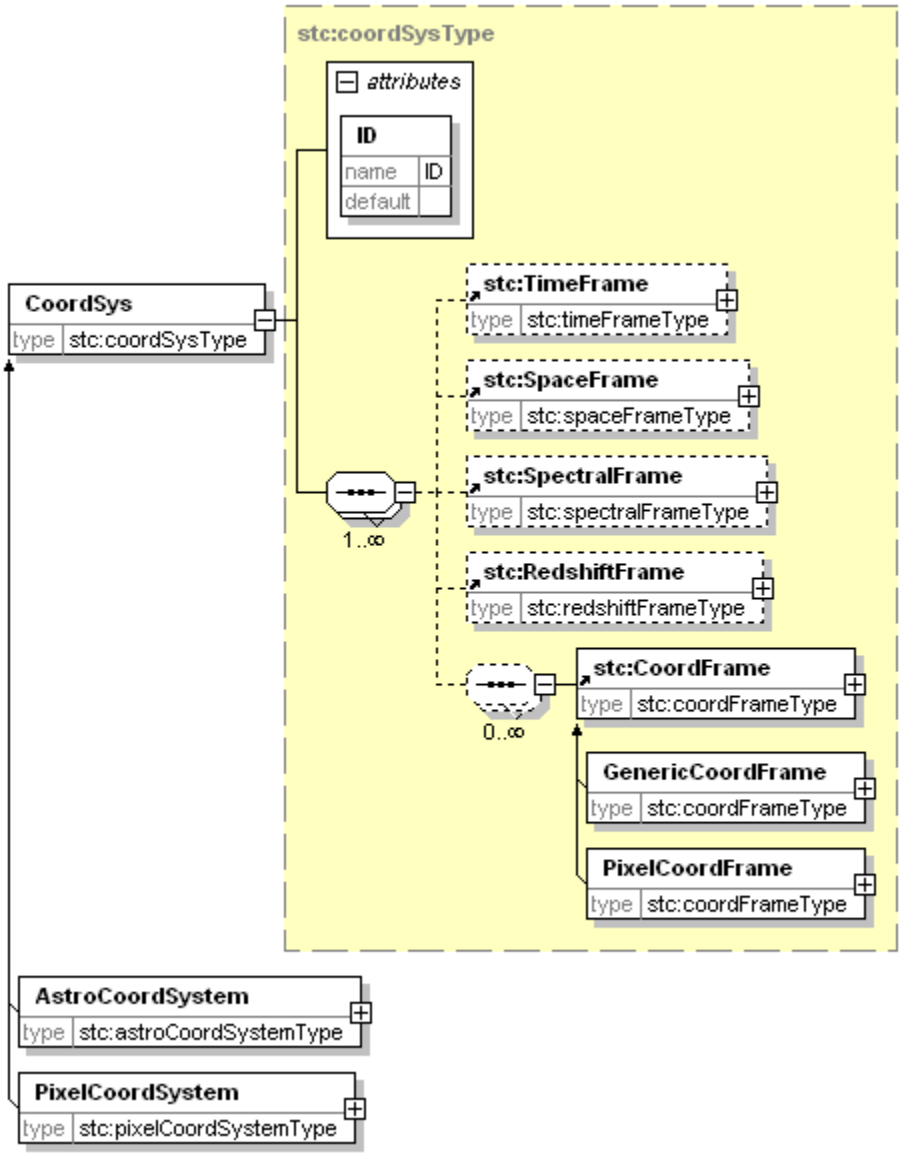
There are currently four types of STCmetadata elements derived, all based on the stcDescriptionType consisting of three elements: CoordSys, Coords, and CoordArea.



3 Coordinate Systems

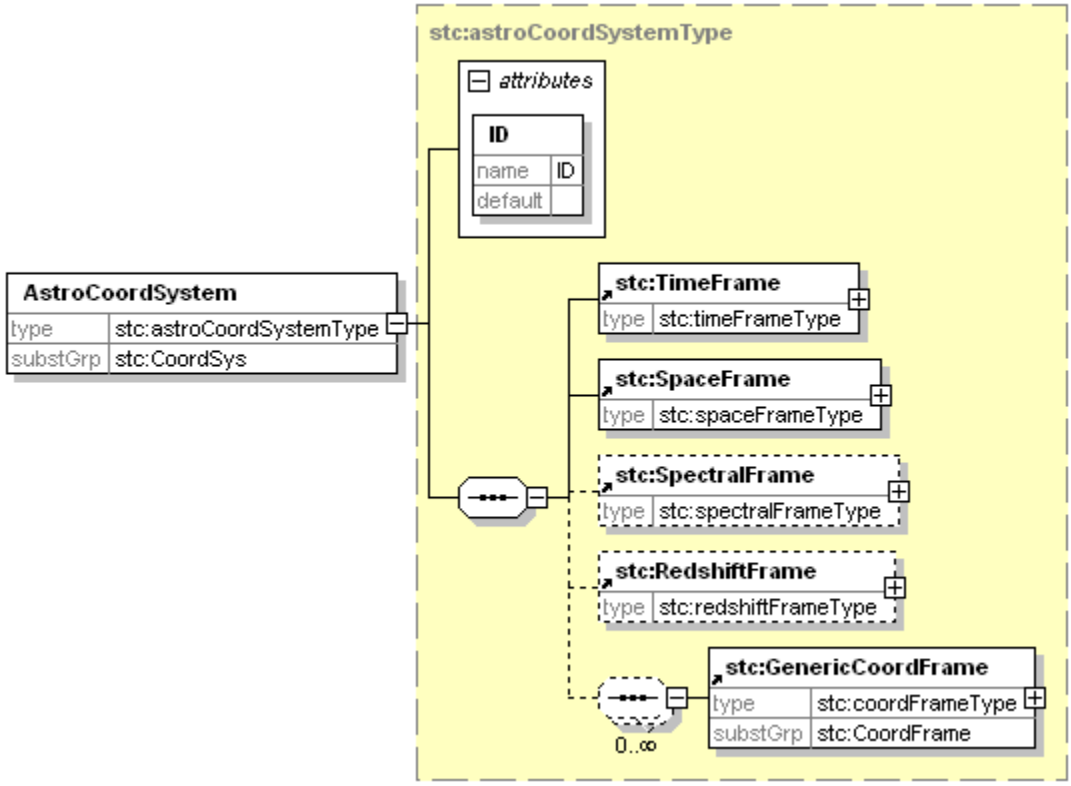
3.1 CoordSys

Basic CoordSys contains one or more CoordFrames and is required to have an ID.
There are 5 specific Frame implementations and 1 generic Frame.
There are two specific coordinate systems derived from CoordSys: AstroCoordSystem and PixelCoordSystem.



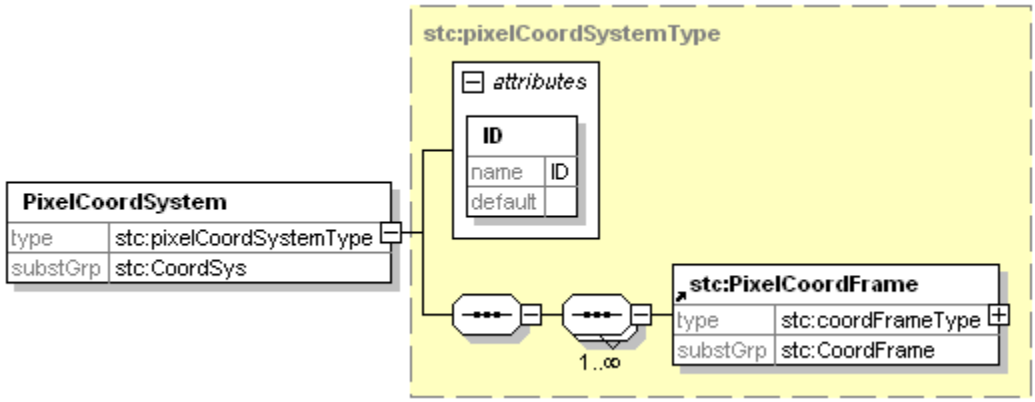
3.2 AstroCoordSys

AstroCoordSystem contains 3 required Frames, an optional Frame, and may contain any number of generic Frames.
It is derived from coordSysType by restriction.



3.3 PixelCoordSys

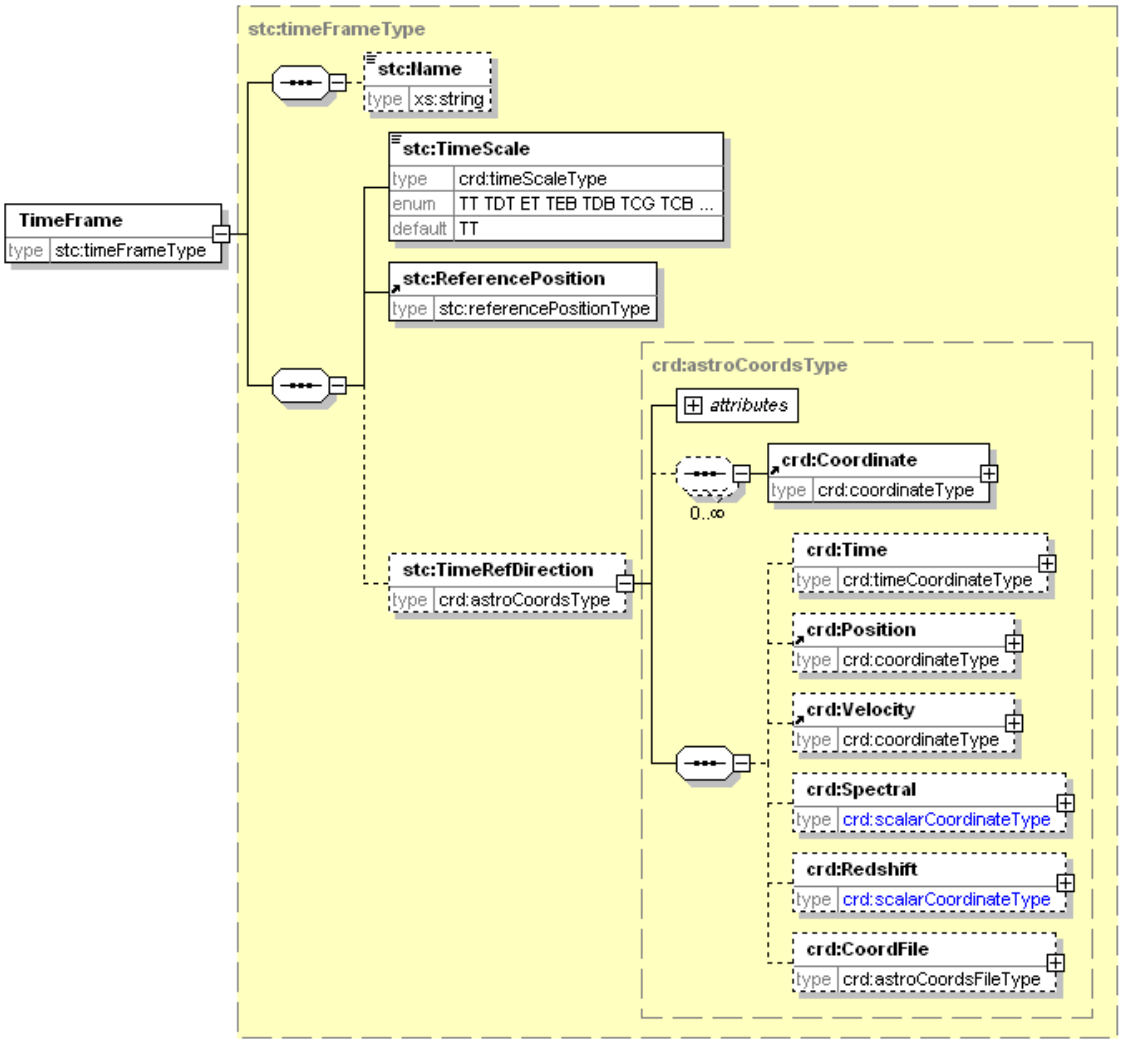
PixelCoordSystem is made up of Frames that only have a name



4 Coordinate Frames

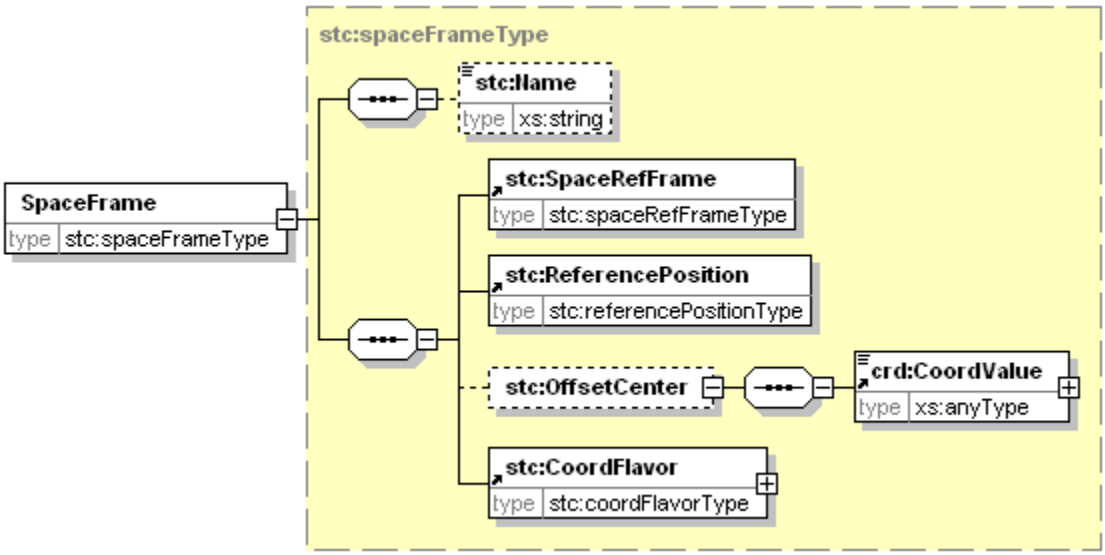
4.1 TimeFrame

The TimeFrame:



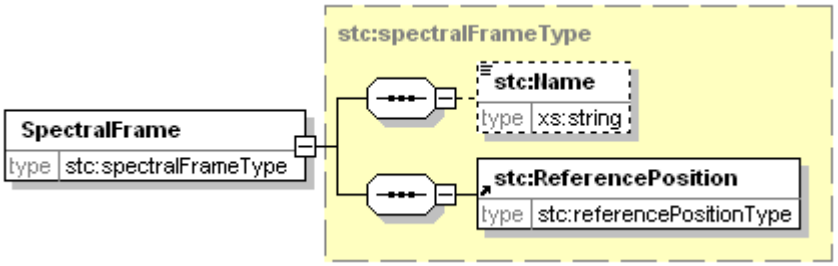
4.2 SpaceFrame

The OffsetCenter is added to accommodate lists with RA and Dec offsets.
Note that this is intended for pure numeric RA and Dec offsets; true angle offsets should be handled through a CustomSpaceRefFrame.



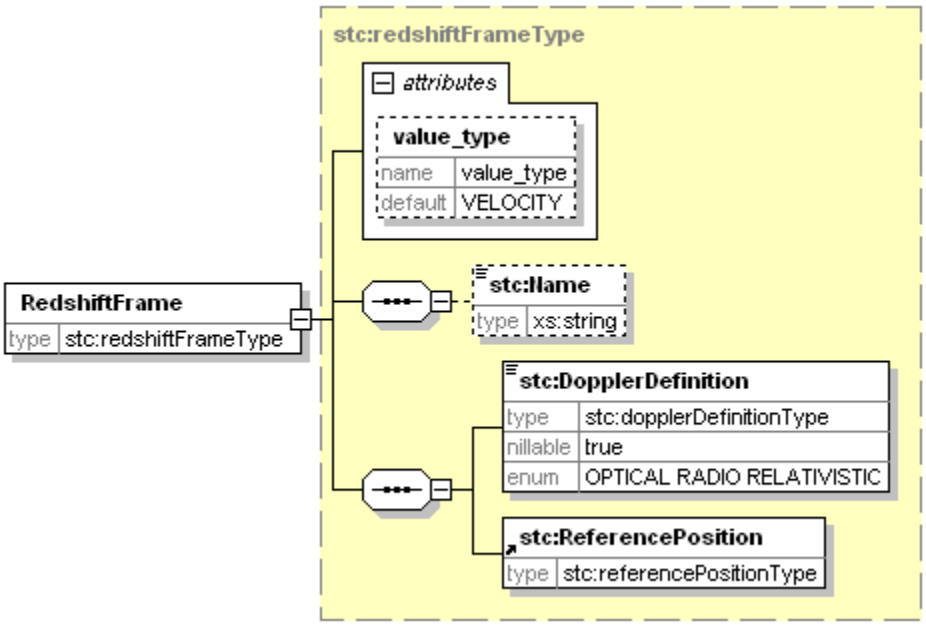
4.3 SpectralFrame

The SpectralFrame is simple:



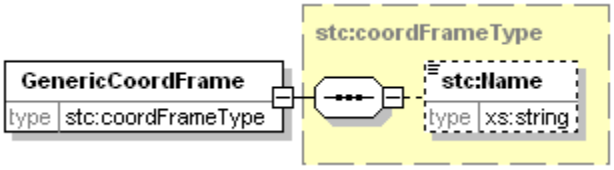
4.4 RedshiftFrame

The RedshiftFrame has a little more:



4.5 GenericCoordFrame and PixelCoordFrame

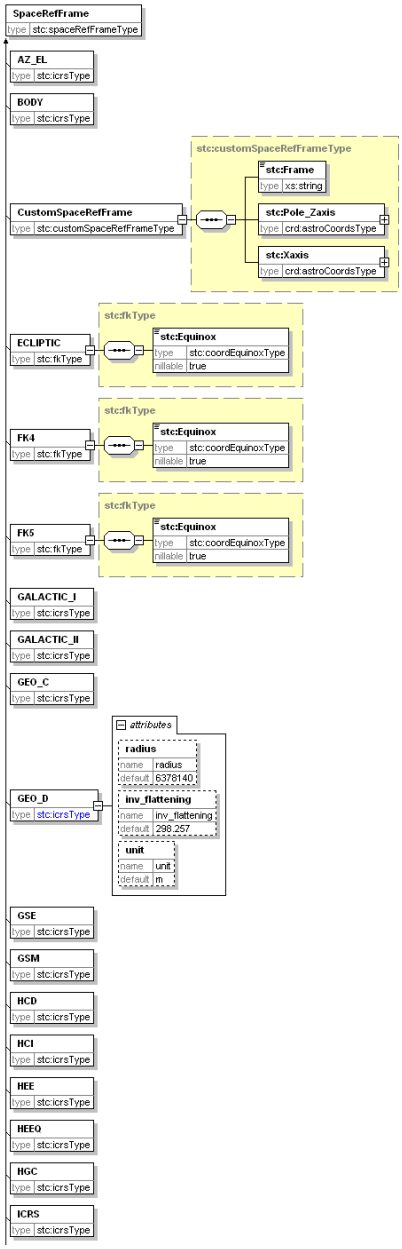
The GenericCoordFrame (and PixelCoordFrame) just have a name:



5 Coordinate Frame Components

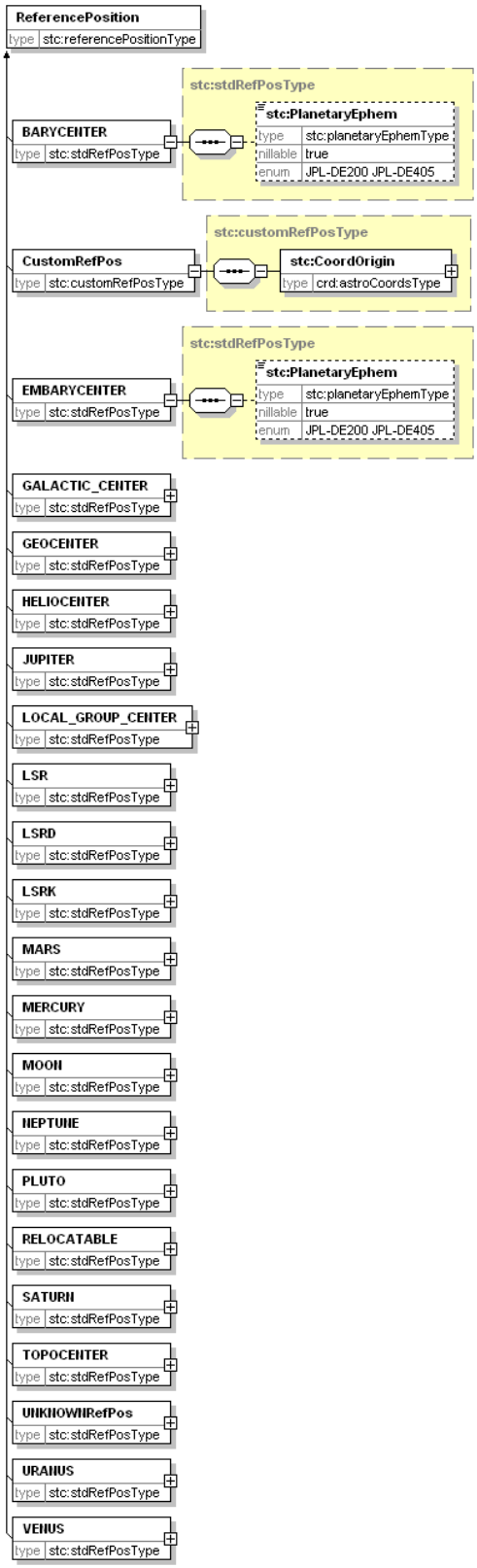
5.1 SpaceRefFrame

The SpaceReferenceFrame (part of SpaceFrame) uses the names of the elements in the SpaceRefFrame substitution group to identify the frame, rather than using a single element with enumerated values; this makes it easier to extend the list – by just adding members to the substitution group – such as solar reference systems.
Note that ICRS does not have an Equinox element. The figure below is truncated.



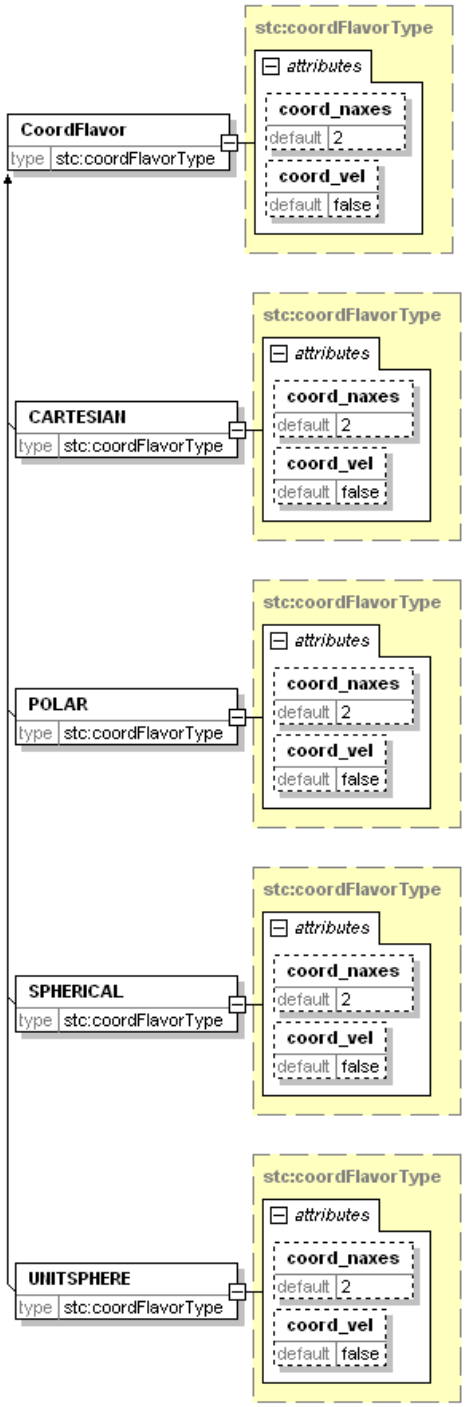
5.2 ReferencePosition

The ReferencePosition uses the same technique:



5.3 CoordFlavor

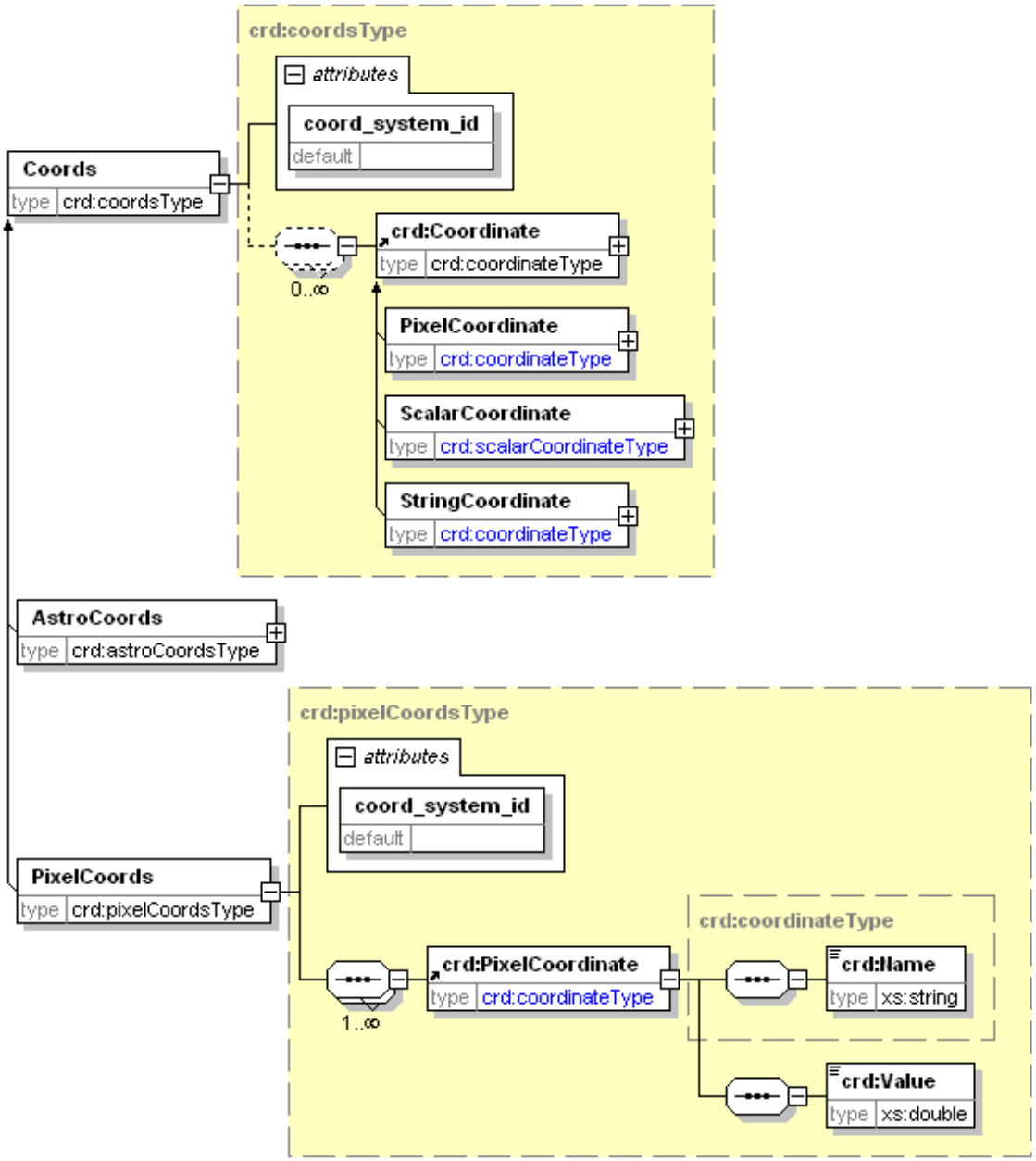
The CoordFlavor also determines the flavor by the names of the substitution group's elements and is similarly easily extensible.
In addition, these elements have two attributes: the number of axes (coord_naxes, default 2) and a Boolean indicating whether velocities are present (coord_vel, default false)



6 Coordinates

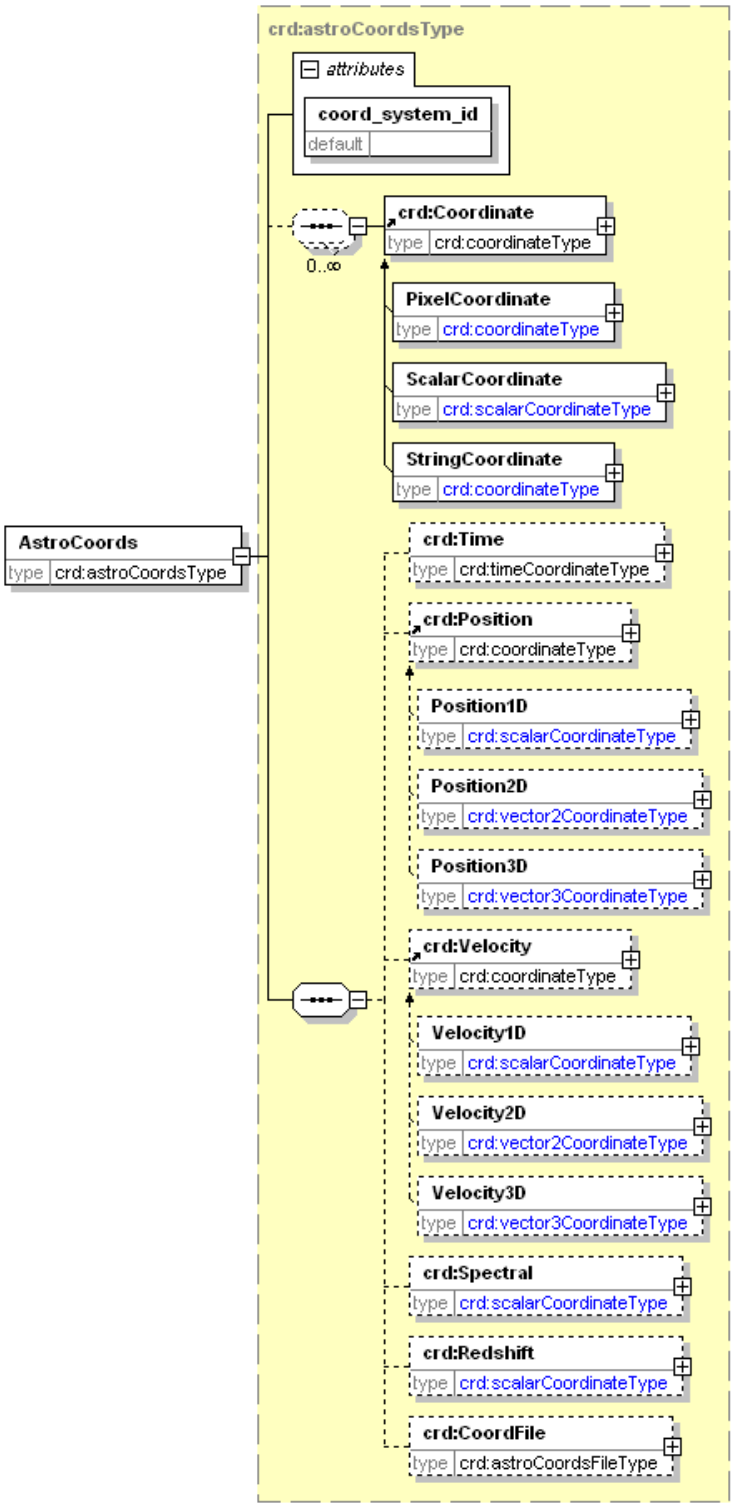
6.1 Coords

Coords is the generic coordinates element. It needs to refer to a specific instance of a CoordSys.
There are two derived classes: AstroCoords (by extension) and PixelCoords (by restriction).



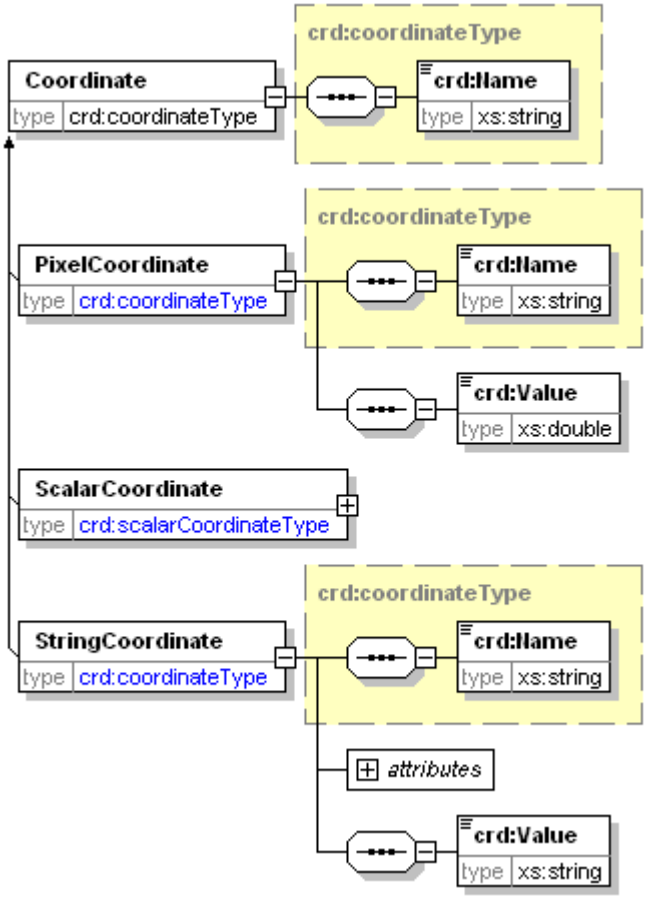
6.2 AstroCoords

AstroCoords contains specific astronomical coordinates (time, space, spectral, redshift) and, optionally, generic coordinates



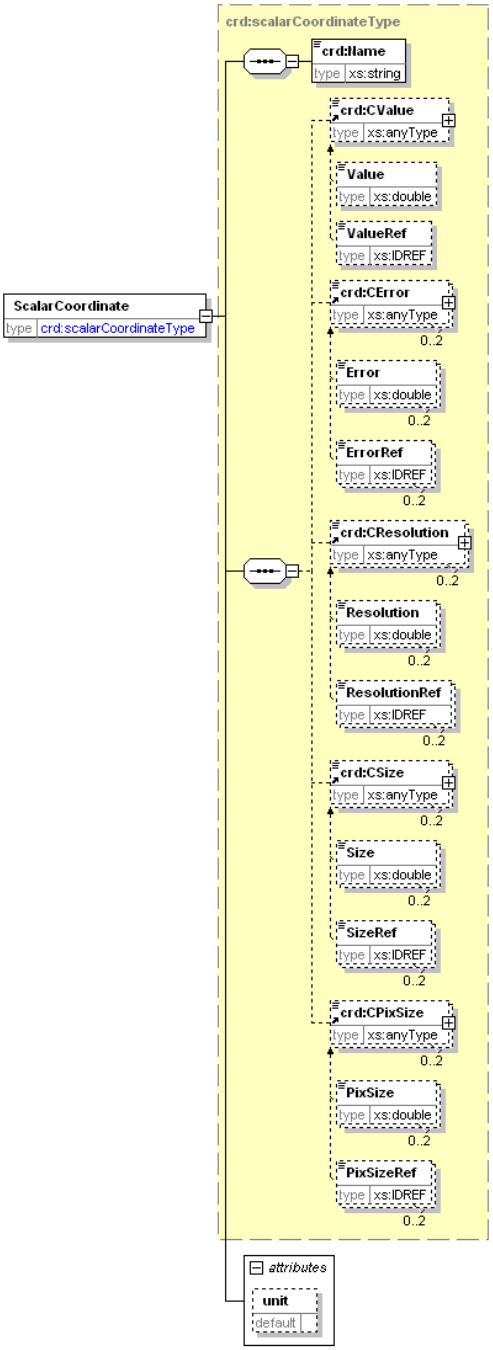
6.3 Coordinate

The simple Coordinate just has a Name. Its simplest derived classes add only a Value (PixelCoordinate) or a Values and an optional Unit (StringCoordinate). ScalarCoordinate is the generalized full coordinate type.



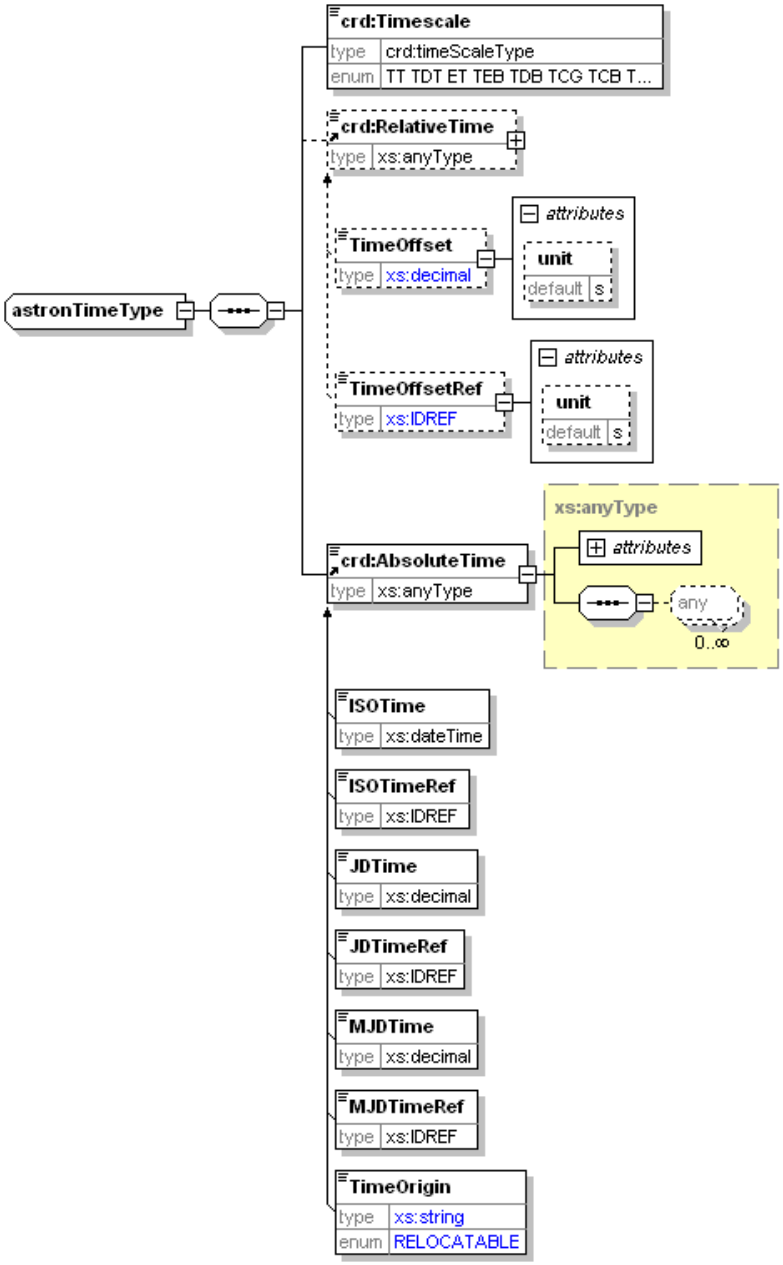
6.4 ScalarCoordinate

ScalarCoordinate, in addition to the Name, has an optional Unit and the full 5 coordinate components (Value, Error, Resolution, Size, and PixelSize), each of which may be an actual value or an IDREF pointing to another element in the document. Not all components need to be present, but at least one of them ought to. Note that components other than Value may appear in pairs, which would indicate a range.



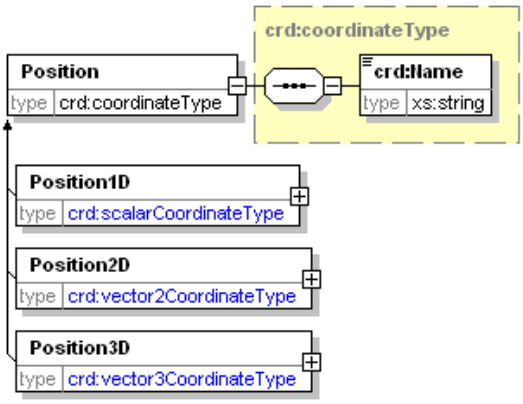
6.5 AstronTime

AstronTime is especially designed for astronomical application. It needs to contain a Timescale and an absolute time. The latter may be expressed in a subset of ISO-8601 or as Julian or Modified Julian Day. In all cases this may be an actual value or an IDREF to another element in the document. Note that (M)JD values are xs:decimal in order to achieve necessary precision. RELOCATABLE TimeOrigin s especially for simulations. In addition, time may be expressed as relative (elapsed) time in which case the AbsoluteTime element provides the reference time; in such a situation will will almost certainly want to use an IDREF for one of AbsoluteTime's derived types.

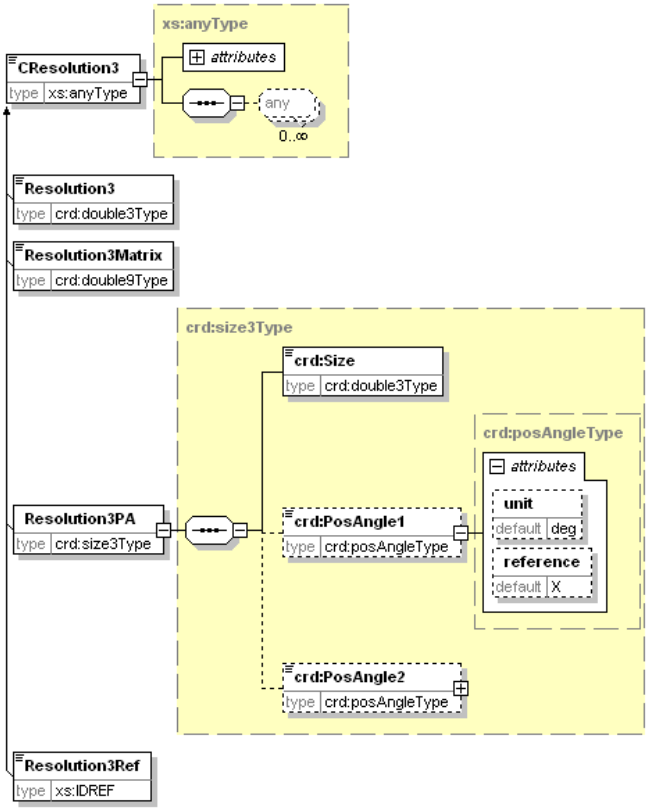


6.6 SpatialPosition

Spatial Position is more complicated Coordinate type since it refers to a compound axis; i.e., it may be 1-, 2-, or 3-dimensional.

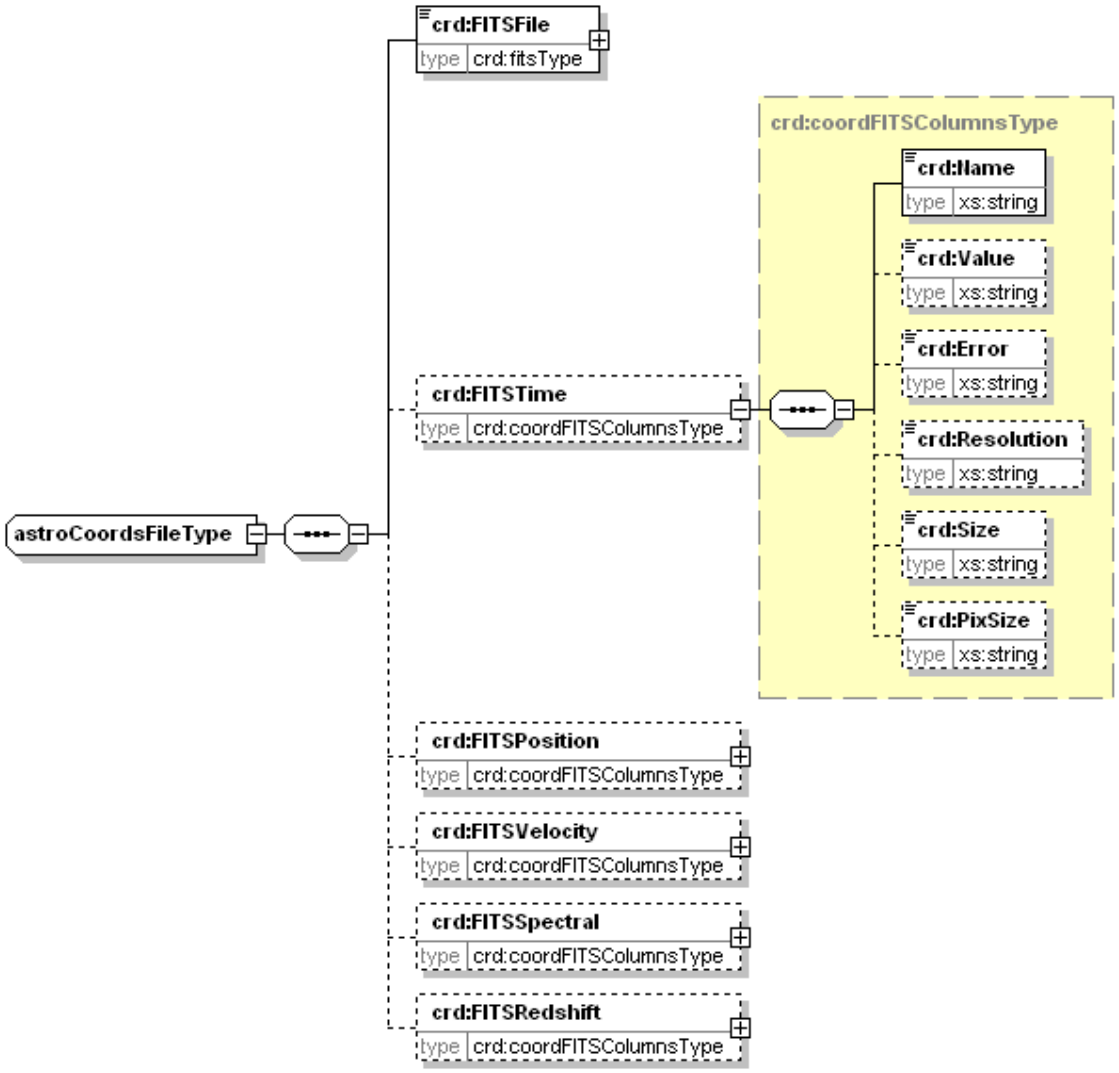


In particular, the multi-dimensional components other than Value are more complicated. Value would be a vector of 3 doubles in Position3D, but the others might include additional position angles or might be formally expressed by a 3x3 matrix. The attributes of a position angle provide its units as well as its definition.



6.7 astroCoordsFileType

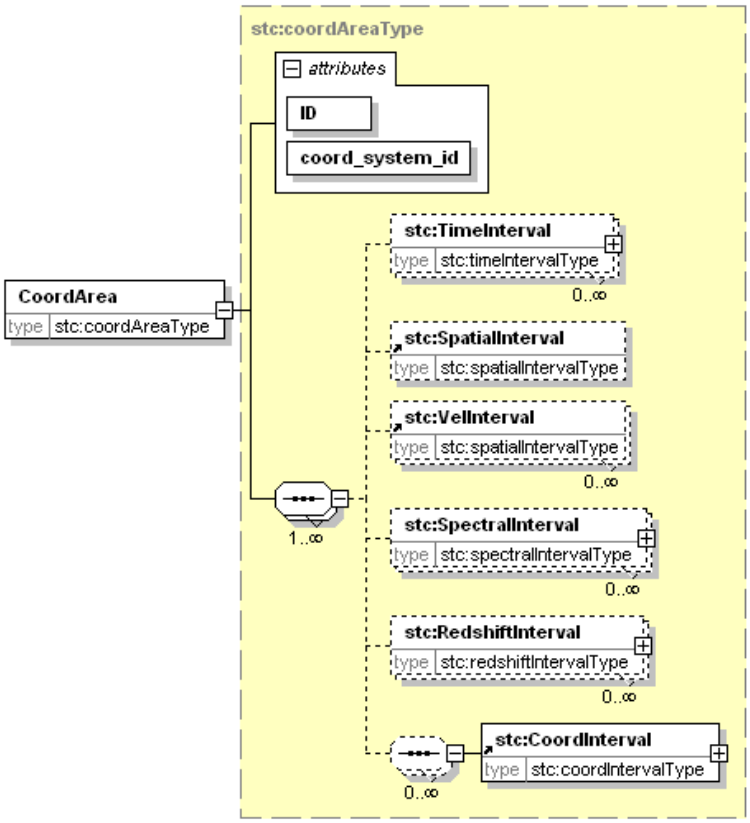
Coordinates in a FITS file:
The AstroCoords element may refer to a specific binary table HDU in a FITS file (specified through FITSFile, derived from anyURI), identifying the columns in which individual coordinate components are contained.



7 Coordinate Area

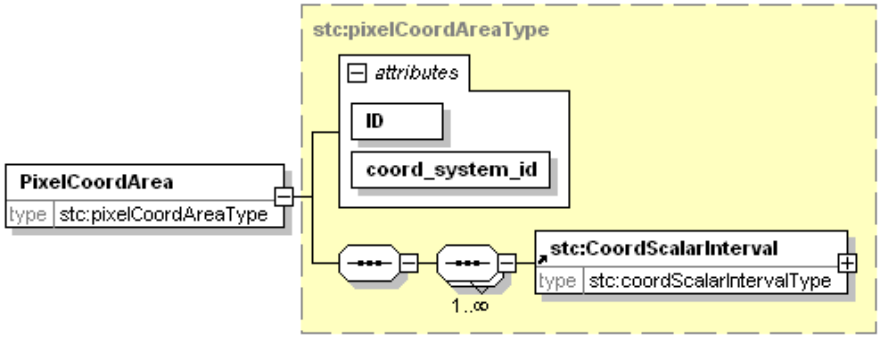
7.1 CoordArea

CoordArea specifies the volume in coordinate space that is occupied by the object to which the STC metadata is attached. It requires an IDREF pointing to a CoordSys. An area may consist of multiple intervals along each axis. Intervals have a FillFactor.



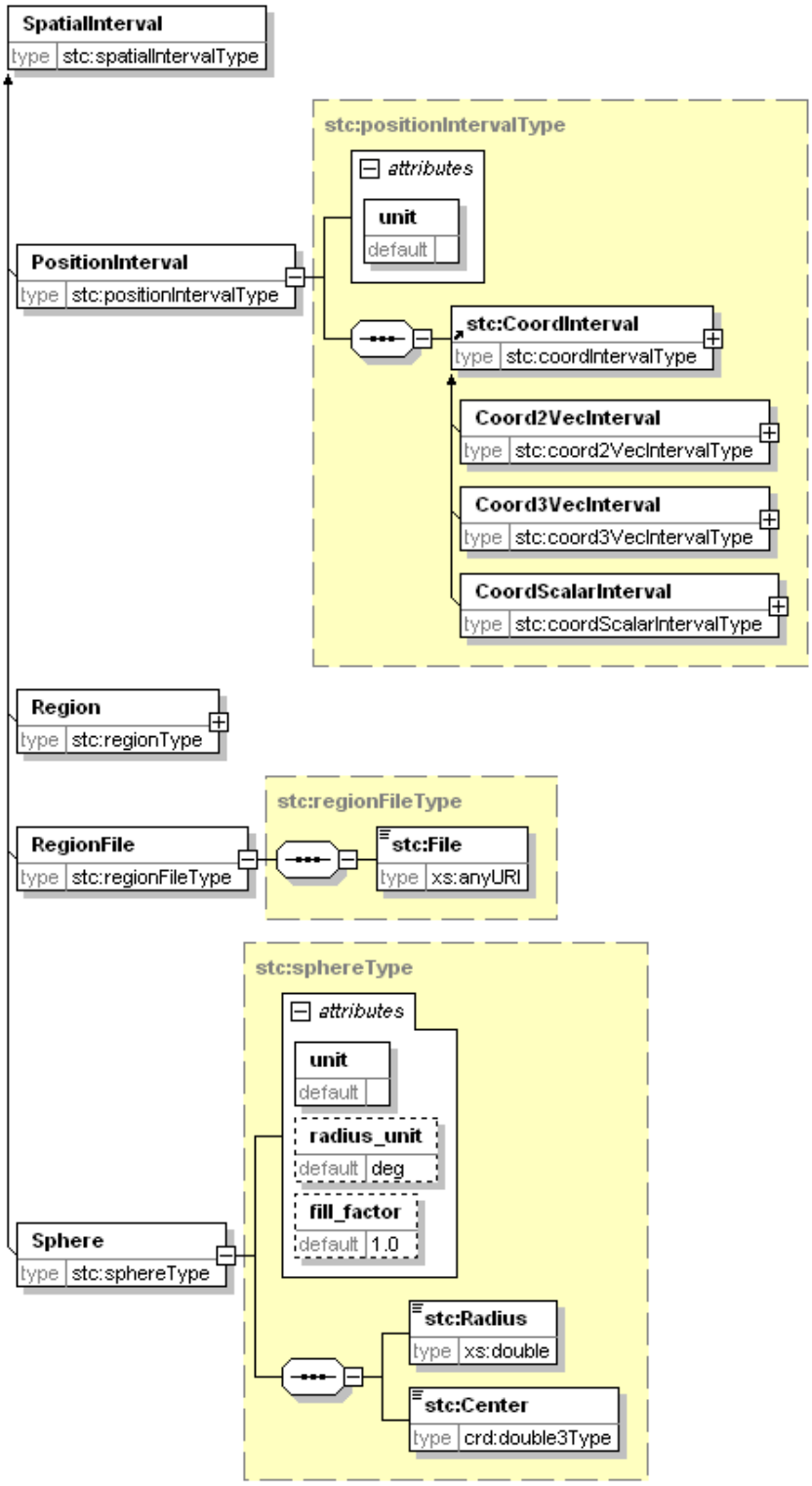
7.2 PixelCoordArea

PixelCoordArea specifies the bounds of a pixilated data object in pixel space.



7.3 SpatialInterval

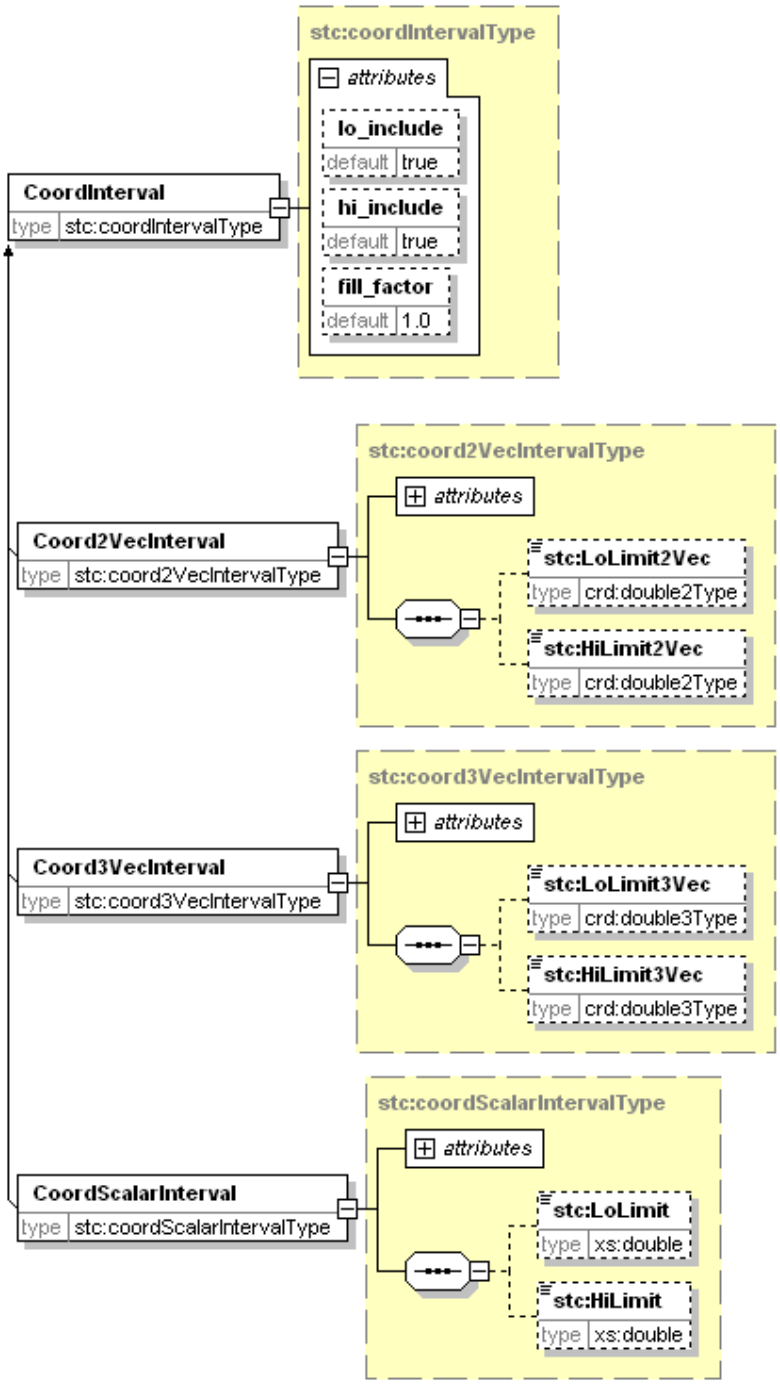
Spatial area is not only more complicated because it may be multi-dimensional by itself, but also because there are additional options: a Region (specified directly or in a file) or a Sphere (or Circle/Cone).



8 Coordinate interval

This is the most fundamental element type to specify an area; it may be 1-, 2-, or 3-dimensional. Bounds may or may not be included. Presence of only one of the bounds indicates a lower, respectively upper, limit.

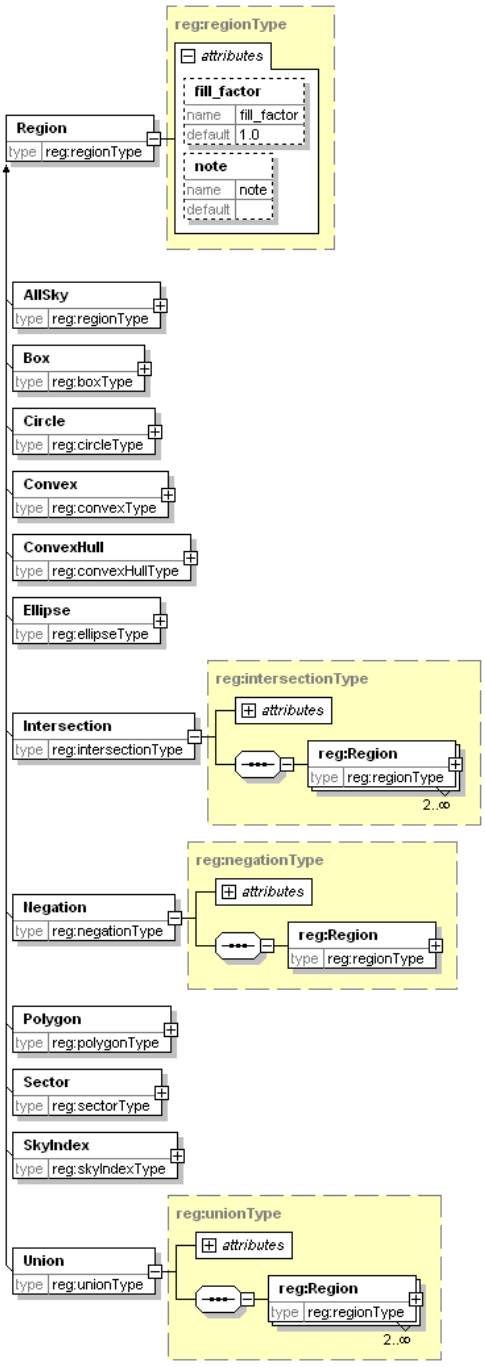
8.1 CoordInterval



9 Spatial Region

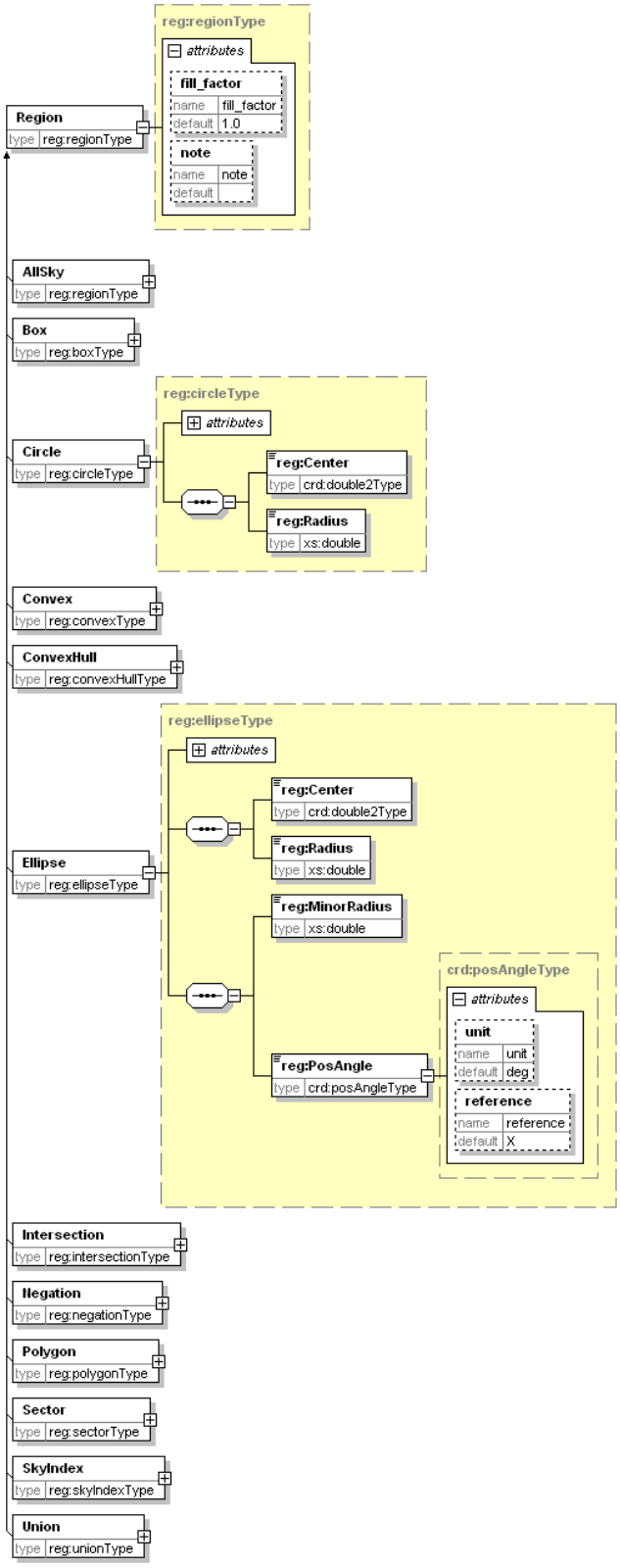
Region is a specialized construct to address the needs for expressing common spatial shapes in a comprehensive way. A Region may consist of a Shape or be the result of an operation performed on one (Negation) or two (Union, Intersection) Regions.

9.1 Negation, Union, Intersection

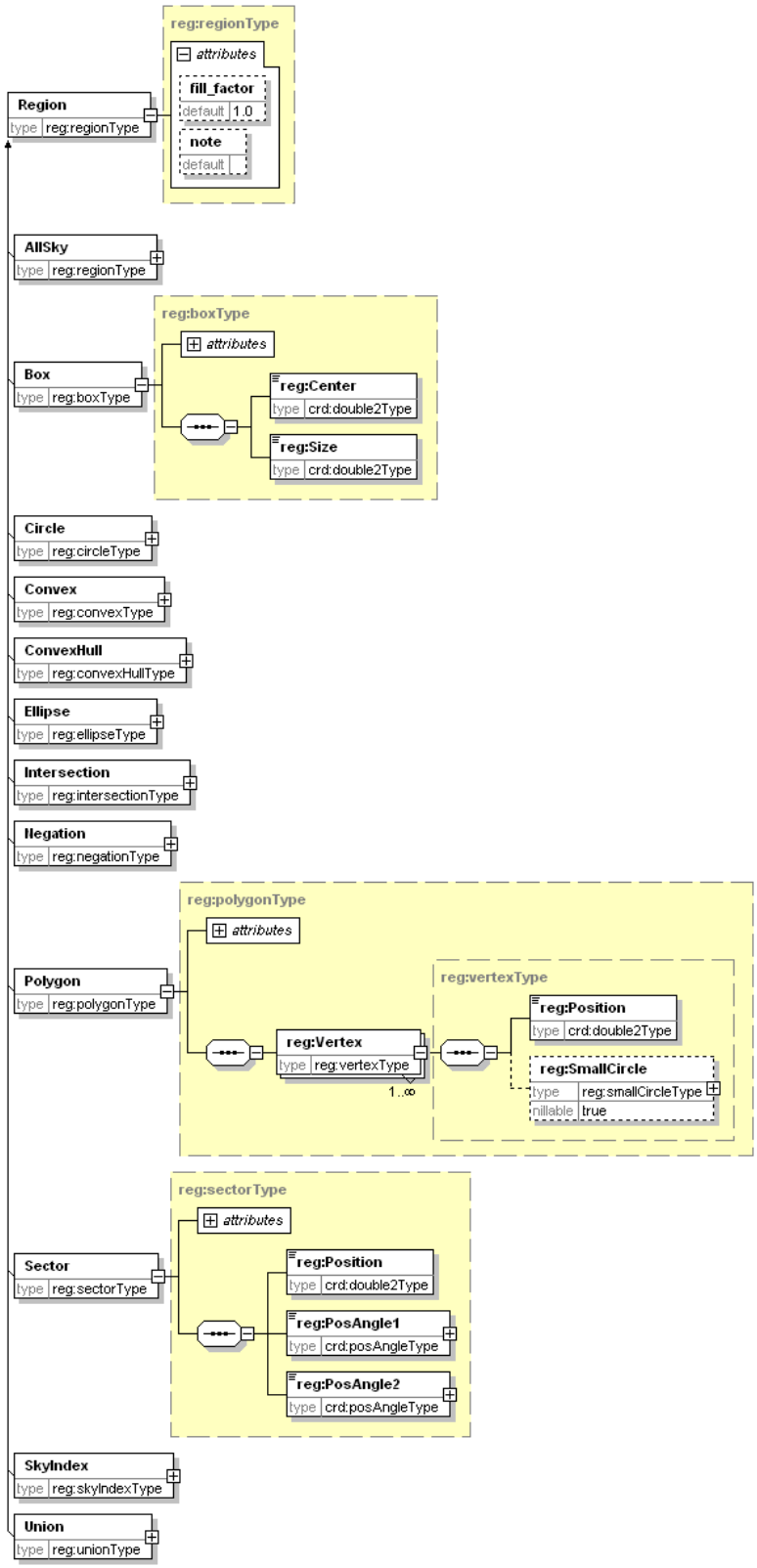


9.2 Allsky, Circle, and Ellipse

AllSky is an empty element (except for FillFactor) that is just a convenience.

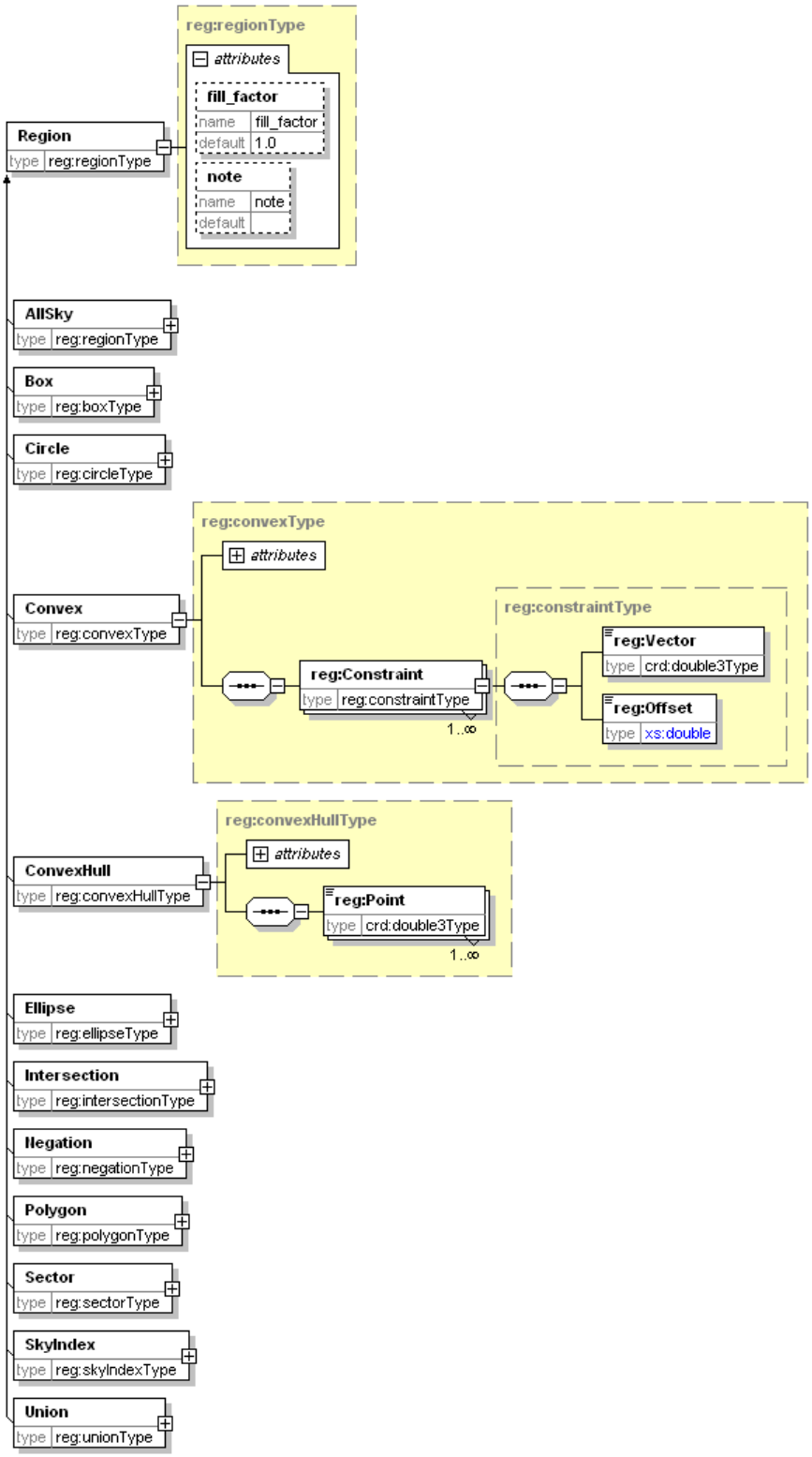


9.3 Polygon, Box, and Sector



9.4 Convex and Convex Hull

Convex and ConvexHull operate on the Unit Sphere.



10 Changes from Previous Versions

References

A. H. Rots, *Space-Time Coordinate Metadata for the Virtual Observatory*, http://www.ivoa.net/Documents/latest/STC.html
A. H. Rots, *STC-S (LinearSTC)*: Space-Time Coordinate (STC) Metadata Linear String Implementation
http://www.ivoa.net/Documents/latest/STC-S.html