



*International  
Virtual  
Observatory  
Alliance*

# **IVOA STC Model: Coordinate Metadata for the Virtual Observatory**

## **I. Coordinate Systems, Frames, and Values**

**Version 2.0**

**IVOA Working Draft 20180414**

**This version:**

WD-STC-2.0-20180414

**Previous version(s):**

**Editor(s):**

Arnold Rots

**Authors:**

Arnold Rots and Mark Cresitello-Dittmar

## Abstract

**STC2** is version 2 of the Data Model for the metadata describing Space-Time, related, and other Coordinates. These metadata are to be used for specifying coordinate-related information for datasets, catalogs, and queries. It consists of five sub-models that have uni-directional dependencies.

1. **Coordinates** provides the metadata for the coordinate frames and coordinate locations. It consists of three packages:
  - 1.1. CoordSystems provides the basic model for constructing coordinate frames and for collecting them in coordinate systems
  - 1.2. Coordinates defines the general model for specifying coordinate values
  - 1.3. Domain sub-packages provide the concrete model design for the five astronomical domains (temporal, spatial, spectral, redshift, and polarization), generic coordinates (any other – usually dependent – coordinate variable), and pixel coordinates
2. **Transforms** models the specification of coordinate frame transformations and are defined as a mapping from one coordinate frame into another
3. **Measurement** extends the Coordinate concept to associate uncertainties and resolution measures with particular coordinate values
4. **CoordinateArea** allows the specification of the volume in coordinate space that a particular dataset occupies
5. **Region** is a specialization of CoordinateArea specifically for two-dimensional spatial coordinates

This document describes the **STC2 Coordinates** model.

## Status of This Document

This is an IVOA Working Draft for review by IVOA members and other interested parties. It is a draft document and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use IVOA Working Drafts as reference materials or to cite them as other than "work in progress".

## Acknowledgements

This document has been developed with support from NSF and NASA under the Virtual Astronomical Observatory (VAO) project, the National Science Foundation's <http://www.nsf.gov> Information Technology Research Program under Cooperative Agreement AST0122449 with The Johns Hopkins University, from the UK Particle Physics and Astronomy Research Council (PPARC) <http://www.pparc.ac.uk>, and from the Euro-VO projects (European Commission 7th program): Euro-VO Aida, VO-ICE and CoSADIE.

## Change Log:

2017-10-12

## Contents

1	Introduction.....	8
1.1	Motivation.....	8
1.2	Requirements.....	8
1.3	Context and Scope.....	8
1.4	Model Representation .....	9
1.5	Role in the IVOA Architecture .....	9
2	Index of elements:.....	10
3	Top-level Packages .....	10
3.1	Coordinate Systems, Frames, Spaces, and Axes .....	10
3.1.1	Class CoordSys.....	11
3.1.2	Class AstroCoordSystem.....	12
3.1.3	Class PixelCoordSystem.....	13
3.1.4	Class CoordFrame.....	13
3.1.5	Class CoordSpace.....	13
3.1.6	Class Axis .....	14
3.1.7	Class CoordAxis.....	14
3.1.8	Class PixelAxis.....	15
3.1.9	Class DiscreteAxis .....	16
3.2	Coordinates .....	16
3.2.1	DataType BasicCoordValue .....	17
3.2.2	DataType CoordValue.....	17
3.2.3	DataType PhysicalCoordValue.....	18
3.2.4	DataType RealCoordValue.....	19
3.2.5	DataType IntegerCoordValue.....	19

- 3.2.6    DataType DiscreteCoordValue ..... 19
- 3.2.7    DataType MultiCoordValue ..... 20
- 4    Coordinate Domains..... 20
- 4.1    Generic Domain..... 20
- 4.1.1    Class GenericFrame ..... 21
- 4.1.2    Class GenericCoordSpace ..... 21
- 4.1.3    DataType GenericCoordValue ..... 22
- 4.2    Temporal Domain..... 23
- 4.2.1    Brief Primer on Time Metadata..... 23
- 4.2.2    Class TimeFrame..... 25
- 4.2.3    Class TimeSpace ..... 27
- 4.2.4    DataType TimeStamp ..... 27
- 4.2.5    DataType TimeOffset..... 28
- 4.2.6    DataType JDTime..... 28
- 4.2.7    DataType JD..... 29
- 4.2.8    DataType MJD ..... 29
- 4.2.9    DataType ISOtime..... 29
- 4.2.10    DataType TimeCoordinateValue ..... 29
- 4.2.11    DataType TimeInstance..... 30
- 4.2.12    DataType Duration ..... 30
- 4.2.13    DataType Phase..... 31
- 4.2.14    DataType TimeLag..... 31
- 4.3    Spatial Domain ..... 31
- 4.3.1    Class SpaceFrame ..... 32
- 4.3.2    Enumeration StdRefFrame ..... 33

4.3.3	DataType Location.....	35
4.3.4	DataType CustomLocation .....	36
4.3.5	DataType StdLocation .....	36
4.3.6	Enumeration StdRefPos.....	37
4.3.7	DataType Epoch.....	38
4.3.8	Class EquatorialFrame .....	38
4.3.9	Class GalacticFrame.....	39
4.3.10	Class EclipticFrame .....	40
4.3.11	Class GeoFrame .....	41
4.3.12	Class SolarFrame .....	42
4.3.13	Class PlanetaryFrame .....	43
4.3.14	Class SpatialSpace .....	44
4.3.15	DataType SpatialCoordValue.....	45
4.3.16	DataType SpatialValue .....	46
4.3.17	DataType SpatialValue1D.....	47
4.3.18	DataType SpatialValue2D.....	47
4.3.19	DataType SpatialValue3D.....	48
4.3.20	DataType SpaceCoord .....	48
4.3.21	DataType Cartesian .....	49
4.3.22	DataType Ecliptic.....	49
4.3.23	DataType Equatorial.....	50
4.3.24	DataType Galactic.....	51
4.3.25	DataType LongLat.....	52
4.3.26	DataType UnitSphere .....	52
4.4	Package Space Instances .....	53

- 4.5 Spectral Domain ..... 54
  - 4.5.1 Class SpectralFrame ..... 54
  - 4.5.2 Class SpectralSpace ..... 55
  - 4.5.3 DataType SpectralValue ..... 55
  - 4.5.4 DataType Energy ..... 56
  - 4.5.5 DataType Frequency..... 56
  - 4.5.6 DataType Wavelength ..... 57
- 4.6 Redshift Domain ..... 57
  - 4.6.1 Class RedshiftFrame ..... 58
  - 4.6.2 Class RedshiftSpace ..... 59
  - 4.6.3 DataType RedshiftValue ..... 59
  - 4.6.4 DataType Redshift ..... 60
  - 4.6.5 DataType DopplerVelocity..... 60
  - 4.6.6 Enumeration DopplerDefinition ..... 61
- 4.7 Polarization Domain ..... 61
  - 4.7.1 Class PolarizationFrame ..... 63
  - 4.7.2 Class PolarizationSpace ..... 63
  - 4.7.3 DataType PolCoordValue ..... 63
  - 4.7.4 Enumeration PolEnum..... 64
  - 4.7.5 DataType PolStokes..... 65
  - 4.7.6 DataType PolCircular ..... 66
  - 4.7.7 DataType PolLinear ..... 67
  - 4.7.8 DataType PolVector..... 67
- 4.8 Pixel Domain..... 68
  - 4.8.1 Class PixelFrame ..... 70

## STC2: I Coordinate Systems, Frames, and Values

4.8.2	Class PixelSpace.....	70
4.8.3	Enumeration Handedness .....	71
4.8.4	DataType PixelIndex .....	71
4.8.5	DataType RealPixelValue.....	72
4.8.6	DataType PixelCoordinate.....	73
4.8.7	DataType PixelCoord1D.....	73
4.8.8	DataType PixelCoord2D.....	74
4.8.9	DataType PixelCoord3D.....	74
5	Imported Models: ivoa Model and IVOA UML Profile .....	75

# 1 Introduction

## 1.1 Motivation

Astronomy, being primarily being a science that crucially depends on observations, has a very basic need for complete, accurate, and unambiguous metadata regarding coordinate information, meaning all coordinates of the observable space and noting that several of these are intertwined. The Data Model described in this document aims to provide a model for such metadata, satisfying the requirements.

## 1.2 Requirements

The primary goal of this document is the specification of a Data Model for coordinate metadata that satisfies the following requirements; the Data Model *shall*:

1. Cover all coordinate axes of observable space: Time, Space, Electro-magnetic Spectrum, Redshift (or Doppler Velocity), Polarization, and have the ability to cover any other incidental coordinates (e.g., temperature)
2. Provide metadata that are complete
3. Provide metadata that are unambiguous
4. Provide metadata that are accurate
5. Be expressed in VO-DML
6. Allow usage of only relevant subsets of the metadata, with the proviso that they must satisfy Requirements 2, 3, and 4
7. Be extensible

## 1.3 Context and Scope

**STC2** is version 2 of the Data Model for the metadata describing Space-Time, related, and other Coordinates. These metadata are to be used for specifying coordinate-related information for datasets, catalogs, and queries. It consists of five sub-models that have uni-directional dependencies.

1. **Coordinates** provides the metadata for the coordinate frames and coordinate locations. It consists of three packages:
  - 1.1. CoordSystems provides the basic model for constructing coordinate frames and for collecting them in coordinate systems
  - 1.2. Coordinates defines the general model for specifying coordinate values
  - 1.3. Domain sub-packages provide the concrete model design for the five astronomical domains (temporal, spatial, spectral, redshift, and polarization), generic coordinates (any other – usually dependent – coordinate variable), and pixel coordinates
2. **Transforms** models the specification of coordinate frame transformations and are defined as a mapping from one coordinate frame into another
3. **Measurement** extends the Coordinate concept to associate uncertainties and resolution measures with particular coordinate values
4. **CoordinateArea** allows the specification of the volume in coordinate space that a particular dataset occupies
5. **Region** is a specialization of CoordinateArea specifically for two-dimensional spatial coordinates



This document describes the **STC2 Coordinates** model.

### 1.4 Model Representation

The model is represented in this document in the Uniform Modeling Language (UML):

- Classes have a **orange header**
- Data types have a **yellow header**
- Generalizations (inheritance) are indicated by **red lines**
- Associations are represented by **green lines**
- Compositions are represented by **blue lines**
- Elements with **red headers** represent parent elements imported from another package
- Elements with **green headers** represent associated elements imported from another package

### 1.5 Role in the IVOA Architecture

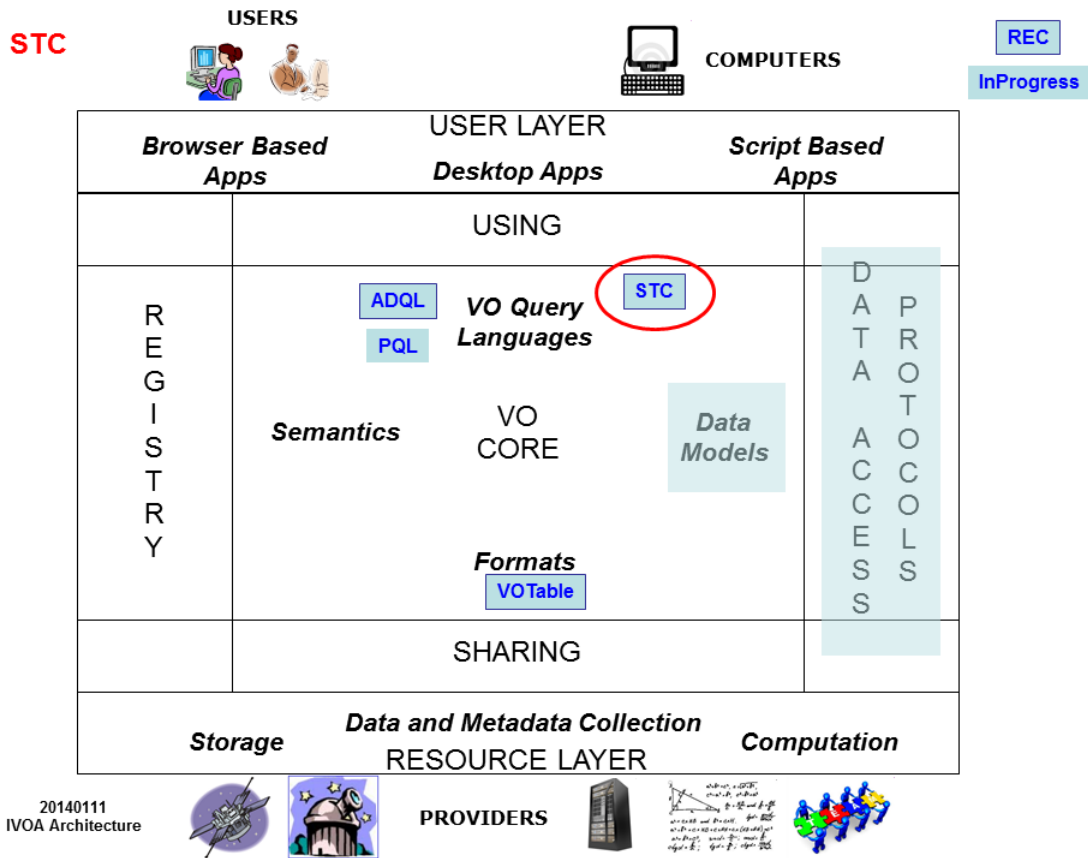


Figure 1 IVOA Standards Context

## 2 Index of elements:

Class	<a href="#">AstroCoordSystem</a> <a href="#">CoordSpace</a> <a href="#">EquatorialFrame</a> <a href="#">GeoFrame</a> <a href="#">PixelSpace</a> <a href="#">RedshiftFrame</a> <a href="#">SpatialSpace</a> <a href="#">TimeSpace</a>	<a href="#">Axis</a> <a href="#">CoordSys</a> <a href="#">GalacticFrame</a> <a href="#">PixelAxis</a> <a href="#">PlanetaryFrame</a> <a href="#">RedshiftSpace</a> <a href="#">SpectralFrame</a>	<a href="#">CoordAxis</a> <a href="#">DiscreteAxis</a> <a href="#">GenericCoordSpace</a> <a href="#">PixelCoordSystem</a> <a href="#">PolarizationFrame</a> <a href="#">SolarFrame</a> <a href="#">SpectralSpace</a>	<a href="#">CoordFrame</a> <a href="#">EclipticFrame</a> <a href="#">GenericFrame</a> <a href="#">PixelFrame</a> <a href="#">PolarizationSpace</a> <a href="#">SpaceFrame</a> <a href="#">TimeFrame</a>
DataType	<a href="#">BasicCoordValue</a> <a href="#">DiscreteCoordValue</a> <a href="#">Energy</a> <a href="#">Galactic</a> <a href="#">ISOtime</a> <a href="#">LongLat</a> <a href="#">PhysicalCoordValue</a> <a href="#">PixelCoordinate</a> <a href="#">PolLinear</a> <a href="#">RealCoordValue</a> <a href="#">RedshiftValue</a> <a href="#">SpatialValue1D</a> <a href="#">StdLocation</a> <a href="#">TimeOffset</a>	<a href="#">Cartesian</a> <a href="#">DopplerVelocity</a> <a href="#">Epoch</a> <a href="#">GenericCoordValue</a> <a href="#">JD</a> <a href="#">MJD</a> <a href="#">PixelCoord1D</a> <a href="#">PixelIndex</a> <a href="#">PolStokes</a> <a href="#">RealPixelValue</a> <a href="#">SpaceCoord</a> <a href="#">SpatialValue2D</a> <a href="#">TimeCoordinateValue</a> <a href="#">TimeStamp</a>	<a href="#">CoordValue</a> <a href="#">Duration</a> <a href="#">Equatorial</a> <a href="#">IntegerCoordValue</a> <a href="#">JDTime</a> <a href="#">MultiCoordValue</a> <a href="#">PixelCoord2D</a> <a href="#">PolCircular</a> <a href="#">PolVector</a> <a href="#">RealQuantity</a> <a href="#">SpatialCoordValue</a> <a href="#">SpatialValue3D</a> <a href="#">TimeInstance</a> <a href="#">UnitSphere</a>	<a href="#">CustomLocation</a> <a href="#">Ecliptic</a> <a href="#">Frequency</a> <a href="#">IntegerQuantity</a> <a href="#">Location</a> <a href="#">Phase</a> <a href="#">PixelCoord3D</a> <a href="#">PolCoordValue</a> <a href="#">Quantity</a> <a href="#">Redshift</a> <a href="#">SpatialValue</a> <a href="#">SpectralValue</a> <a href="#">TimeLag</a> <a href="#">Wavelength</a>
Enumeration	<a href="#">DopplerDefinition</a> <a href="#">GalacticFrames</a> <a href="#">PolCircularEnum</a> <a href="#">PolVectorEnum</a> <a href="#">TimeScale</a>	<a href="#">EclipticFrames</a> <a href="#">GeoRefFrames</a> <a href="#">PolEnum</a> <a href="#">SolarRefFrames</a>	<a href="#">EpochType</a> <a href="#">Handedness</a> <a href="#">PolLinearEnum</a> <a href="#">StdRefFrame</a>	<a href="#">EquatorialRefFrames</a> <a href="#">PlanetRefFrames</a> <a href="#">PolStokesEnum</a> <a href="#">StdRefPos</a>
Package	<a href="#">coords</a> <a href="#">pixel</a> <a href="#">SpaceInstances</a>	<a href="#">coordsystem</a> <a href="#">polarization</a> <a href="#">spectral</a>	<a href="#">domain</a> <a href="#">redshift</a> <a href="#">time</a>	<a href="#">generic</a> <a href="#">space</a>

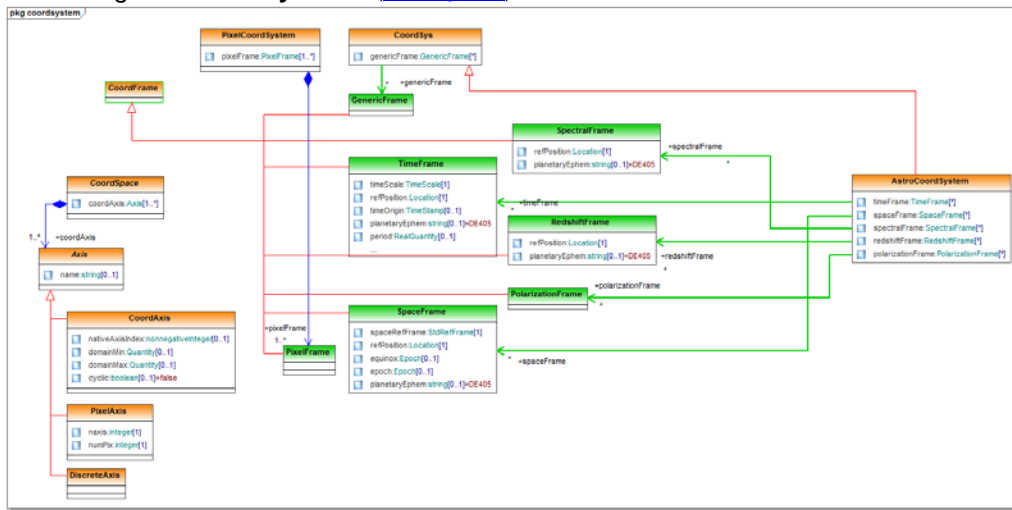
## 3 Top-level Packages

### 3.1 Coordinate Systems, Frames, Spaces, and Axes

#### Package **coordsystem**

owner	<a href="#">stc2</a> <a href="#">coordinates</a>
properties	qualified name <code>stc2_coordinates::coordsystem</code>
ownedDiagrams	<a href="#">CoordSystems</a>
ownedMember	<a href="#">AstroCoordSystem</a> <a href="#">Axis</a> <a href="#">CoordAxis</a> <a href="#">CoordFrame</a> <a href="#">CoordSpace</a> <a href="#">CoordSys</a> <a href="#">DiscreteAxis</a> <a href="#">PixelAxis</a> <a href="#">PixelCoordSystem</a>
source of relation	Dependency <a href="#">IVOA_UML_Profile</a> <a href="#">ivoa</a>

Class Diagram **CoordSystems** ([coordsystem](#))



3.1.1 Class **CoordSys**

diagram	
owner	<a href="#">coordsystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::CoordSys</code> abstract <code>false</code>
ownedMember	<a href="#">genericFrame</a>
specific	<a href="#">AstroCoordSystem</a>
shown on diagram	<a href="#">CoordSystems</a>
documentation	A Coordinate System is a collection of Coordinate Frames. This class is the base-level coordinate system and MAY contain any number of generic coordinate frames.

Property **CoordSys::genericFrame**

owner	<a href="#">CoordSys</a>
properties	qualified name <code>stc2_coordinates::coordsystem::CoordSys::genericFrame</code> multiplicity <code>*</code> type <a href="#">GenericFrame</a>

### 3.1.2 Class AstroCoordSystem

diagram	
owner	<a href="#">coordsystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::AstroCoordSystem</code> abstract <code>false</code>
ownedMember	<a href="#">polarizationFrame</a> <a href="#">redshiftFrame</a> <a href="#">spaceFrame</a> <a href="#">spectralFrame</a> <a href="#">timeFrame</a>
general	<a href="#">CoordSys</a>
shown on diagram	<a href="#">CoordSystems</a>
documentation	An AstroCoordSystem MAY contain any number of coordinate frames: GenericFrame, TimeFrame, SpaceFrame, SpectralFrame, RedshiftFrame, PolarizationFrame. In addition, it SHALL contain the common reference position (refPosition) for the astronomical frames and it MAY contain the name of a planetary ephemeris (which SHOULD be included when appropriate).

#### Property AstroCoordSystem::polarizationFrame

owner	<a href="#">AstroCoordSystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::AstroCoordSystem::polarizationFrame</code> multiplicity <code>*</code> type <a href="#">PolarizationFrame</a>

#### Property AstroCoordSystem::redshiftFrame

owner	<a href="#">AstroCoordSystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::AstroCoordSystem::redshiftFrame</code> multiplicity <code>*</code> type <a href="#">RedshiftFrame</a>

#### Property AstroCoordSystem::spaceFrame

owner	<a href="#">AstroCoordSystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::AstroCoordSystem::spaceFrame</code> multiplicity <code>*</code> type <a href="#">SpaceFrame</a>

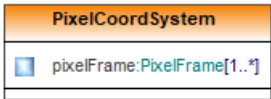
#### Property AstroCoordSystem::spectralFrame

owner	<a href="#">AstroCoordSystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::AstroCoordSystem::spectralFrame</code> multiplicity <code>*</code> type <a href="#">SpectralFrame</a>

#### Property AstroCoordSystem::timeFrame

owner	<a href="#">AstroCoordSystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::AstroCoordSystem::timeFrame</code> multiplicity <code>*</code> type <a href="#">TimeFrame</a>

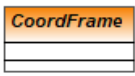
### 3.1.3 Class PixelCoordSystem

diagram	
owner	<a href="#">coordsystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::PixelCoordSystem</code> abstract <code>false</code>
ownedMember	<a href="#">pixelFrame</a>
shown on diagram	<a href="#">CoordSystems</a> <a href="#">PixelDomain</a>
documentation	A Pixel Coordinate System contains one or more Pixel Frames.


#### Property PixelCoordSystem::pixelFrame

owner	<a href="#">PixelCoordSystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::PixelCoordSystem::pixelFrame</code> multiplicity <code>1..*</code> type <a href="#">PixelFrame</a> aggregation <code>composite</code>

### 3.1.4 Class CoordFrame

diagram	
owner	<a href="#">coordsystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::CoordFrame</code> abstract <code>true</code>
specific	<a href="#">GenericFrame</a> <a href="#">PixelFrame</a> <a href="#">PolarizationFrame</a> <a href="#">RedshiftFrame</a> <a href="#">SpaceFrame</a> <a href="#">SpectralFrame</a> <a href="#">TimeFrame</a>
typedElements	DataType <a href="#">BasicCoordValue</a> Property <a href="#">coordFrame</a>
shown on diagram	<a href="#">Coords</a> <a href="#">CoordSystems</a> <a href="#">GenericDomain</a> <a href="#">PixelDomain</a> <a href="#">PolarizationDomain</a> <a href="#">RedshiftDomain</a> <a href="#">SpatialDomain</a> <a href="#">SpectralDomain</a> <a href="#">TimeDomain</a>
documentation	This is the abstract empty base class for all coordinate frames. A Frame SHALL contain one corresponding Coordinate Space.

### 3.1.5 Class CoordSpace

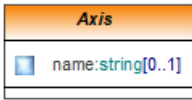
diagram	
owner	<a href="#">coordsystem</a>
properties	qualified name <code>stc2_coordinates::coordsystem::CoordSpace</code> abstract <code>true</code>
ownedMember	<a href="#">coordAxis</a>
specific	<a href="#">GenericCoordSpace</a> <a href="#">PixelSpace</a> <a href="#">PolarizationSpace</a> <a href="#">RedshiftSpace</a> <a href="#">SpatialSpace</a> <a href="#">SpectralSpace</a> <a href="#">TimeSpace</a>
shown on diagram	<a href="#">CoordSystems</a> <a href="#">GenericDomain</a> <a href="#">PixelDomain</a> <a href="#">PolarizationDomain</a> <a href="#">RedshiftDomain</a> <a href="#">SpatialDomain</a> <a href="#">SpectralDomain</a> <a href="#">TimeDomain</a>

documentation	The CoordSpace is the abstract class from which all coordinate spaces are derived. It contains up to three coordinate axes.
---------------	---

Property **CoordSpace::coordAxis**

owner	<a href="#">CoordSpace</a>
properties	qualified name stc2_coordinates::coordsystem::CoordSpace::coordAxis multiplicity 1..* type <a href="#">Axis</a> aggregation Composite

### 3.1.6 Class Axis

diagram	
owner	<a href="#">coordsystem</a>
properties	qualified name stc2_coordinates::coordsystem::Axis abstract true
ownedMember	<a href="#">name</a>
specific	<a href="#">CoordAxis</a> <a href="#">DiscreteAxis</a> <a href="#">PixelAxis</a>
typedElements	Class <a href="#">CoordSpace</a> Property <a href="#">coordAxis</a> DataType <a href="#">CoordValue</a> Property <a href="#">coordAxis</a>
shown on diagram	<a href="#">Coords</a> <a href="#">CoordSystems</a>
documentation	Axis is the abstract parent class for all coordinate axes. It SHALL contain the name of the axis.

Property **Axis::name**

owner	<a href="#">Axis</a>
properties	qualified name stc2_coordinates::coordsystem::Axis::name multiplicity 0..1 type string
documentation	Label for the first axis.

### 3.1.7 Class CoordAxis

diagram	
owner	<a href="#">coordsystem</a>
properties	qualified name stc2_coordinates::coordsystem::CoordAxis abstract false
ownedMember	<a href="#">cyclic</a> <a href="#">domainMax</a> <a href="#">domainMin</a> <a href="#">nativeAxisIndex</a>
general	<a href="#">Axis</a>
typedElements	Class <a href="#">GenericCoordSpace</a> Property <a href="#">coordAxis</a> DataType <a href="#">PhysicalCoordValue</a> Property <a href="#">coordAxis</a> Class <a href="#">RedshiftSpace</a> Property <a href="#">coordAxis</a>

	Class <a href="#">SpatialSpace</a> Property <a href="#">coordAxis</a> Class <a href="#">SpectralSpace</a> Property <a href="#">coordAxis</a> Class <a href="#">TimeSpace</a> Property <a href="#">coordAxis</a> DataType <a href="#">TimeStamp</a> Property <a href="#">coordAxis</a>
shown on diagram	<a href="#">Coords</a> <a href="#">CoordSystems</a> <a href="#">GenericDomain</a> <a href="#">RedshiftDomain</a> <a href="#">SpatialDomain</a> <a href="#">SpectralDomain</a> <a href="#">TimeDomain</a>
documentation	CoordAxis is the most common type of coordinate axis, derived from Axis. It MAY contain a nativeAxisIndex if its coordinate space consists of more than one axis; the minimum and maximum allowable coordinate value along the axis in its domain (only if meaningful, of course); and a flag indicating whether the axis is cyclic (default: false).

Property **CoordAxis::cyclic**

owner	<a href="#">CoordAxis</a>
properties	qualified name stc2_coordinates::coordsystem::CoordAxis::cyclic multiplicity 0..1 type boolean

Property **CoordAxis::domainMax**

owner	<a href="#">CoordAxis</a>
properties	qualified name stc2_coordinates::coordsystem::CoordAxis::domainMax multiplicity 0..1 type <a href="#">Quantity</a>

Property **CoordAxis::domainMin**

owner	<a href="#">CoordAxis</a>
properties	qualified name stc2_coordinates::coordsystem::CoordAxis::domainMin multiplicity 0..1 type <a href="#">Quantity</a>

Property **CoordAxis::nativeAxisIndex**

owner	<a href="#">CoordAxis</a>
properties	qualified name stc2_coordinates::coordsystem::CoordAxis::nativeAxisIndex multiplicity 0..1 type nonnegativeInteger

### 3.1.8 Class PixelAxis

diagram	<pre> classDiagram     class PixelAxis {         naxis: integer[1]         numPix: integer[1]     }                 </pre>
owner	<a href="#">coordsystem</a>
properties	qualified name stc2_coordinates::coordsystem::PixelAxis abstract false
ownedMember	<a href="#">naxis</a> <a href="#">numPix</a>
general	<a href="#">Axis</a>
typedElements	DataType <a href="#">PixelIndex</a> Property <a href="#">coordAxis</a> Class <a href="#">PixelSpace</a> Property <a href="#">coordAxis</a> DataType <a href="#">RealPixelValue</a> Property <a href="#">coordAxis</a>
shown on diagram	<a href="#">CoordSystems</a> <a href="#">PixelDomain</a>
documentation	PixelAxis is the derived Axis class for pixel spaces. It SHALL contain its axis index in the PixelSpace and the number of pixels along the axis.

Property **PixelAxis::naxis**

owner	<a href="#">PixelAxis</a>						
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::coordsystem::PixelAxis::naxis</td> </tr> <tr> <td>multiplicity</td> <td>1</td> </tr> <tr> <td>type</td> <td>integer</td> </tr> </table>	qualified name	stc2_coordinates::coordsystem::PixelAxis::naxis	multiplicity	1	type	integer
qualified name	stc2_coordinates::coordsystem::PixelAxis::naxis						
multiplicity	1						
type	integer						

Property **PixelAxis::numPix**

owner	<a href="#">PixelAxis</a>						
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::coordsystem::PixelAxis::numPix</td> </tr> <tr> <td>multiplicity</td> <td>1</td> </tr> <tr> <td>type</td> <td>integer</td> </tr> </table>	qualified name	stc2_coordinates::coordsystem::PixelAxis::numPix	multiplicity	1	type	integer
qualified name	stc2_coordinates::coordsystem::PixelAxis::numPix						
multiplicity	1						
type	integer						

### 3.1.9 Class DiscreteAxis

diagram									
owner	<a href="#">coordsystem</a>								
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::coordsystem::DiscreteAxis</td> </tr> <tr> <td>abstract</td> <td>false</td> </tr> </table>	qualified name	stc2_coordinates::coordsystem::DiscreteAxis	abstract	false				
qualified name	stc2_coordinates::coordsystem::DiscreteAxis								
abstract	false								
general	<a href="#">Axis</a>								
typedElements	<table> <tr> <td>DataType</td> <td><a href="#">DiscreteCoordValue</a></td> <td>Property</td> <td><a href="#">coordAxis</a></td> </tr> <tr> <td>Class</td> <td><a href="#">PolarizationSpace</a></td> <td>Property</td> <td><a href="#">coordAxis</a></td> </tr> </table>	DataType	<a href="#">DiscreteCoordValue</a>	Property	<a href="#">coordAxis</a>	Class	<a href="#">PolarizationSpace</a>	Property	<a href="#">coordAxis</a>
DataType	<a href="#">DiscreteCoordValue</a>	Property	<a href="#">coordAxis</a>						
Class	<a href="#">PolarizationSpace</a>	Property	<a href="#">coordAxis</a>						
shown on diagram	<a href="#">Coords</a> <a href="#">CoordSystems</a> <a href="#">PolarizationDomain</a>								
documentation	DiscreteAxis is specifically intended for enumerated coordinates, such as polarization.								

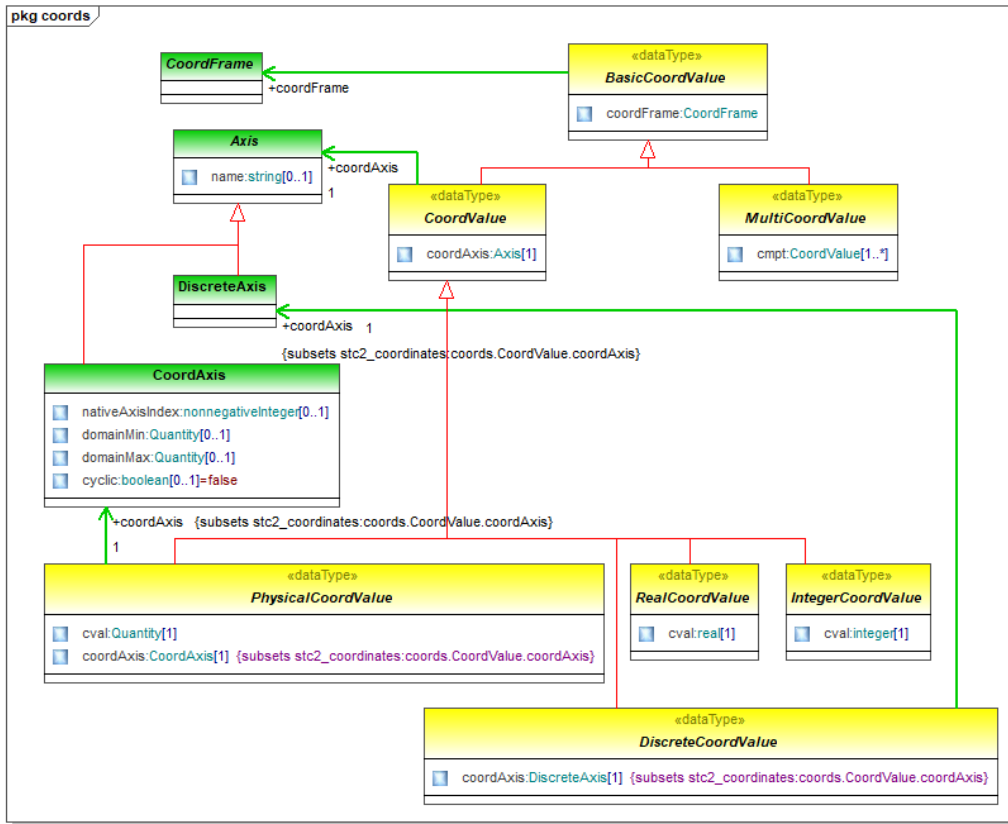
## 3.2 Coordinates

Package **coords**

owner	<a href="#">stc2_coordinates</a>		
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::coords</td> </tr> </table>	qualified name	stc2_coordinates::coords
qualified name	stc2_coordinates::coords		
ownedDiagrams	<a href="#">Coords</a>		
ownedMember	<a href="#">BasicCoordValue</a> <a href="#">CoordValue</a> <a href="#">DiscreteCoordValue</a> <a href="#">IntegerCoordValue</a> <a href="#">MultiCoordValue</a> <a href="#">PhysicalCoordValue</a> <a href="#">RealCoordValue</a>		



Class Diagram **Coords** ([coords](#))



### 3.2.1 DataType BasicCoordValue

owner	<a href="#">coords</a>
properties	qualified name <code>stc2_coordinates::coords::BasicCoordValue</code> abstract <code>true</code>
ownedMember	<a href="#">CoordFrame</a>
specific	<a href="#">CoordValue</a> <a href="#">MultiCoordValue</a> <a href="#">SpaceCoord</a> <a href="#">TimeStamp</a>
shown on diagram	<a href="#">Coords</a> <a href="#">SpatialDomain</a> <a href="#">TimeDomain</a>
documentation	Abstract top-level coordinate value data type. Derived from this are various 1-dimensional coordinate value types and one multi-dimensional type. It refers to a <a href="#">CoordFrame</a> .

#### Property **BasicCoordValue::coordFrame**

owner	<a href="#">BasicCoordValue</a>
properties	qualified name <code>stc2_coordinates::coords::BasicCoordValue::coordFrame</code> type <a href="#">CoordFrame</a>

### 3.2.2 DataType CoordValue

owner	<a href="#">coords</a>
properties	qualified name <code>stc2_coordinates::coords::CoordValue</code> abstract <code>true</code>

ownedMember	<a href="#">coordAxis</a>
general	<a href="#">BasicCoordValue</a>
specific	<a href="#">DiscreteCoordValue</a> <a href="#">IntegerCoordValue</a> <a href="#">PhysicalCoordValue</a> <a href="#">RealCoordValue</a>
typedElements	DataType <a href="#">MultiCoordValue</a> Property <a href="#">cmpt</a>
shown on diagram	<a href="#">Coords</a> <a href="#">SpatialDomain</a>
documentation	Abstract top level 1-dimensional coordinate value data type. There are four sub-types: simple integer and real for pixel spaces; discrete for enumerated coordinates; and the most commonly used physical coordinate value. It refers to a CoordFrame and to a CoordSpace object using coordAxis through an Axis.

Property **CoordValue::coordAxis**

owner	<a href="#">CoordValue</a>
properties	qualified name <code>stc2_coordinates::coords::CoordValue::coordAxis</code> multiplicity 1 type <a href="#">Axis</a>

### 3.2.3 DataType PhysicalCoordValue

owner	<a href="#">coords</a>
properties	qualified name <code>stc2_coordinates::coords::PhysicalCoordValue</code> abstract true
ownedMember	<a href="#">coordAxis</a> <a href="#">cval</a> <a href="#">Subset</a>
general	<a href="#">CoordValue</a>
specific	<a href="#">GenericCoordValue</a> <a href="#">RedshiftValue</a> <a href="#">SpatialCoordValue</a> <a href="#">SpectralValue</a> <a href="#">TimeCoordinateValue</a>
shown on diagram	<a href="#">Coords</a> <a href="#">GenericDomain</a> <a href="#">RedshiftDomain</a> <a href="#">SpatialDomain</a> <a href="#">SpectralDomain</a> <a href="#">TimeDomain</a>
documentation	Most common coordinate value type. The value is an (abstract) Quantity that needs to be further specified in each domain. The coordAxis is subsetted to an object of type CoordAxis.

Constraint **PhysicalCoordValue::Subset**

owner	<a href="#">PhysicalCoordValue</a>
properties	qualified name <code>stc2_coordinates::coords::PhysicalCoordValue::Subset</code> specification <code>subsets stc2_coordinates::coords.CoordValue.coordAxis</code> constrained elements <a href="#">coordAxis</a>

Property **PhysicalCoordValue::coordAxis**

owner	<a href="#">PhysicalCoordValue</a>
properties	qualified name <code>stc2_coordinates::coords::PhysicalCoordValue::coordAxis</code> multiplicity 1 type <a href="#">CoordAxis</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates::coords.CoordValue.coordAxis</code>

Property **PhysicalCoordValue::cval**

owner	<a href="#">PhysicalCoordValue</a>
properties	qualified name <code>stc2_coordinates::coords::PhysicalCoordValue::cval</code> multiplicity 1 type <a href="#">Quantity</a>

### 3.2.4 DataType RealCoordValue

owner	<a href="#">coords</a>
properties	qualified name <code>stc2_coordinates::coords::RealCoordValue</code> abstract <code>true</code>
ownedMember	<a href="#">cval</a>
general	<a href="#">CoordValue</a>
specific	<a href="#">RealPixelValue</a>
shown on diagram	<a href="#">Coords</a> <a href="#">PixelDomain</a>
documentation	Real-valued coordinate specifically intended for use in pixel spaces.

#### Property RealCoordValue::cval

owner	<a href="#">RealCoordValue</a>
properties	qualified name <code>stc2_coordinates::coords::RealCoordValue::cval</code> multiplicity <code>1</code> type <code>real</code>

### 3.2.5 DataType IntegerCoordValue

owner	<a href="#">coords</a>
properties	qualified name <code>stc2_coordinates::coords::IntegerCoordValue</code> abstract <code>true</code>
ownedMember	<a href="#">cval</a>
general	<a href="#">CoordValue</a>
specific	<a href="#">PixelIndex</a>
shown on diagram	<a href="#">Coords</a> <a href="#">PixelDomain</a>
documentation	Integer-valued coordinate specifically intended for pixel index values.

#### Property IntegerCoordValue::cval

owner	<a href="#">IntegerCoordValue</a>
properties	qualified name <code>stc2_coordinates::coords::IntegerCoordValue::cval</code> multiplicity <code>1</code> type <code>integer</code>

### 3.2.6 DataType DiscreteCoordValue

owner	<a href="#">coords</a>
properties	qualified name <code>stc2_coordinates::coords::DiscreteCoordValue</code> abstract <code>true</code>
ownedMember	<a href="#">coordAxis</a> <a href="#">Subset</a>
general	<a href="#">CoordValue</a>
specific	<a href="#">PolCoordValue</a>
shown on diagram	<a href="#">Coords</a> <a href="#">PolarizationDomain</a>
documentation	Coordinate value for discrete (enumerated) coordinate axes. The actual type of the value needs to be specified in the

	coordinate domain. The coordAxis is subsetted to DiscreteAxis. For an example, see the polarization domain.
--	---

**Constraint DiscreteCoordValue::Subset**

owner	<a href="#">DiscreteCoordValue</a>
properties	qualified name stc2_coordinates::coords::DiscreteCoordValue::Subset specification subsets stc2_coordinates::coords::CoordValue::coordAxis constrained elements <a href="#">coordAxis</a>

**Property DiscreteCoordValue::coordAxis**

owner	<a href="#">DiscreteCoordValue</a>
properties	qualified name stc2_coordinates::coords::DiscreteCoordValue::coordAxis multiplicity 1 type <a href="#">DiscreteAxis</a>
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coords::CoordValue::coordAxis

### 3.2.7 DataType MultiCoordValue

owner	<a href="#">coords</a>
properties	qualified name stc2_coordinates::coords::MultiCoordValue abstract true
ownedMember	<a href="#">cmpt</a>
general	<a href="#">BasicCoordValue</a>
specific	<a href="#">PixelCoordinate</a> <a href="#">SpatialValue</a>
shown on diagram	<a href="#">Coords</a> <a href="#">PixelDomain</a> <a href="#">SpatialDomain</a>
documentation	The value for vector coordinates. The vector is built up from scalar CoordValue components and refers to a CoordFrame.

**Property MultiCoordValue::cmpt**

owner	<a href="#">MultiCoordValue</a>
properties	qualified name stc2_coordinates::coords::MultiCoordValue::cmpt multiplicity 1..* type <a href="#">CoordValue</a>

## 4 Coordinate Domains

**Package domain**

owner	<a href="#">stc2_coordinates</a>
properties	qualified name stc2_coordinates::domain
ownedMember	<a href="#">generic</a> <a href="#">pixel</a> <a href="#">polarization</a> <a href="#">redshift</a> <a href="#">space</a> <a href="#">spectral</a> <a href="#">time</a>

### 4.1 Generic Domain

**Package generic**

owner	<a href="#">domain</a>
properties	qualified name stc2_coordinates::domain::generic



ownedMember	<a href="#">coordAxis Subset</a>
general	<a href="#">CoordSpace</a>
shown on diagram	<a href="#">GenericDomain</a>
documentation	The GenericCoordSpace SHALL include one CoordAxis object.

**Constraint GenericCoordSpace::Subset**

owner	<a href="#">GenericCoordSpace</a>
properties	qualified name <code>stc2_coordinates::domain::generic::GenericCoordSpace::Subset</code> specification <code>subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis</code> constrained elements <a href="#">coordAxis</a>

**Property GenericCoordSpace::coordAxis**

owner	<a href="#">GenericCoordSpace</a>
properties	qualified name <code>stc2_coordinates::domain::generic::GenericCoordSpace::coordAxis</code> multiplicity <code>1</code> type <a href="#">CoordAxis</a> aggregation <code>composite</code>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis</code>

### 4.1.3 DataType GenericCoordValue

owner	<a href="#">generic</a>
properties	qualified name <code>stc2_coordinates::domain::generic::GenericCoordValue</code> abstract <code>false</code>
ownedMember	<a href="#">coordFrame cval</a> <a href="#">Subset1</a> <a href="#">Subset2</a>
general	<a href="#">PhysicalCoordValue</a>
shown on diagram	<a href="#">GenericDomain</a>
documentation	This data type contains the generic coordinate value and a reference to a GenericFrame and to a Coordinate Space object through a coordinate axis.

**Constraint GenericCoordValue::Subset1**

owner	<a href="#">GenericCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::generic::GenericCoordValue::Subset1</code> specification <code>subsets stc2_coordinates:coords.PhysicalCoordValue.cval</code> constrained elements <a href="#">cval</a>

**Constraint GenericCoordValue::Subset2**

owner	<a href="#">GenericCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::generic::GenericCoordValue::Subset2</code> specification <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code> constrained elements <a href="#">coordFrame</a>

**Property GenericCoordValue::coordFrame**

owner	<a href="#">GenericCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::generic::GenericCoordValue::coordFrame</code> type <a href="#">GenericFrame</a>
constraints	<a href="#">Subset2</a> : <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code>

Property **GenericCoordValue::cval**

owner	<a href="#">GenericCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::generic::GenericCoordValue::cval</code> multiplicity 1 type <a href="#">RealQuantity</a>
constraints	<a href="#">Subset1</a> : subsets <code>stc2_coordinates::coords.PhysicalCoordValue.cval</code>

## 4.2 Temporal Domain

### 4.2.1 Brief Primer on Time Metadata

For reference and more information, see: FITS WCS Paper IV (Rots et al., 2015 A&A 574, 36).

#### 1. Required:

- 1.1 Record time stamps in JD, MJD, ISO-8601, or elapsed time (if in elapsed time, you need to give a zero point in a time stamp that is not provided in elapsed time, of course)
- 1.2 Provide the time scale used (TT, TDB, TAI, GPS, ET, UTC, TCG, TCB)
- 1.3 Provide the reference position (place where the time is measured)

#### 2. Note the following:

JD and MJD do not imply a time scale; it needs to be provided separately.

JD and MJD are dimensionless, though a unit of "day" is implied.

It's a bad idea to mix UTC with JD or MJD, since not all UTC days are the same length.

Use instead the restricted form of ISO-8601: `[[+|-]c]ccyy-mm-dd[Thh[:mm[:ss[:ss...]]]]`

No time zone characters.

TDB runs on average synchronously with TT, but corrects for the relativistic effects caused by deviations in the earth's orbit from perfect circularity and constant gravitational potential.

#### 3. Recommendations:

Also provide an estimate of the uncertainty in your timestamps.

Avoid UTC. It is trivial to convert the times provided by, e.g., space agencies to TT immediately when you get them and it will save headaches later on.

Use TT: it's the official IAU time scale, continuous with ET and the one solar system ephemerides are based on.

TAI and GPS are acceptable alternatives, with constant offsets from TT.

Use the same reference position for time and space and make sure it is commensurate with your time scale. For instance, when you convert to the barycenter, also convert to TDB.

Beware that the barycenter is not the heliocenter.

Be specific in labeling the time axis; e.g.: `JD(TT;GEOCENTER)` or `MJD(TDB;BARYCENTER)`.

Use proleptic Gregorian dates for ISO-8601.

#### 4. Do never use:

TJD, HJD, BJD, etc.

These are not officially recognized and suggest certain metadata values, but leave considerable ambiguity as to what those metadata values actually are. Instead, specify your metadata explicitly. It avoids confusion later on and isn't much more work.

#### 5. What if you deal with incomplete data?

If you don't know the time scale and/or reference position, you can provide them as UNKNOWN and set the systematic error/uncertainty to, say, 1000 s. 100 s will do if only the time scale is unknown.

6. What else is there to know?

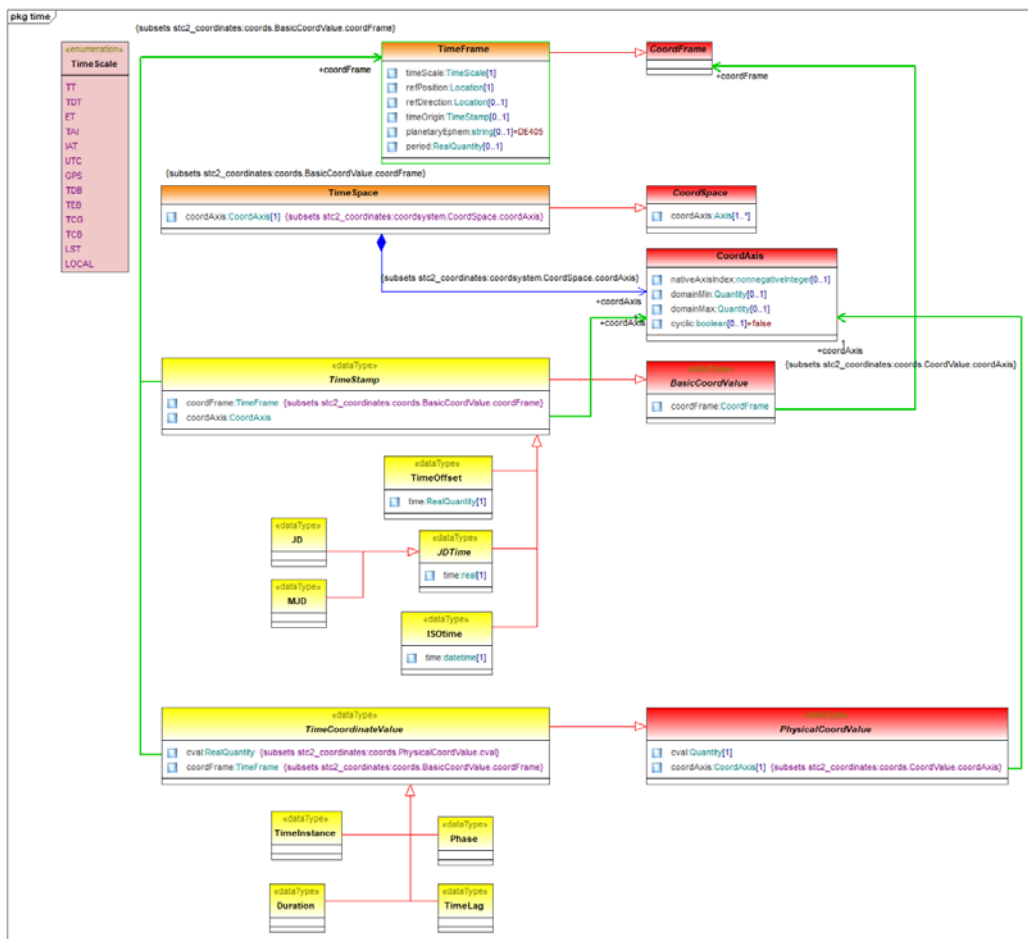
Quit a lot, especially the so-called coordinate time scales (TCG and TCB).

Because TDB runs on average synchronously with TT, but in a very different potential well, you may have realized that this requires different values for fundamental physical constants in the barycenter. That's awkward and the coordinate time scales fix that by running at different rates. Eventually these may come into more common use, but at least for my lifetime I assume we will be sticking with TT and TDB. More in the cited A&A paper.

Package **time**

owner	<a href="#">domain</a>
properties	qualified name <code>stc2_coordinates::domain::time</code>
ownedDiagrams	<a href="#">TimeDomain</a>
ownedMember	<a href="#">Duration</a> <a href="#">ISOtime</a> <a href="#">JD</a> <a href="#">JDTime</a> <a href="#">MJD</a> <a href="#">Phase</a> <a href="#">TimeCoordinateValue</a> <a href="#">TimeFrame</a> <a href="#">TimeInstance</a> <a href="#">TimeLag</a> <a href="#">TimeOffset</a> <a href="#">TimeScale</a> <a href="#">TimeSpace</a> <a href="#">TimeStamp</a>

Class Diagram **TimeDomain** ([time](#))





## 4.2.2 Class TimeFrame

diagram	
owner	<a href="#">time</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeFrame</code> abstract <code>false</code>
ownedMember	<a href="#">period</a> <a href="#">planetaryEphem</a> <a href="#">refDirection</a> <a href="#">refPosition</a> <a href="#">Subset</a> <a href="#">timeOrigin</a> <a href="#">timeScale</a>
general	<a href="#">CoordFrame</a>
typedElements	Class <a href="#">AstroCoordSystem</a> Property <a href="#">timeFrame</a> DataType <a href="#">TimeCoordinateValue</a> Property <a href="#">coordFrame</a> DataType <a href="#">TimeStamp</a> Property <a href="#">coordFrame</a>
shown on diagram	<a href="#">CoordSystems</a> <a href="#">TimeDomain</a>
documentation	A TimeFrame SHALL include a Time Scale and a TimeSpace and MAY include a Reference Direction, and/or a Time Origin (for elapsed time).

### Property TimeFrame::period

owner	<a href="#">TimeFrame</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeFrame::period</code> multiplicity <code>0..1</code> type <a href="#">RealQuantity</a>

### Property TimeFrame::planetaryEphem

owner	<a href="#">TimeFrame</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeFrame::planetaryEphem</code> multiplicity <code>0..1</code> type <code>string</code> default <code>DE405</code>

### Property TimeFrame::refDirection

owner	<a href="#">TimeFrame</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeFrame::refDirection</code> multiplicity <code>0..1</code> type <a href="#">Location</a>
documentation	The reference direction is needed if time stamps are transformed to a time frame with a different reference position (contained in the (Astro)CoordSystem). In those situations the solar system ephemeris also comes into play.

### Property TimeFrame::refPosition

owner	<a href="#">TimeFrame</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeFrame::refPosition</code> multiplicity <code>1</code> type <a href="#">Location</a>

### Property TimeFrame::timeOrigin

owner	<a href="#">TimeFrame</a>
-------	---------------------------

## STC2: I Coordinate Systems, Frames, and Values

properties	qualified name <code>stc2_coordinates::domain::time::TimeFrame::timeOrigin</code> multiplicity <code>0..1</code> type <a href="#">TimeStamp</a>
documentation	The Time Origin needs to refer to the same TimeAxis as the TimeOffsets that refer to this TimeFrame. This may seem like a recursive anomaly, but is necessary in order to prevent accidents.

### Property `TimeFrame::timeScale`

owner	<a href="#">TimeFrame</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeFrame::timeScale</code> multiplicity <code>1</code> type <a href="#">TimeScale</a>
documentation	The Time Scale sets the reference frame for the Time Frame.

### Enumeration `TimeScale`

diagram	Timescale	Description	Comments
	TT	Terrestrial Time	
	TDT	Terrestrial Dynamic Time; synonym for TT	
	ET	Ephemeris Time: predecessor of, and continuous with, TT	
	TAI	International Atomic Time; 32.184 s behind TT	
	IAT	Synonym for TAI	
	UTC	Coordinated Universal Time; 32 s behind TAI in 2000-2005	Includes leap seconds Pre-1972 times will be assumed to be UT/GMT
	GPS	Global Positioning System's time scale; 19 s behind TAI, 51.184 s behind TT	This time scale may become important in the future
	TDB	Barycentric Dynamical Time; synchronous with TT, except for variations in earth's orbital motion	Requires specification of the solar system and planetary ephemeris used
	TEB	Barycentric Ephemeris Time; independent variable in solar system ephemeris, linear function of TT	In most cases where TEB is specified, TDB is really the one used
	TCG	Geocentric Coordinate Time; properly relativistic time, running a factor $7 \cdot 10^{-10}$ faster than TT	
	TCB	Barycentric Coordinate Time; properly relativistic time, running a factor $1.5 \cdot 10^{-8}$ faster than TDB	
	LST	Local Siderial Time	Ground-based observations only
	LOCAL	"Local" time	Only to be used for simulations, in conjunction with RELOCATABLE spatial coordinates
owner	<a href="#">time</a>		
properties	qualified name <code>stc2_coordinates::domain::time::TimeScale</code> abstract <code>false</code>		
ownedMember	<b>ET GPS IAT LOCAL LST TAI TCB TCG TDB TDT TEB TT UTC</b>		
typedElements	Class <a href="#">TimeFrame</a> Property <a href="#">timeScale</a>		
shown on diagram	<a href="#">TimeDomain</a>		
documentation	The enumerated list of recognized Time Scales		

### 4.2.3 Class TimeSpace

diagram	
owner	<a href="#">time</a>
properties	qualified name stc2_coordinates::domain::time::TimeSpace abstract false
ownedMember	<a href="#">coordAxis Subset</a>
general	<a href="#">CoordSpace</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	The TimeSpace SHALL include one CoordAxis object, which will typically not include any of its optional attributes other than Name, although this will be left open in case it is deemed desirable to allow derived time axes such as phase to be specified.

#### Constraint TimeSpace::Subset

owner	<a href="#">TimeSpace</a>
properties	qualified name stc2_coordinates::domain::time::TimeSpace::Subset specification subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis constrained elements <a href="#">coordAxis</a>

#### Property TimeSpace::coordAxis

owner	<a href="#">TimeSpace</a>
properties	qualified name stc2_coordinates::domain::time::TimeSpace::coordAxis multiplicity 1 type <a href="#">CoordAxis</a> aggregation composite
constraints	<a href="#">Subset:</a> subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis

### 4.2.4 DataType TimeStamp

owner	<a href="#">time</a>
properties	qualified name stc2_coordinates::domain::time::TimeStamp abstract true
ownedMember	<a href="#">coordAxis coordFrame Subset</a>
general	<a href="#">BasicCoordValue</a>
specific	<a href="#">ISOtime JDTime TimeOffset</a>
typedElements	Class <a href="#">TimeFrame</a> Property <a href="#">timeOrigin</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	A TimeStamp specifies a specific time instance and may be expressed as JD, MJD, ISO-8601, or an offset from a specific point in time (which has to be a TimeStamp). The contents of the TimeStamp references an instance of a time axis and a TimeFrame.

#### Constraint TimeStamp::Subset

owner	<a href="#">TimeStamp</a>
properties	qualified name stc2_coordinates::domain::time::TimeStamp::Subset specification subsets stc2_coordinates:coords.BasicCoordValue.coordFrame

	constrained elements <a href="#">coordFrame</a>
--	---

Property **TimeStamp::coordAxis**

owner	<a href="#">TimeStamp</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeStamp::coordAxis</code> type <a href="#">CoordAxis</a>

Property **TimeStamp::coordFrame**

owner	<a href="#">TimeStamp</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeStamp::coordFrame</code> type <a href="#">TimeFrame</a>
constraints	<a href="#">Subset</a> : subsets <code>stc2_coordinates::coords.BasicCoordValue.coordFrame</code>

### 4.2.5 DataType TimeOffset

owner	<a href="#">time</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeOffset</code> abstract <code>false</code>
ownedMember	<a href="#">time</a>
general	<a href="#">TimeStamp</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	TimeOffset specifies the offset in time of the TimeStamp relative to the TimeOrigin in the associated TimeFrame.

Property **TimeOffset::time**

owner	<a href="#">TimeOffset</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeOffset::time</code> multiplicity <code>1</code> type <a href="#">RealQuantity</a>

### 4.2.6 DataType JDTime

owner	<a href="#">time</a>
properties	qualified name <code>stc2_coordinates::domain::time::JDTime</code> abstract <code>true</code>
ownedMember	<a href="#">time</a>
general	<a href="#">TimeStamp</a>
specific	<a href="#">JD_MJD</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	Basic astronomical time is specified as Julian Dates (or Modified Julian Dates). The numbers are unitless, although the implied unit is, of course, 'd'.

Property **JDTime::time**

owner	<a href="#">JDTime</a>
properties	qualified name <code>stc2_coordinates::domain::time::JDTime::time</code> multiplicity <code>1</code> type <code>real</code>

### 4.2.7 DataType JD

owner	<a href="#">time</a>
properties	qualified name <code>stc2_coordinates::domain::time::JD</code> abstract <code>false</code>
general	<a href="#">JDTime</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	JD is a Time Stamp expressed in Julian Days. Note that JD does not properly specify a time stamp, unless it is related to a time scale and a reference position. One should be aware that precision can easily become an issue with JD, as the numbers tend to be large.

### 4.2.8 DataType MJD

owner	<a href="#">time</a>
properties	qualified name <code>stc2_coordinates::domain::time::MJD</code> abstract <code>false</code>
general	<a href="#">JDTime</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	MJD is a Time Stamp expressed in Modified Julian Days. $T(\text{MJD})=T(\text{JD})-2440000.5$ .

### 4.2.9 DataType ISOtime

owner	<a href="#">time</a>
properties	qualified name <code>stc2_coordinates::domain::time::ISOtime</code> abstract <code>false</code>
ownedMember	<a href="#">time</a>
general	<a href="#">TimeStamp</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	ISOtime is a Time Stamp expressed as an ISO-8601 string, within the restrictions imposed by the IVOA: no time zone information is allowed, the string needs to be of the form <code>[+ -]ccyy-mm-dd[Thh:mm:ss[.s...]]</code>

#### Property ISOtime::time

owner	<a href="#">ISOtime</a>
properties	qualified name <code>stc2_coordinates::domain::time::ISOtime::time</code> multiplicity <code>1</code> type <code>datetime</code>

### 4.2.10 DataType TimeCoordinateValue

owner	<a href="#">time</a>
properties	qualified name <code>stc2_coordinates::domain::time::TimeCoordinateValue</code> abstract <code>true</code>
ownedMember	<a href="#">coordFrame</a> <a href="#">cval</a> <a href="#">Subset1</a> <a href="#">Subset2</a>
general	<a href="#">PhysicalCoordValue</a>
specific	<a href="#">Duration</a> <a href="#">Phase</a> <a href="#">TimeInstance</a> <a href="#">TimeLag</a>

shown on diagram	<a href="#">TimeDomain</a>
documentation	A TimeStampValue specifies a specific time instance and may be expressed as JD, MJD, ISO-8601, or an offset from a specific point in time (which has to be a Time Stamp). The contents of the Time Stamp references an instance of a time axis.

**Constraint TimeCoordinateValue::Subset1**

owner	<a href="#">TimeCoordinateValue</a>
properties	qualified name stc2_coordinates::domain::time::TimeCoordinateValue::Subset1 specification subsets stc2_coordinates:coords.PhysicalCoordValue.cval constrained elements <a href="#">cval</a>

**Constraint TimeCoordinateValue::Subset2**

owner	<a href="#">TimeCoordinateValue</a>
properties	qualified name stc2_coordinates::domain::time::TimeCoordinateValue::Subset2 specification subsets stc2_coordinates:coords.BasicCoordValue.coordFrame constrained elements <a href="#">coordFrame</a>

**Property TimeCoordinateValue::coordFrame**

owner	<a href="#">TimeCoordinateValue</a>
properties	qualified name stc2_coordinates::domain::time::TimeCoordinateValue::coordFrame type <a href="#">TimeFrame</a>
constraints	<a href="#">Subset2</a> : subsets stc2_coordinates:coords.BasicCoordValue.coordFrame

**Property TimeCoordinateValue::cval**

owner	<a href="#">TimeCoordinateValue</a>
properties	qualified name stc2_coordinates::domain::time::TimeCoordinateValue::cval type <a href="#">RealQuantity</a>
constraints	<a href="#">Subset1</a> : subsets stc2_coordinates:coords.PhysicalCoordValue.cval

### 4.2.11 DataType TimeInstance

owner	<a href="#">time</a>
properties	qualified name stc2_coordinates::domain::time::TimeInstance abstract false
general	<a href="#">TimeCoordinateValue</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	TimeInstance specifies a time instance in the associated TimeFrame.

### 4.2.12 DataType Duration

owner	<a href="#">time</a>
properties	qualified name stc2_coordinates::domain::time::Duration abstract false
general	<a href="#">TimeCoordinateValue</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	Duration specifies a time duration in the associated TimeFrame.

### 4.2.13 DataType Phase

owner	<a href="#">time</a>
properties	qualified name stc2_coordinates::domain::time::Phase abstract false
general	<a href="#">TimeCoordinateValue</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	Phase specifies a phase instance in the associated TimeFrame.

### 4.2.14 DataType TimeLag

owner	<a href="#">time</a>
properties	qualified name stc2_coordinates::domain::time::TimeLag abstract false
general	<a href="#">TimeCoordinateValue</a>
shown on diagram	<a href="#">TimeDomain</a>
documentation	TimeLag specifies a time lag in the associated TimeFrame.

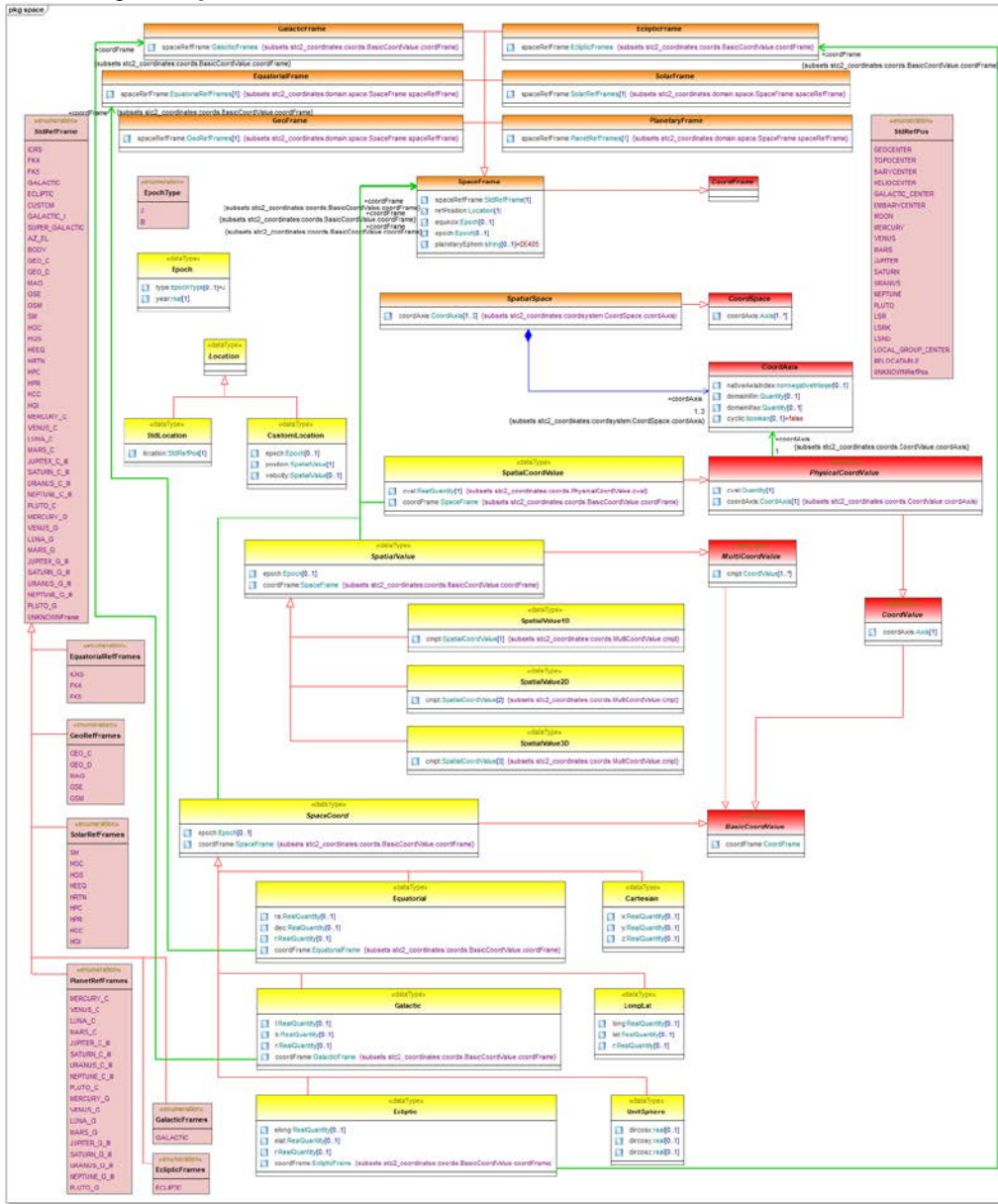
## 4.3 Spatial Domain

Spatial coordinates may be 1-, 2-, or 3-dimensional, Cartesian or spherical. In the future one may consider adding direction cosines (3-D unit sphere), polar, HealPix, HTM, etc. A reference frame (orientation) and reference position (origin) are required. Positions as well as velocities (but not redshifts or Doppler velocities) are the coordinate types in this domain.

### Package space

owner	<a href="#">domain</a>
properties	qualified name stc2_coordinates::domain::space
ownedDiagrams	<a href="#">SpatialDomain</a>
ownedMember	<a href="#">Cartesian</a> <a href="#">CustomLocation</a> <a href="#">Ecliptic</a> <a href="#">EclipticFrame</a> <a href="#">EclipticFrames</a> <a href="#">Epoch</a> <a href="#">EpochType</a> <a href="#">Equatorial</a> <a href="#">EquatorialFrame</a> <a href="#">EquatorialRefFrames</a> <a href="#">Galactic</a> <a href="#">GalacticFrame</a> <a href="#">GalacticFrames</a> <a href="#">GeoFrame</a> <a href="#">GeoRefFrames</a> <a href="#">Location</a> <a href="#">LongLat</a> <a href="#">PlanetaryFrame</a> <a href="#">PlanetRefFrames</a> <a href="#">SolarFrame</a> <a href="#">SolarRefFrames</a> <a href="#">SpaceCoord</a> <a href="#">SpaceFrame</a> <a href="#">SpatialCoordValue</a> <a href="#">SpatialSpace</a> <a href="#">SpatialValue</a> <a href="#">SpatialValue1D</a> <a href="#">SpatialValue2D</a> <a href="#">SpatialValue3D</a> <a href="#">StdLocation</a> <a href="#">StdRefFrame</a> <a href="#">StdRefPos</a> <a href="#">UnitSphere</a>

Class Diagram **SpatialDomain** (*space*)



4.3.1 Class SpaceFrame

diagram	<pre> classDiagram     class SpaceFrame {         spaceRefFrame: StdRefFrame[1]         refPosition: Location[1]         equinox: Epoch[0..1]         epoch: Epoch[0..1]         planetaryEphem: string[0..1]=DE405     }         </pre>
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpaceFrame</code> abstract <code>false</code>



ownedMember	<a href="#">epoch</a> <a href="#">equinox</a> <a href="#">planetaryEphem</a> <a href="#">refPosition</a> <a href="#">spaceRefFrame</a> <a href="#">Subset</a>																
general	<a href="#">CoordFrame</a>																
specific	<a href="#">EclipticFrame</a> <a href="#">EquatorialFrame</a> <a href="#">GalacticFrame</a> <a href="#">GeoFrame</a> <a href="#">PlanetaryFrame</a> <a href="#">SolarFrame</a>																
typedElements	<table border="0"> <tr> <td>Class</td> <td><a href="#">AstroCoordSystem</a></td> <td>Property</td> <td><a href="#">spaceFrame</a></td> </tr> <tr> <td>Data Type</td> <td><a href="#">SpaceCoord</a></td> <td>Property</td> <td><a href="#">coordFrame</a></td> </tr> <tr> <td>Data Type</td> <td><a href="#">SpatialCoordValue</a></td> <td>Property</td> <td><a href="#">coordFrame</a></td> </tr> <tr> <td>Data Type</td> <td><a href="#">SpatialValue</a></td> <td>Property</td> <td><a href="#">coordFrame</a></td> </tr> </table>	Class	<a href="#">AstroCoordSystem</a>	Property	<a href="#">spaceFrame</a>	Data Type	<a href="#">SpaceCoord</a>	Property	<a href="#">coordFrame</a>	Data Type	<a href="#">SpatialCoordValue</a>	Property	<a href="#">coordFrame</a>	Data Type	<a href="#">SpatialValue</a>	Property	<a href="#">coordFrame</a>
Class	<a href="#">AstroCoordSystem</a>	Property	<a href="#">spaceFrame</a>														
Data Type	<a href="#">SpaceCoord</a>	Property	<a href="#">coordFrame</a>														
Data Type	<a href="#">SpatialCoordValue</a>	Property	<a href="#">coordFrame</a>														
Data Type	<a href="#">SpatialValue</a>	Property	<a href="#">coordFrame</a>														
shown on diagram	<a href="#">CoordSystems</a> <a href="#">SpatialDomain</a>																
documentation	A Space Frame is specified by its Reference Frame (currently only standard reference frames are allowed) and a Reference Position (provided in the associated AstroCoordSystem object); an Equinox is optional and only required for pre-ICRS reference frames; an epoch is also optional. A planetary ephemeris may be provided if relevant; the default is DE405. It is to be referenced by spatial coordinates (Position and Velocity).																

Property **SpaceFrame::epoch**

owner	<a href="#">SpaceFrame</a>						
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::SpaceFrame::epoch</td> </tr> <tr> <td>multiplicity</td> <td>0..1</td> </tr> <tr> <td>type</td> <td><a href="#">Epoch</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::space::SpaceFrame::epoch	multiplicity	0..1	type	<a href="#">Epoch</a>
qualified name	stc2_coordinates::domain::space::SpaceFrame::epoch						
multiplicity	0..1						
type	<a href="#">Epoch</a>						

Property **SpaceFrame::equinox**

owner	<a href="#">SpaceFrame</a>						
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::SpaceFrame::equinox</td> </tr> <tr> <td>multiplicity</td> <td>0..1</td> </tr> <tr> <td>type</td> <td><a href="#">Epoch</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::space::SpaceFrame::equinox	multiplicity	0..1	type	<a href="#">Epoch</a>
qualified name	stc2_coordinates::domain::space::SpaceFrame::equinox						
multiplicity	0..1						
type	<a href="#">Epoch</a>						

Property **SpaceFrame::planetaryEphem**

owner	<a href="#">SpaceFrame</a>								
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::SpaceFrame::planetaryEphem</td> </tr> <tr> <td>multiplicity</td> <td>0..1</td> </tr> <tr> <td>type</td> <td>string</td> </tr> <tr> <td>default</td> <td>DE405</td> </tr> </table>	qualified name	stc2_coordinates::domain::space::SpaceFrame::planetaryEphem	multiplicity	0..1	type	string	default	DE405
qualified name	stc2_coordinates::domain::space::SpaceFrame::planetaryEphem								
multiplicity	0..1								
type	string								
default	DE405								

Property **SpaceFrame::refPosition**

owner	<a href="#">SpaceFrame</a>						
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::SpaceFrame::refPosition</td> </tr> <tr> <td>multiplicity</td> <td>1</td> </tr> <tr> <td>type</td> <td><a href="#">Location</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::space::SpaceFrame::refPosition	multiplicity	1	type	<a href="#">Location</a>
qualified name	stc2_coordinates::domain::space::SpaceFrame::refPosition						
multiplicity	1						
type	<a href="#">Location</a>						

Property **SpaceFrame::spaceRefFrame**

owner	<a href="#">SpaceFrame</a>						
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::SpaceFrame::spaceRefFrame</td> </tr> <tr> <td>multiplicity</td> <td>1</td> </tr> <tr> <td>type</td> <td><a href="#">StdRefFrame</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::space::SpaceFrame::spaceRefFrame	multiplicity	1	type	<a href="#">StdRefFrame</a>
qualified name	stc2_coordinates::domain::space::SpaceFrame::spaceRefFrame						
multiplicity	1						
type	<a href="#">StdRefFrame</a>						

### 4.3.2 Enumeration StdRefFrame

diagram	Reference Frame	Description	Comments
	FK4	Fundamental Katalog, system 4; Besselian	Requires Equinox; default B1950.0 Left-handed in spherical coordinates

STC2: I Coordinate Systems, Frames, and Values

FK5	Fundamental Katalog, system 5; Julian	Requires Equinox; default J2000.0 Left-handed in spherical coordinates
ECLIPTIC	Ecliptic coordinates	Left-handed in spherical coordinates
ICRS	International Celestial Reference System	Left-handed in spherical coordinates
GALACTIC_I	Old Galactic coordinates	Left-handed in spherical coordinates
GALACTIC_II	“New” Galactic coordinates	Left-handed in spherical coordinates
SUPER_GALACTIC	Super-galactic coordinates: pole at GALACTIC_II (47.37,+6.32) origin at GALACTIC_II (137.37,0)	Left-handed in spherical coordinates
AZ_EL	Local azimuth and elevation	Ground-based observatories Azimuth: from north through east
BODY	Generic “BODY” coordinates	
GEO_C	Geographic (geocentric) coordinates: longitude, latitude, geocentric distance	3-D spherical or 3-D Cartesian
GEO_D	Geodetic coordinates: longitude, latitude, elevation	Semi-major axis and inverse flattening of the reference spheroid may need to be provided; default is IAU 1976 (6378140 m, 298.2577)
MAG	Geomagnetic coordinates	See F&H (2002)
GSE	Geocentric Solar Ecliptic coordinates	See F&H (2002)
GSM	Geocentric Solar Magnetic coordinates	See F&H (2002)
SM	Solar Magnetic coordinates	See F&H (2002)
HGC	Heliographic coordinates (Carrington)	See Explanatory Supplement, Section 7.2 Thompson (2006), Section 2.2
HGS	Heliographic coordinates (Stonyhurst)	See Explanatory Supplement, Section 7.2 Thompson (2006), Section 2.2
HEEQ	Heliographic Earth Equatorial coordinates	See F&H (2002); related to Heliographic (Stonyhurst), see Thompson (2006), Section 2.1
HRTN	Heliocentric Radial-Tangential-Normal coordinates	See F&H (2002)
HPC	Helioprojective Cartesian coordinates	See Thompson (2006), Section 4.1, 2- or 3-dimensional (angular coordinates); left-handed
HPR	Helioprojective Polar coordinates	See Thompson (2006), Section 4.1, 2-dimensional (angular coordinates); left-handed
HCC	Heliocentric Cartesian coordinates	See Thompson (2006), Section 3.1 (linear coordinates); right-handed
HGI	Heliographic Inertial coordinates	See F&H (2002)
MERCURY_C	Planetocentric coordinates on Mercury	See Explanatory Supplement, Section 7.4
VENUS_C	Planetocentric coordinates on Venus	See Explanatory Supplement, Section 7.4
LUNA_C	Selenocentric coordinates	See Explanatory Supplement, Section 7.3
MARS_C	Planetocentric coordinates on Mars	See Explanatory Supplement, Section 7.4
JUPITER_C_III	Planetocentric coordinates on Jupiter, system III	See Explanatory Supplement, Section 7.4
SATURN_C_III	Planetocentric coordinates on Saturn, system III	See Explanatory Supplement, Section 7.4
URANUS_C_III	Planetocentric coordinates on Uranus, system III	See Explanatory Supplement, Section 7.4
NEPTUNE_C_III	Planetocentric coordinates on Neptune, system III	See Explanatory Supplement, Section 7.4

	PLUTO_C	Planetocentric coordinates on Pluto	See Explanatory Supplement, Section 7.4
	MERCURY_G	Planetographic coordinates on Mercury	See Explanatory Supplement, Section 7.4 Left-handed
	VENUS_G	Planetographic coordinates on Venus	See Explanatory Supplement, Section 7.4
	LUNA_G	Selenographic coordinates	See Explanatory Supplement, Section 7.3
	MARS_G	Planetographic coordinates on Mars	See Explanatory Supplement, Section 7.4 Left-handed
	JUPITER_G_III	Planetographic coordinates on Jupiter, system III	See Explanatory Supplement, Section 7.4 Left-handed
	SATURN_G_III	Planetographic coordinates on Saturn, system III	See Explanatory Supplement, Section 7.4 Left-handed
	URANUS_G_III	Planetographic coordinates on Uranus, system III	See Explanatory Supplement, Section 7.4
	NEPTUNE_G_III	Planetographic coordinates on Neptune, system III	See Explanatory Supplement, Section 7.4 Left-handed
	PLUTO_G	Planetographic coordinates on Pluto	See Explanatory Supplement, Section 7.4
	UNKNOWNFrame	Unknown reference frame	Only to be used as a last resort or for simulations The client is responsible for assigning a suitable default
owner	<a href="#">space</a>		
properties	qualified name	stc2_coordinates::domain::space::StdRefFrame	
	abstract	false	
ownedMember	AZ_EL BODY CUSTOM ECLIPTIC FK4 FK5 GALACTIC GALACTIC_I GEO_C GEO_D GSE GSM HCC HEEQ HGC HGI HGS HPC HPR HRTN ICRS JUPITER_C_III JUPITER_G_III LUNA_C LUNA_G MAG MARS_C MARS_G MERCURY_C MERCURY_G NEPTUNE_C_III NEPTUNE_G_III PLUTO_C PLUTO_G SATURN_C_III SATURN_G_III SM SUPER_GALACTIC UNKNOWNFrame URANUS_C_III URANUS_G_III VENUS_C VENUS_G		
specific	<a href="#">EclipticFrames</a> <a href="#">EquatorialRefFrames</a> <a href="#">GalacticFrames</a> <a href="#">GeoRefFrames</a> <a href="#">PlanetRefFrames</a> <a href="#">SolarRefFrames</a>		
typedElements	Class	<a href="#">SpaceFrame</a>	Property <a href="#">spaceRefFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>		
documentation	This is the enumerated list of allowed standard spatial reference frames.		

### 4.3.3 DataType Location

owner	<a href="#">space</a>		
properties	qualified name	stc2_coordinates::domain::space::Location	
	abstract	true	
specific	<a href="#">CustomLocation</a> <a href="#">StdLocation</a>		
typedElements	Class	<a href="#">RedshiftFrame</a>	Property <a href="#">refPosition</a>
	Class	<a href="#">SpaceFrame</a>	Property <a href="#">refPosition</a>
	Class	<a href="#">SpectralFrame</a>	Property <a href="#">refPosition</a>
	Class	<a href="#">TimeFrame</a>	Property <a href="#">refDirection</a> <a href="#">refPosition</a>
shown on diagram	<a href="#">SpatialDomain</a>		
documentation	The abstract data type that specifies a specific spatial location. It may either be a StdLocation (which absolute location in phase space is known by definition, if necessary aided by a planetary ephemeris) or a Custom Spatial Location (which provides coordinates in an associated SpaceFrame). Although this data type may be used anywhere, it is especially designed for us as a Reference Position in the AstroCoordSystem object.		

### 4.3.4 DataType CustomLocation

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::CustomLocation</code> abstract <code>false</code>
ownedMember	<a href="#">epoch</a> <a href="#">position</a> <a href="#">velocity</a>
general	<a href="#">Location</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Derived from data type Location. This is the abstract data type (to be instantiated as a set of 1D, 2D, or 3D spatial coordinates specifying the position and/or velocity. This abstract data type contains a reference to an instance of a SpatialFrame and, optionally, an Epoch.

#### Property CustomLocation::epoch

owner	<a href="#">CustomLocation</a>
properties	qualified name <code>stc2_coordinates::domain::space::CustomLocation::epoch</code> multiplicity <code>0..1</code> type <a href="#">Epoch</a>

#### Property CustomLocation::position

owner	<a href="#">CustomLocation</a>
properties	qualified name <code>stc2_coordinates::domain::space::CustomLocation::position</code> multiplicity <code>1</code> type <a href="#">SpatialValue</a>

#### Property CustomLocation::velocity

owner	<a href="#">CustomLocation</a>
properties	qualified name <code>stc2_coordinates::domain::space::CustomLocation::velocity</code> multiplicity <code>0..1</code> type <a href="#">SpatialValue</a>

### 4.3.5 DataType StdLocation

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::StdLocation</code> abstract <code>false</code>
ownedMember	<a href="#">location</a>
general	<a href="#">Location</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Derived from data type Location. An absolute a priori known location in phase space (position and velocity), chosen from the StdRefPos enumeration. Considering that the GEOCENTER is really the only place for which we know the absolute location at all times, all other locations require the specification of a planetary ephemeris, although that may be defaulted to DE405. LSR[KD] are reserved for spectral and redshift frames. TOPOCENTER (location of the observer) is special in that it assumes that the observing location is available through other means (e.g., a geographic location or an orbit ephemeris). RELOCATABLE is available for simulations. UNKNOWNRefPos should only be used if absolutely necessary.

#### Property StdLocation::location

owner	<a href="#">StdLocation</a>
properties	qualified name <code>stc2_coordinates::domain::space::StdLocation::location</code> multiplicity <code>1</code>

	type <a href="#">StdRefPos</a>
--	--------------------------------

### 4.3.6 Enumeration StdRefPos

diagram	Reference Position	Description	Comments
	GEOCENTER	Center of the earth	
	BARYCENTER	Center of the solar system barycenter	
	HELIOCENTER	Center of the sun	
	TOPOCENTER	“Local”; in most cases this will mean: the location of the telescope	
	LSR or LSRK	Kinematic Local Standard of Rest: 20 km s <sup>-1</sup> in the direction of GALACTIC_II(56,+23)	Only to be used for redshifts and Doppler velocities, and spectral coordinate
	LSRD	Dynamic Local Standard of Rest: 16.6 km s <sup>-1</sup> in the direction of GALACTIC_II(53,+25)	Only to be used for redshifts and Doppler velocities, and spectral coordinate
	GALACTIC_CENTER	Center of the Galaxy: 220 km s <sup>-1</sup> in the direction of GALACTIC_II(90,0) wrt LSRD	
	LOCAL_GROUP_CENTER	Center of the Local Group: 300 km s <sup>-1</sup> in the direction of GALACTIC_II(90,0) wrt BARYCENTER	Only to be used for redshifts and Doppler velocities, and spectral coordinate
	EMBARYCENTER	Earth-moon barycenter	
	MOON	Center of the moon	
	MERCURY	Center of Mercury	
	VENUS	Center of Venus	
	MARS	Center of Mars	
	JUPITER	Center of Jupiter	
	SATURN	Center of Saturn	
	URANUS	Center of Uranus	
	NEPTUNE	Center of Neptune	
	PLUTO	Center of Pluto	
	RELOCATABLE	Relocatable center; for simulations	Only to be used for spatial coordinates
	UNKNOWNRefPos	Unknown reference position	Only to be used as a last resort. The client is responsible for assigning a suitable default
owner	<a href="#">space</a>		
properties	qualified name <code>stc2_coordinates::domain::space::StdRefPos</code> abstract <code>false</code>		
ownedMember	<b>BARYCENTER EMBARYCENTER GALACTIC_CENTER GEOCENTER HELIOCENTER JUPITER LOCAL_GROUP_CENTER LSR LSRD LSRK MARS MERCURY MOON NEPTUNE PLUTO RELOCATABLE SATURN TOPOCENTER UNKNOWNRefPos URANUS VENUS</b>		
typedElements	DataType <a href="#">StdLocation</a> Property <a href="#">location</a>		
shown on diagram	<a href="#">SpatialDomain</a>		
documentation	This is the enumerated list of allowed standard spatial and temporal reference positions.		

### 4.3.7 DataType Epoch

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::Epoch</code> abstract <code>false</code>
ownedMember	<a href="#">type</a> <a href="#">year</a>
typedElements	DataType <a href="#">CustomLocation</a> Property <a href="#">epoch</a> DataType <a href="#">SpaceCoord</a> Property <a href="#">epoch</a> Class <a href="#">SpaceFrame</a> Property <a href="#">epoch</a> <a href="#">equinox</a> DataType <a href="#">SpatialValue</a> Property <a href="#">epoch</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Epoch data type, containing EpochType (default: J) and epoch expressed in years.

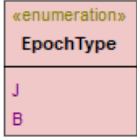
#### Property Epoch::type

owner	<a href="#">Epoch</a>
properties	qualified name <code>stc2_coordinates::domain::space::Epoch::type</code> multiplicity <code>0..1</code> type <a href="#">EpochType</a> default <code>J</code>

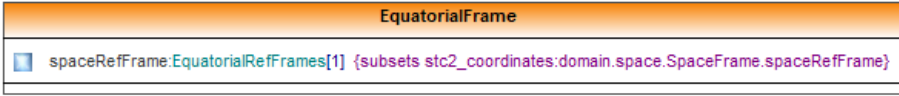
#### Property Epoch::year

owner	<a href="#">Epoch</a>
properties	qualified name <code>stc2_coordinates::domain::space::Epoch::year</code> multiplicity <code>1</code> type <code>real</code>

#### Enumeration EpochType

diagram	
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::EpochType</code> abstract <code>false</code>
ownedMember	<b>B J</b>
typedElements	DataType <a href="#">Epoch</a> Property <a href="#">type</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Epoch type: Julian or Besselian.

### 4.3.8 Class EquatorialFrame

diagram	
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::EquatorialFrame</code> abstract <code>false</code>

ownedMember	<a href="#">spaceRefFrame Subset</a>
general	<a href="#">SpaceFrame</a>
typedElements	Data Type <a href="#">Equatorial</a> Property <a href="#">coordFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Specialized spatial coordinate frame: only equatorial coordinates (ICRS, FK4, FK5) are allowed.

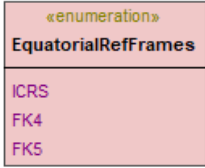
### Constraint [EquatorialFrame::Subset](#)

owner	<a href="#">EquatorialFrame</a>
properties	qualified name <code>stc2_coordinates::domain::space::EquatorialFrame::Subset</code> specification <code>subsets stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame</code> constrained elements <a href="#">spaceRefFrame</a>


### Property [EquatorialFrame::spaceRefFrame](#)

owner	<a href="#">EquatorialFrame</a>
properties	qualified name <code>stc2_coordinates::domain::space::EquatorialFrame::spaceRefFrame</code> multiplicity <code>1</code> type <a href="#">EquatorialRefFrames</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame</code>

### Enumeration [EquatorialRefFrames](#)

diagram	
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::EquatorialRefFrames</code> abstract <code>false</code>
ownedMember	<b>FK4 FK5 ICRS</b>
general	<a href="#">StdRefFrame</a>
typedElements	Class <a href="#">EquatorialFrame</a> Property <a href="#">spaceRefFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	This is the enumerated subset list of allowed standard equatorial spatial reference frames.

## 4.3.9 Class [GalacticFrame](#)

diagram	
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::GalacticFrame</code> abstract <code>false</code>
ownedMember	<a href="#">spaceRefFrame Subset</a>

general	<a href="#">SpaceFrame</a>
typedElements	DataType <a href="#">Galactic</a> Property <a href="#">coordFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Specialized spatial coordinate frame: only Galactic coordinates (GALACTIC) are allowed.

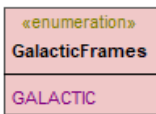
**Constraint GalacticFrame::Subset**

owner	<a href="#">GalacticFrame</a>
properties	qualified name <code>stc2_coordinates::domain::space::GalacticFrame::Subset</code> specification <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code> constrained elements <a href="#">spaceRefFrame</a>

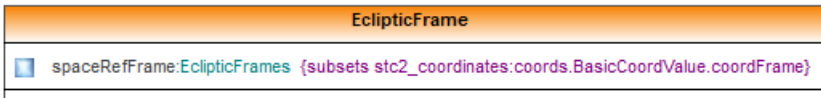
**Property GalacticFrame::spaceRefFrame**

owner	<a href="#">GalacticFrame</a>
properties	qualified name <code>stc2_coordinates::domain::space::GalacticFrame::spaceRefFrame</code> type <a href="#">GalacticFrames</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code>

**Enumeration GalacticFrames**

diagram	
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::GalacticFrames</code> abstract <code>false</code>
ownedMember	<b>GALACTIC</b>
general	<a href="#">StdRefFrame</a>
typedElements	Class <a href="#">GalacticFrame</a> Property <a href="#">spaceRefFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	This is the enumerated subset list of allowed standard Galactic spatial reference frames; GALACTIC only.

**4.3.10 Class EclipticFrame**

diagram	
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::EclipticFrame</code> abstract <code>false</code>
ownedMember	<a href="#">spaceRefFrame Subset</a>
general	<a href="#">SpaceFrame</a>
typedElements	DataType <a href="#">Ecliptic</a> Property <a href="#">coordFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>



documentation	Specialized spatial coordinate frame: only ecliptic coordinates (ECLIPTIC) are allowed.
---------------	---

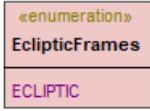
**Constraint `EclipticFrame::Subset`**

owner	<a href="#">EclipticFrame</a>
properties	qualified name <code>stc2_coordinates::domain::space::EclipticFrame::Subset</code> specification <code>subsets stc2_coordinates::coords.BasicCoordValue.coordFrame</code> constrained elements <a href="#">spaceRefFrame</a>

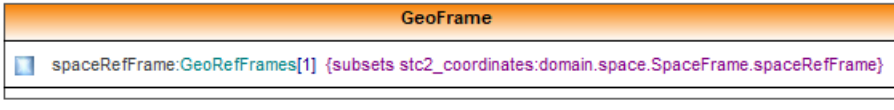
**Property `EclipticFrame::spaceRefFrame`**

owner	<a href="#">EclipticFrame</a>
properties	qualified name <code>stc2_coordinates::domain::space::EclipticFrame::spaceRefFrame</code> type <a href="#">EclipticFrames</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates::coords.BasicCoordValue.coordFrame</code>

**Enumeration `EclipticFrames`**

diagram	
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::EclipticFrames</code> abstract <code>false</code>
ownedMember	<b>ECLIPTIC</b>
general	<a href="#">StdRefFrame</a>
typedElements	Class <a href="#">EclipticFrame</a> Property <a href="#">spaceRefFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	This is the enumerated subset list of allowed standard ecliptic spatial reference frames. ECLIPTIC only.

**4.3.11 Class `GeoFrame`**

diagram	
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::GeoFrame</code> abstract <code>false</code>
ownedMember	<a href="#">spaceRefFrame Subset</a>
general	<a href="#">SpaceFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Specialized spatial coordinate frame: only geographic coordinates are allowed.

**Constraint `GeoFrame::Subset`**

owner	<a href="#">GeoFrame</a>
properties	qualified name <code>stc2_coordinates::domain::space::GeoFrame::Subset</code>

	specification	subsets stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame
	constrained elements	<a href="#">spaceRefFrame</a>

**Property GeoFrame::spaceRefFrame**

owner	<a href="#">GeoFrame</a>	
properties	qualified name	stc2_coordinates::domain::space::GeoFrame::spaceRefFrame
	multiplicity	1
	type	<a href="#">GeoRefFrames</a>
constraints	<a href="#">Subset:</a>	subsets stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame

**Enumeration GeoRefFrames**

diagram		
owner	<a href="#">space</a>	
properties	qualified name	stc2_coordinates::domain::space::GeoRefFrames
	abstract	false
ownedMember	<b>GEO_C GEO_D GSE GSM MAG</b>	
general	<a href="#">StdRefFrame</a>	
typedElements	Class <a href="#">GeoFrame</a>	Property <a href="#">spaceRefFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>	
documentation	This is the enumerated subset list of allowed standard geographic spatial reference frames.	

**4.3.12 Class SolarFrame**

diagram		
owner	<a href="#">space</a>	
properties	qualified name	stc2_coordinates::domain::space::SolarFrame
	abstract	false
ownedMember	<a href="#">spaceRefFrame</a> <a href="#">Subset</a>	
general	<a href="#">SpaceFrame</a>	
shown on diagram	<a href="#">SpatialDomain</a>	
documentation	Specialized spatial coordinate frame: only solar coordinates are allowed.	

**Constraint SolarFrame::Subset**

owner	<a href="#">SolarFrame</a>	
properties	qualified name	stc2_coordinates::domain::space::SolarFrame::Subset
	specification	subsets stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame
	constrained elements	<a href="#">spaceRefFrame</a>

Property **SolarFrame::spaceRefFrame**

owner	<a href="#">SolarFrame</a>
properties	qualified name <code>stc2_coordinates::domain::space::SolarFrame::spaceRefFrame</code> multiplicity 1 type <a href="#">SolarRefFrames</a>
constraints	<a href="#">Subset</a> : subsets <code>stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame</code>

Enumeration **SolarRefFrames**

diagram	<pre> classDiagram     class SolarRefFrames {         SM         HGC         HGS         HEEQ         HRTN         HPC         HPR         HCC         HGI     }         </pre>
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::SolarRefFrames</code> abstract false
ownedMember	<b>HCC HEEQ HGC HGI HGS HPC HPR HRTN SM</b>
general	<a href="#">StdRefFrame</a>
typedElements	Class <a href="#">SolarFrame</a> Property <a href="#">spaceRefFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	This is the enumerated subset list of allowed standard solar spatial reference frames.

### 4.3.13 Class PlanetaryFrame

diagram	<pre> classDiagram     class PlanetaryFrame {         spaceRefFrame:PlanetRefFrames[1] {subsets stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame}     }         </pre>
owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::PlanetaryFrame</code> abstract false
ownedMember	<a href="#">spaceRefFrame Subset</a>
general	<a href="#">SpaceFrame</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Specialized spatial coordinate frame: only planetary coordinates are allowed.

Constraint **PlanetaryFrame::Subset**

owner	<a href="#">PlanetaryFrame</a>
-------	--------------------------------

## STC2: I Coordinate Systems, Frames, and Values

properties	qualified name specification constrained elements	stc2_coordinates::domain::space::PlanetaryFrame::Subset subsets stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame <a href="#">spaceRefFrame</a>
------------	---	--

### Property **PlanetaryFrame::spaceRefFrame**

owner	<a href="#">PlanetaryFrame</a>	
properties	qualified name multiplicity type	stc2_coordinates::domain::space::PlanetaryFrame::spaceRefFrame 1 <a href="#">PlanetRefFrames</a>
constraints	<a href="#">Subset:</a> subsets stc2_coordinates:domain.space.SpaceFrame.spaceRefFrame	

### Enumeration **PlanetRefFrames**

diagram	<pre> classDiagram     class PlanetRefFrames {         MERCURY_C         VENUS_C         LUNA_C         MARS_C         JUPITER_C_III         SATURN_C_III         URANUS_C_III         NEPTUNE_C_III         PLUTO_C         MERCURY_G         VENUS_G         LUNA_G         MARS_G         JUPITER_G_III         SATURN_G_III         URANUS_G_III         NEPTUNE_G_III         PLUTO_G     }         </pre>	
owner	<a href="#">space</a>	
properties	qualified name abstract	stc2_coordinates::domain::space::PlanetRefFrames false
ownedMember	JUPITER_C_III JUPITER_G_III LUNA_C LUNA_G MARS_C MARS_G MERCURY_C MERCURY_G NEPTUNE_C_III NEPTUNE_G_III PLUTO_C PLUTO_G SATURN_C_III SATURN_G_III URANUS_C_III URANUS_G_III VENUS_C VENUS_G	
general	<a href="#">StdRefFrame</a>	
typedElements	Class <a href="#">PlanetaryFrame</a> Property <a href="#">spaceRefFrame</a>	
shown on diagram	<a href="#">SpatialDomain</a>	
documentation	This is the enumerated list of allowed standard planetary reference frames.	

### 4.3.14 Class **SpatialSpace**

diagram	<pre> classDiagram     class SpatialSpace {         coordAxis:CoordAxis[1..3] {subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis}     }         </pre>
owner	<a href="#">space</a>

properties	qualified name abstract	stc2_coordinates::domain::space::SpatialSpace true
ownedMember	<a href="#">coordAxis Subset</a>	
general	<a href="#">CoordSpace</a>	
shown on diagram	<a href="#">SpatialDomain</a>	
documentation	The abstract SpatialSpace is derived from CoordSpace and SHALL contain one, two, or three coordinate axes whose attributes may all be relevant. Different flavors may be derived from this class: Spherical, Cartesian, Cylindrical, Polar, UnitSphere, HTM, HealPix. At this time only Spherical, Cartesian, and UnitSphere are provided.	

**Constraint SpatialSpace::Subset**

owner	<a href="#">SpatialSpace</a>	
properties	qualified name specification constrained elements	stc2_coordinates::domain::space::SpatialSpace::Subset subsets stc2_coordinates::coordsystem::CoordSpace::coordAxis <a href="#">coordAxis</a>

**Property SpatialSpace::coordAxis**

owner	<a href="#">SpatialSpace</a>	
properties	qualified name multiplicity type aggregation	stc2_coordinates::domain::space::SpatialSpace::coordAxis 1..3 <a href="#">CoordAxis</a> composite
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coordsystem::CoordSpace::coordAxis	

**4.3.15 DataType SpatialCoordValue**

owner	<a href="#">space</a>	
properties	qualified name abstract	stc2_coordinates::domain::space::SpatialCoordValue false
ownedMember	<a href="#">coordFrame cval Subset1 Subset2</a>	
general	<a href="#">PhysicalCoordValue</a>	
typedElements	DataType DataType DataType	<a href="#">SpatialValue1D</a> <a href="#">SpatialValue2D</a> <a href="#">SpatialValue3D</a> Property <a href="#">cmpt</a> <a href="#">cmpt</a> <a href="#">cmpt</a>
shown on diagram	<a href="#">SpatialDomain</a>	
documentation	Single-axis spatial coordinate value, to be used as component in the general SpatialValue data type.	

**Constraint SpatialCoordValue::Subset1**

owner	<a href="#">SpatialCoordValue</a>	
properties	qualified name specification constrained elements	stc2_coordinates::domain::space::SpatialCoordValue::Subset1 subsets stc2_coordinates::coords::PhysicalCoordValue::cval <a href="#">cval</a>

**Constraint SpatialCoordValue::Subset2**

owner	<a href="#">SpatialCoordValue</a>	
properties	qualified name specification constrained elements	stc2_coordinates::domain::space::SpatialCoordValue::Subset2 subsets stc2_coordinates::coords::BasicCoordValue::coordFrame <a href="#">coordFrame</a>

Property **SpatialCoordValue::coordFrame**

owner	<a href="#">SpatialCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialCoordValue::coordFrame</code>
constraints	<a href="#">Subset2</a> : subsets <code>stc2_coordinates::coords.BasicCoordValue.coordFrame</code>

Property **SpatialCoordValue::cval**

owner	<a href="#">SpatialCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialCoordValue::cval</code> multiplicity 1 type <a href="#">RealQuantity</a>
constraints	<a href="#">Subset1</a> : subsets <code>stc2_coordinates::coords.PhysicalCoordValue.cval</code>

### 4.3.16 DataType **SpatialValue**

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue</code> abstract true
ownedMember	<a href="#">coordFrame</a> <a href="#">epoch</a> <a href="#">Subset</a>
general	<a href="#">MultiCoordValue</a>
specific	<a href="#">SpatialValue1D</a> <a href="#">SpatialValue2D</a> <a href="#">SpatialValue3D</a>
typedElements	DataType <a href="#">CustomLocation</a> Property <a href="#">position</a> <a href="#">velocity</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	The abstract data type <code>SpatialValue</code> contains a reference to a <code>SpatialFrame</code> and to <code>Spatial Space</code> through the coordinate axis. It allows the 1-, 2-, and 3-dimensional coordinate values to be specified as derived data types. It MAY include an epoch. The spatial coordinate value is built from <code>SpatialCoordValue</code> scalar components. Note that <code>SpatialValue</code> may contain either positions or velocities. The latter are restricted to true space velocities; Doppler velocities belong in the redshift domain.

Constraint **SpatialValue::Subset**

owner	<a href="#">SpatialValue</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue::Subset</code> specification subsets <code>stc2_coordinates::coords.BasicCoordValue.coordFrame</code> constrained elements <a href="#">coordFrame</a>

Property **SpatialValue::coordFrame**

owner	<a href="#">SpatialValue</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue::coordFrame</code> type <a href="#">SpaceFrame</a>
constraints	<a href="#">Subset</a> : subsets <code>stc2_coordinates::coords.BasicCoordValue.coordFrame</code>

Property **SpatialValue::epoch**

owner	<a href="#">SpatialValue</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue::epoch</code> multiplicity 0..1 type <a href="#">Epoch</a>

### 4.3.17 DataType SpatialValue1D

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue1D</code> abstract <code>false</code>
ownedMember	<a href="#">cmpt Subset</a>
general	<a href="#">SpatialValue</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	1-dimensional spatial coordinate value.

#### Constraint `SpatialValue1D::Subset`

owner	<a href="#">SpatialValue1D</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue1D::Subset</code> specification <code>subsets stc2_coordinates:coords.MultiCoordValue.cmpt</code> constrained elements <a href="#">cmpt</a>

#### Property `SpatialValue1D::cmpt`

owner	<a href="#">SpatialValue1D</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue1D::cmpt</code> multiplicity <code>1</code> type <a href="#">SpatialCoordValue</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:coords.MultiCoordValue.cmpt</code>

### 4.3.18 DataType SpatialValue2D

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue2D</code> abstract <code>false</code>
ownedMember	<a href="#">cmpt Subset</a>
general	<a href="#">SpatialValue</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	2-dimensional spatial coordinate value. Each component references its own axis.

#### Constraint `SpatialValue2D::Subset`

owner	<a href="#">SpatialValue2D</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue2D::Subset</code> specification <code>subsets stc2_coordinates:coords.MultiCoordValue.cmpt</code> constrained elements <a href="#">cmpt</a>

#### Property `SpatialValue2D::cmpt`

owner	<a href="#">SpatialValue2D</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue2D::cmpt</code> multiplicity <code>2</code> type <a href="#">SpatialCoordValue</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:coords.MultiCoordValue.cmpt</code>

### 4.3.19 DataType SpatialValue3D

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue3D</code> abstract <code>false</code>
ownedMember	<a href="#">cmpt Subset</a>
general	<a href="#">SpatialValue</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	3-dimensional spatial coordinate value. Each component references its own axis.

#### Constraint `SpatialValue3D::Subset`

owner	<a href="#">SpatialValue3D</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue3D::Subset</code> specification <code>subsets stc2_coordinates:coords.MultiCoordValue.cmpt</code> constrained elements <a href="#">cmpt</a>

#### Property `SpatialValue3D::cmpt`

owner	<a href="#">SpatialValue3D</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpatialValue3D::cmpt</code> multiplicity <code>3</code> type <a href="#">SpatialCoordValue</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:coords.MultiCoordValue.cmpt</code>

### 4.3.20 DataType SpaceCoord

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpaceCoord</code> abstract <code>true</code>
ownedMember	<a href="#">coordFrame epoch Subset</a>
general	<a href="#">BasicCoordValue</a>
specific	<a href="#">Cartesian</a> <a href="#">Ecliptic</a> <a href="#">Equatorial</a> <a href="#">Galactic</a> <a href="#">LongLat</a> <a href="#">UnitSphere</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Prototype shortcut spatial coordinate value. If an epoch is provided it will override the epoch of the Spatial Frame (if present).

#### Constraint `SpaceCoord::Subset`

owner	<a href="#">SpaceCoord</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpaceCoord::Subset</code> specification <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code> constrained elements <a href="#">coordFrame</a>

#### Property `SpaceCoord::coordFrame`

owner	<a href="#">SpaceCoord</a>
properties	qualified name <code>stc2_coordinates::domain::space::SpaceCoord::coordFrame</code> unique <code>true</code> type <a href="#">SpaceFrame</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code>



Property **SpaceCoord::epoch**

owner	<a href="#">SpaceCoord</a>								
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::SpaceCoord::epoch</td> </tr> <tr> <td>unique</td> <td>true</td> </tr> <tr> <td>multiplicity</td> <td>0..1</td> </tr> <tr> <td>type</td> <td><a href="#">Epoch</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::space::SpaceCoord::epoch	unique	true	multiplicity	0..1	type	<a href="#">Epoch</a>
qualified name	stc2_coordinates::domain::space::SpaceCoord::epoch								
unique	true								
multiplicity	0..1								
type	<a href="#">Epoch</a>								

## 4.3.21 DataType Cartesian

owner	<a href="#">space</a>				
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::Cartesian</td> </tr> <tr> <td>abstract</td> <td>false</td> </tr> </table>	qualified name	stc2_coordinates::domain::space::Cartesian	abstract	false
qualified name	stc2_coordinates::domain::space::Cartesian				
abstract	false				
ownedMember	<a href="#">x y z</a>				
general	<a href="#">SpaceCoord</a>				
shown on diagram	<a href="#">SpatialDomain</a>				
documentation	Shortcut for Cartesian spatial coordinates.				

Property **Cartesian::x**

owner	<a href="#">Cartesian</a>						
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::Cartesian::x</td> </tr> <tr> <td>multiplicity</td> <td>0..1</td> </tr> <tr> <td>type</td> <td><a href="#">RealQuantity</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::space::Cartesian::x	multiplicity	0..1	type	<a href="#">RealQuantity</a>
qualified name	stc2_coordinates::domain::space::Cartesian::x						
multiplicity	0..1						
type	<a href="#">RealQuantity</a>						

Property **Cartesian::y**

owner	<a href="#">Cartesian</a>						
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::Cartesian::y</td> </tr> <tr> <td>multiplicity</td> <td>0..1</td> </tr> <tr> <td>type</td> <td><a href="#">RealQuantity</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::space::Cartesian::y	multiplicity	0..1	type	<a href="#">RealQuantity</a>
qualified name	stc2_coordinates::domain::space::Cartesian::y						
multiplicity	0..1						
type	<a href="#">RealQuantity</a>						

Property **Cartesian::z**

owner	<a href="#">Cartesian</a>						
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::Cartesian::z</td> </tr> <tr> <td>multiplicity</td> <td>0..1</td> </tr> <tr> <td>type</td> <td><a href="#">RealQuantity</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::space::Cartesian::z	multiplicity	0..1	type	<a href="#">RealQuantity</a>
qualified name	stc2_coordinates::domain::space::Cartesian::z						
multiplicity	0..1						
type	<a href="#">RealQuantity</a>						

## 4.3.22 DataType Ecliptic

owner	<a href="#">space</a>				
properties	<table> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::space::Ecliptic</td> </tr> <tr> <td>abstract</td> <td>false</td> </tr> </table>	qualified name	stc2_coordinates::domain::space::Ecliptic	abstract	false
qualified name	stc2_coordinates::domain::space::Ecliptic				
abstract	false				
ownedMember	<a href="#">coordFrame elat elong r Subset</a>				
general	<a href="#">SpaceCoord</a>				
shown on diagram	<a href="#">SpatialDomain</a>				
documentation	Shortcut for spherical ecliptic spatial coordinates.				

**Constraint [Ecliptic::Subset](#)**

owner	<a href="#">Ecliptic</a>
properties	qualified name <code>stc2_coordinates::domain::space::Ecliptic::Subset</code> specification <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code> constrained elements <a href="#">coordFrame</a>

**Property [Ecliptic::coordFrame](#)**

owner	<a href="#">Ecliptic</a>
properties	qualified name <code>stc2_coordinates::domain::space::Ecliptic::coordFrame</code> type <a href="#">EclipticFrame</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code>

**Property [Ecliptic::elat](#)**

owner	<a href="#">Ecliptic</a>
properties	qualified name <code>stc2_coordinates::domain::space::Ecliptic::elat</code> multiplicity <code>0..1</code> type <a href="#">RealQuantity</a>

**Property [Ecliptic::elong](#)**

owner	<a href="#">Ecliptic</a>
properties	qualified name <code>stc2_coordinates::domain::space::Ecliptic::elong</code> multiplicity <code>0..1</code> type <a href="#">RealQuantity</a>

**Property [Ecliptic::r](#)**

owner	<a href="#">Ecliptic</a>
properties	qualified name <code>stc2_coordinates::domain::space::Ecliptic::r</code> multiplicity <code>0..1</code> type <a href="#">RealQuantity</a>

### 4.3.23 Data Type [Equatorial](#)

owner	<a href="#">space</a>
properties	qualified name <code>stc2_coordinates::domain::space::Equatorial</code> abstract <code>false</code>
ownedMember	<a href="#">coordFrame</a> <a href="#">dec r ra</a> <a href="#">Subset</a>
general	<a href="#">SpaceCoord</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Shortcut for spherical equatorial spatial coordinates.

**Constraint [Equatorial::Subset](#)**

owner	<a href="#">Equatorial</a>
properties	qualified name <code>stc2_coordinates::domain::space::Equatorial::Subset</code> specification <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code> constrained elements <a href="#">coordFrame</a>

**Property [Equatorial::coordFrame](#)**

owner	<a href="#">Equatorial</a>
properties	qualified name <code>stc2_coordinates::domain::space::Equatorial::coordFrame</code> visibility <code>public</code>

	type <a href="#">EquatorialFrame</a>
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coords.BasicCoordValue.coordFrame

Property **Equatorial::ra**

owner	<a href="#">Equatorial</a>
properties	qualified name stc2_coordinates::domain::space::Equatorial::ra multiplicity 0..1 type <a href="#">RealQuantity</a>

Property **Equatorial::dec**

owner	<a href="#">Equatorial</a>
properties	qualified name stc2_coordinates::domain::space::Equatorial::dec multiplicity 0..1 type <a href="#">RealQuantity</a>

Property **Equatorial::r**

owner	<a href="#">Equatorial</a>
properties	qualified name stc2_coordinates::domain::space::Equatorial::r multiplicity 0..1 type <a href="#">RealQuantity</a>

### 4.3.24 DataType Galactic

owner	<a href="#">space</a>
properties	qualified name stc2_coordinates::domain::space::Galactic abstract false
ownedMember	<a href="#">b coordFrame</a>   <a href="#">r Subset</a>
general	<a href="#">SpaceCoord</a>
shown on diagram	<a href="#">SpatialDomain</a>
documentation	Shortcut for spherical Galactic spatial coordinates.

Constraint **Galactic::Subset**

owner	<a href="#">Galactic</a>
properties	qualified name stc2_coordinates::domain::space::Galactic::Subset specification subsets stc2_coordinates::coords.BasicCoordValue.coordFrame constrained elements <a href="#">coordFrame</a>

Property **Galactic::coordFrame**

owner	<a href="#">Galactic</a>
properties	qualified name stc2_coordinates::domain::space::Galactic::coordFrame type <a href="#">GalacticFrame</a>
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coords.BasicCoordValue.coordFrame

Property **Galactic::l**

owner	<a href="#">Galactic</a>
properties	qualified name stc2_coordinates::domain::space::Galactic::l unique true multiplicity 0..1 type <a href="#">RealQuantity</a>

Property **Galactic::b**

owner	<a href="#">Galactic</a>	
properties	qualified name	stc2_coordinates::domain::space::Galactic::b
	multiplicity	0..1
	type	<a href="#">RealQuantity</a>

Property **Galactic::r**

owner	<a href="#">Galactic</a>	
properties	qualified name	stc2_coordinates::domain::space::Galactic::r
	multiplicity	0..1
	type	<a href="#">RealQuantity</a>

### 4.3.25 DataType LongLat

owner	<a href="#">space</a>	
properties	qualified name	stc2_coordinates::domain::space::LongLat
	abstract	false
ownedMember	<a href="#">lat long r</a>	
general	<a href="#">SpaceCoord</a>	
shown on diagram	<a href="#">SpatialDomain</a>	
documentation	Shortcut for (generic) spherical spatial coordinates other than Equatorial, Galactic, and Ecliptic.	

Property **LongLat::long**

owner	<a href="#">LongLat</a>	
properties	qualified name	stc2_coordinates::domain::space::LongLat::long
	multiplicity	0..1
	type	<a href="#">RealQuantity</a>

Property **LongLat::lat**

owner	<a href="#">LongLat</a>	
properties	qualified name	stc2_coordinates::domain::space::LongLat::lat
	multiplicity	0..1
	type	<a href="#">RealQuantity</a>
documentation	Shortcut for spherical spatial coordinates.	

Property **LongLat::r**

owner	<a href="#">LongLat</a>	
properties	qualified name	stc2_coordinates::domain::space::LongLat::r
	multiplicity	0..1
	type	<a href="#">RealQuantity</a>

### 4.3.26 DataType UnitSphere

owner	<a href="#">space</a>	
properties	qualified name	stc2_coordinates::domain::space::UnitSphere
	abstract	false
ownedMember	<a href="#">dircosx dircosy dircosz</a>	
general	<a href="#">SpaceCoord</a>	

shown on diagram	<a href="#">SpatialDomain</a>
documentation	Shortcut for unit sphere spatial coordinates (direction cosines).

Property **UnitSphere::dircosx**

owner	<a href="#">UnitSphere</a>
properties	qualified name stc2_coordinates::domain::space::UnitSphere::dircosx multiplicity 0..1 type real

Property **UnitSphere::dircosy**

owner	<a href="#">UnitSphere</a>
properties	qualified name stc2_coordinates::domain::space::UnitSphere::dircosy multiplicity 0..1 type real

Property **UnitSphere::dircosz**

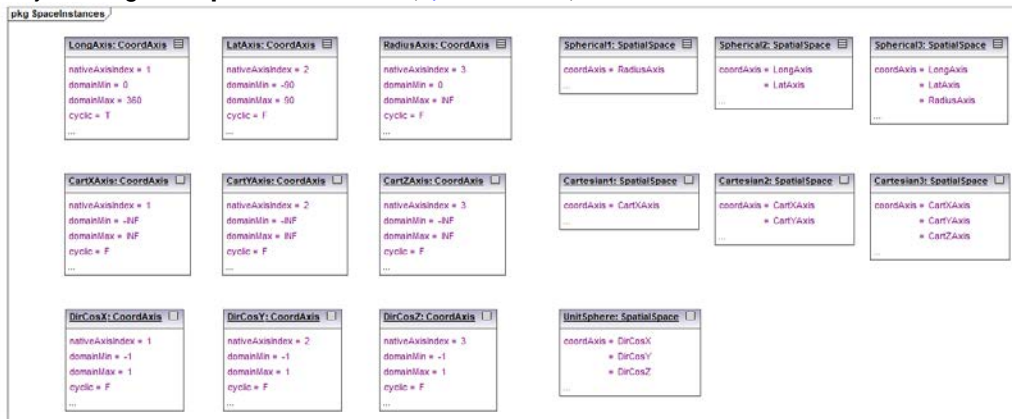
owner	<a href="#">UnitSphere</a>
properties	qualified name stc2_coordinates::domain::space::UnitSphere::dircosz multiplicity 0..1 type real

## 4.4 Package Space Instances

Package **SpaceInstances**

owner	<a href="#">stc2_coordinates</a>
properties	qualified name stc2_coordinates::SpaceInstances visibility public «modelement» false «modelimport» false
ownedDiagrams	<a href="#">SpatialShortcuts</a>
ownedMember	<b>Cartesian1 Cartesian2 Cartesian3 CartXAxis CartYAxis CartZAxis DirCosX DirCosY DirCosZ LatAxis LongAxis RadiusAxis Spherical1 Spherical2 Spherical3 UnitSphere</b>

Object Diagram **SpatialShortcuts** ([SpaceInstances](#))



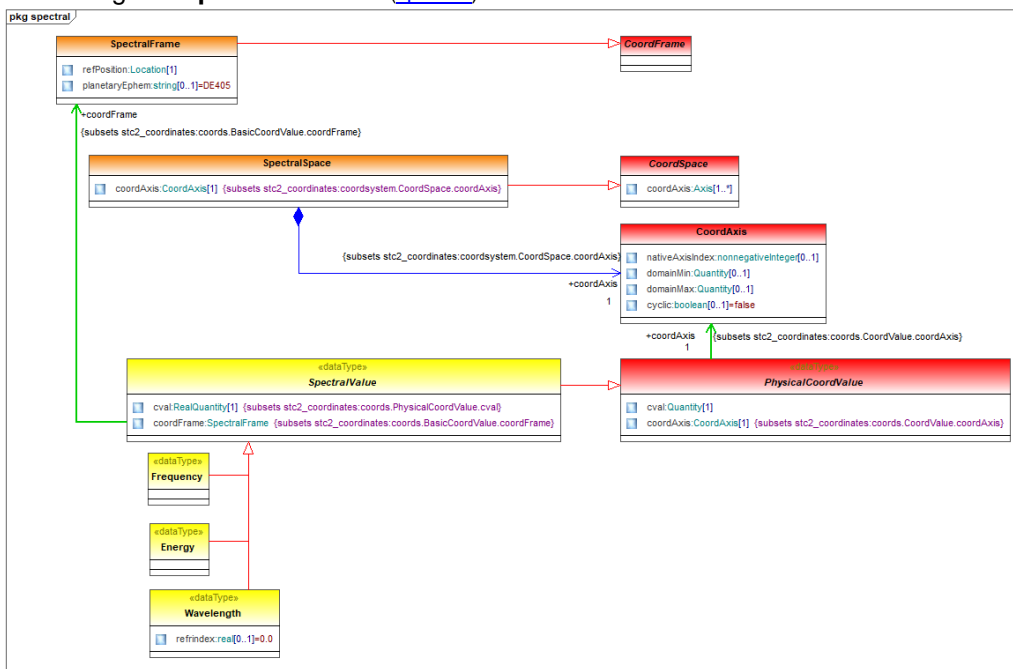
### 4.5 Spectral Domain

This domain's coordinates may be energy, frequency, or wavelength which are the physical properties of the photons – not redshift or Doppler velocity.

Package **spectral**

owner	<a href="#">domain</a>
properties	qualified name stc2_coordinates::domain::spectral
ownedDiagrams	<a href="#">SpectralDomain</a>
ownedMember	<a href="#">Energy</a> <a href="#">Frequency</a> <a href="#">SpectralFrame</a> <a href="#">SpectralSpace</a> <a href="#">SpectralValue</a> <a href="#">Wavelength</a>

Class Diagram **SpectralDomain** ([spectral](#))



#### 4.5.1 Class SpectralFrame

diagram	
owner	<a href="#">spectral</a>
properties	qualified name stc2_coordinates::domain::spectral::SpectralFrame abstract false
ownedMember	<a href="#">planetaryEphem</a> <a href="#">refPosition</a> <a href="#">Subset</a>
general	<a href="#">CoordFrame</a>
typedElements	Class <a href="#">AstroCoordSystem</a> Property <a href="#">spectralFrame</a> Data Type <a href="#">SpectralValue</a> Property <a href="#">coordFrame</a>

shown on diagram	<a href="#">CoordSystems SpectralDomain</a>
documentation	The SpectralFrame is the Frame for spectral coordinates. It requires a Reference Position (in the AstroCoordSystem and which need to be specified in phase space) and SHALL contain a Spectral Space object.

Property **SpectralFrame::planetaryEphem**

owner	<a href="#">SpectralFrame</a>
properties	qualified name stc2_coordinates::domain::spectral::SpectralFrame::planetaryEphem multiplicity 0..1 default DE405

Property **SpectralFrame::refPosition**

owner	<a href="#">SpectralFrame</a>
properties	qualified name stc2_coordinates::domain::spectral::SpectralFrame::refPosition multiplicity 1 type <a href="#">Location</a>

### 4.5.2 Class SpectralSpace

diagram	
owner	<a href="#">spectral</a>
properties	qualified name stc2_coordinates::domain::spectral::SpectralSpace abstract false
ownedMember	<a href="#">coordAxis Subset</a>
general	<a href="#">CoordSpace</a>
shown on diagram	<a href="#">SpectralDomain</a>
documentation	The SpectralSpace SHALL include one CoordAxis object, which will typically not include any of its optional attributes other than Name, although domainMin could trivially be set to zero, of course.

Constraint **SpectralSpace::Subset**

owner	<a href="#">SpectralSpace</a>
properties	qualified name stc2_coordinates::domain::spectral::SpectralSpace::Subset specification subsets stc2_coordinates::coordsystem.CoordSpace.coordAxis constrained elements <a href="#">coordAxis</a>

Property **SpectralSpace::coordAxis**

owner	<a href="#">SpectralSpace</a>
properties	qualified name stc2_coordinates::domain::spectral::SpectralSpace::coordAxis multiplicity 1 type <a href="#">CoordAxis</a> aggregation composite
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coordsystem.CoordSpace.coordAxis

### 4.5.3 DataType SpectralValue

owner	<a href="#">spectral</a>
-------	--------------------------

## STC2: I Coordinate Systems, Frames, and Values

properties	qualified name abstract	stc2_coordinates::domain::spectral::SpectralValue true
ownedMember	<a href="#">coordFrame</a> <a href="#">cval</a> <a href="#">Subset1</a> <a href="#">Subset2</a>	
general	<a href="#">PhysicalCoordValue</a>	
specific	<a href="#">Energy</a> <a href="#">Frequency</a> <a href="#">Wavelength</a>	
shown on diagram	<a href="#">SpectralDomain</a>	
documentation	The abstract data type SpectralValue contains a reference to a SpectralFrame and to a Spectral Space through the coordinate axis. It allows the equivalent data types Frequency, Energy, and Wavelength coordinate values to be specified as derived data types.	

### Constraint **SpectralValue::Subset1**

owner	<a href="#">SpectralValue</a>	
properties	qualified name specification constrained elements	stc2_coordinates::domain::spectral::SpectralValue::Subset1 subsets stc2_coordinates::coords::PhysicalCoordValue.cval <a href="#">cval</a>

### Constraint **SpectralValue::Subset2**

owner	<a href="#">SpectralValue</a>	
properties	qualified name specification constrained elements	stc2_coordinates::domain::spectral::SpectralValue::Subset2 subsets stc2_coordinates::coords::BasicCoordValue.coordFrame <a href="#">coordFrame</a>

### Property **SpectralValue::coordFrame**

owner	<a href="#">SpectralValue</a>	
properties	qualified name type	stc2_coordinates::domain::spectral::SpectralValue::coordFrame <a href="#">SpectralFrame</a>
constraints	<a href="#">Subset2</a> : subsets stc2_coordinates::coords::BasicCoordValue.coordFrame	

### Property **SpectralValue::cval**

owner	<a href="#">SpectralValue</a>	
properties	qualified name multiplicity type	stc2_coordinates::domain::spectral::SpectralValue::cval 1 <a href="#">RealQuantity</a>
constraints	<a href="#">Subset1</a> : subsets stc2_coordinates::coords::PhysicalCoordValue.cval	

## 4.5.4 DataType Energy

owner	<a href="#">spectral</a>	
properties	qualified name abstract	stc2_coordinates::domain::spectral::Energy false
general	<a href="#">SpectralValue</a>	
shown on diagram	<a href="#">SpectralDomain</a>	
documentation	Spectral coordinate value expressed as an energy.	

## 4.5.5 DataType Frequency

owner	<a href="#">spectral</a>	
-------	--------------------------	--



properties	qualified name abstract	stc2_coordinates::domain::spectral::Frequency false
general	<a href="#">SpectralValue</a>	
shown on diagram	<a href="#">SpectralDomain</a>	
documentation	Spectral coordinate value expressed as a frequency.	

### 4.5.6 DataType Wavelength

owner	<a href="#">spectral</a>	
properties	qualified name abstract	stc2_coordinates::domain::spectral::Wavelength false
ownedMember	<a href="#">refrindex</a>	
general	<a href="#">SpectralValue</a>	
shown on diagram	<a href="#">SpectralDomain</a>	
documentation	Spectral coordinate value expressed as a wavelength. Note that a refractive index is required for proper interpretation of wavelength; if absent, a value of 0.0 (vacuum) will be assumed.	

#### Property Wavelength::refrindex

owner	<a href="#">Wavelength</a>	
properties	qualified name multiplicity type default	stc2_coordinates::domain::spectral::Wavelength::refrindex 0..1 real 0.0
documentation	Refractive index of the medium in which the wavelength is measured; default=0.0.	

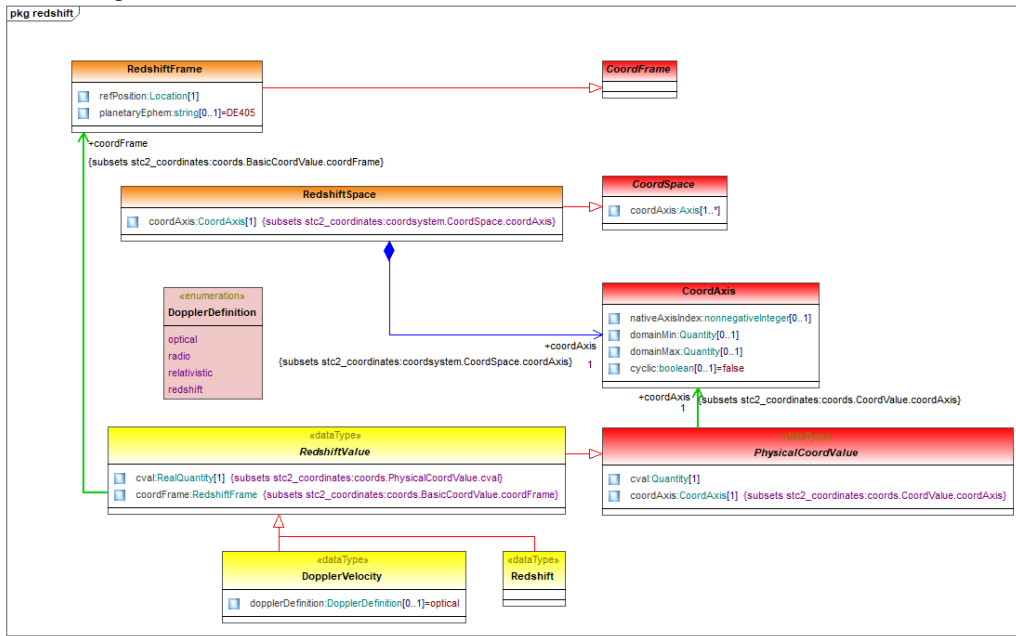
## 4.6 Redshift Domain

The Redshift domain accommodates redshift and Doppler velocity coordinates, i.e., the properties derived from spectral line analysis in conjunction with a line rest frequency.

#### Package redshift

owner	<a href="#">domain</a>	
properties	qualified name	stc2_coordinates::domain::redshift
ownedDiagrams	<a href="#">RedshiftDomain</a>	
ownedMember	<a href="#">DopplerDefinition</a> <a href="#">DopplerVelocity</a> <a href="#">Redshift</a> <a href="#">RedshiftFrame</a> <a href="#">RedshiftSpace</a> <a href="#">RedshiftValue</a>	

Class Diagram **RedshiftDomain** ([redshift](#))



A RedshiftCoord contains one of four possible representations of redshift or Doppler velocity, identified by its name.

4.6.1 Class RedshiftFrame

diagram	
owner	<a href="#">redshift</a>
properties	qualified name <code>stc2_coordinates::domain::redshift::RedshiftFrame</code> abstract <code>false</code>
ownedMember	<a href="#">planetaryEphem</a> <a href="#">refPosition</a> <a href="#">Subset</a>
general	<a href="#">CoordFrame</a>
typedElements	Class <a href="#">AstroCoordSystem</a> Property <a href="#">redshiftFrame</a> DataType <a href="#">RedshiftValue</a> Property <a href="#">coordFrame</a>
shown on diagram	<a href="#">CoordSystems</a> <a href="#">RedshiftDomain</a>
documentation	The RedshiftFrame is the Frame for redshift and Doppler velocity coordinates. It requires a Reference Position in the associated AstroCoordSystem object (which need to be specified in phase space) and SHALL include a RedshiftSpace.

Constraint **RedshiftFrame::Subset**

owner	<a href="#">RedshiftFrame</a>
properties	qualified name <code>stc2_coordinates::domain::redshift::RedshiftFrame::Subset</code> specification <code>subsets stc2_coordinates:coordsystem.CoordFrame.coordSpace</code>

Property **RedshiftFrame::planetaryEphem**

owner	<a href="#">RedshiftFrame</a>
-------	-------------------------------

properties	qualified name	stc2_coordinates::domain::redshift::RedshiftFrame::planetaryEphem
	multiplicity	0..1
	type	string
	default	DE405

Property **RedshiftFrame::refPosition**

owner	<a href="#">RedshiftFrame</a>	
properties	qualified name	stc2_coordinates::domain::redshift::RedshiftFrame::refPosition
	multiplicity	1
	type	<a href="#">Location</a>

### 4.6.2 Class RedshiftSpace

diagram		
owner	<a href="#">redshift</a>	
properties	qualified name	stc2_coordinates::domain::redshift::RedshiftSpace
	abstract	false
ownedMember	<a href="#">coordAxis Subset</a>	
general	<a href="#">CoordSpace</a>	
shown on diagram	<a href="#">RedshiftDomain</a>	
documentation	The RedshiftSpace SHALL include one CoordAxis object, which will typically not include any of its optional attributes other than Name.	

Constraint **RedshiftSpace::Subset**

owner	<a href="#">RedshiftSpace</a>	
properties	qualified name	stc2_coordinates::domain::redshift::RedshiftSpace::Subset
	specification	subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis
	constrained elements	<a href="#">coordAxis</a>

Property **RedshiftSpace::coordAxis**

owner	<a href="#">RedshiftSpace</a>	
properties	qualified name	stc2_coordinates::domain::redshift::RedshiftSpace::coordAxis
	multiplicity	1
	type	<a href="#">CoordAxis</a>
	aggregation	composite
constraints	<a href="#">Subset:</a>	subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis

### 4.6.3 DataType RedshiftValue

owner	<a href="#">redshift</a>	
properties	qualified name	stc2_coordinates::domain::redshift::RedshiftValue
	abstract	true
ownedMember	<a href="#">coordFrame cval Subset1 Subset2</a>	
general	<a href="#">PhysicalCoordValue</a>	
specific	<a href="#">DopplerVelocity Redshift</a>	
shown on	<a href="#">RedshiftDomain</a>	

diagram	
documentation	The abstract data type RedshiftValue contains a reference to a RedshiftFrame and to a Spectral Space object through the coordinate axis. It allows the equivalent data types Redshift and DopplerVelocity coordinate values to be specified as derived data types.

**Constraint RedshiftValue::Subset1**

owner	<a href="#">RedshiftValue</a>						
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::redshift::RedshiftValue::Subset1</td> </tr> <tr> <td>specification</td> <td>subsets stc2_coordinates:coords.PhysicalCoordValue.cval</td> </tr> <tr> <td>constrained elements</td> <td><a href="#">cval</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::redshift::RedshiftValue::Subset1	specification	subsets stc2_coordinates:coords.PhysicalCoordValue.cval	constrained elements	<a href="#">cval</a>
qualified name	stc2_coordinates::domain::redshift::RedshiftValue::Subset1						
specification	subsets stc2_coordinates:coords.PhysicalCoordValue.cval						
constrained elements	<a href="#">cval</a>						

**Constraint RedshiftValue::Subset2**

owner	<a href="#">RedshiftValue</a>						
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::redshift::RedshiftValue::Subset2</td> </tr> <tr> <td>specification</td> <td>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</td> </tr> <tr> <td>constrained elements</td> <td><a href="#">coordFrame</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::redshift::RedshiftValue::Subset2	specification	subsets stc2_coordinates:coords.BasicCoordValue.coordFrame	constrained elements	<a href="#">coordFrame</a>
qualified name	stc2_coordinates::domain::redshift::RedshiftValue::Subset2						
specification	subsets stc2_coordinates:coords.BasicCoordValue.coordFrame						
constrained elements	<a href="#">coordFrame</a>						

**Property RedshiftValue::coordFrame**

owner	<a href="#">RedshiftValue</a>				
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::redshift::RedshiftValue::coordFrame</td> </tr> <tr> <td>type</td> <td><a href="#">RedshiftFrame</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::redshift::RedshiftValue::coordFrame	type	<a href="#">RedshiftFrame</a>
qualified name	stc2_coordinates::domain::redshift::RedshiftValue::coordFrame				
type	<a href="#">RedshiftFrame</a>				
constraints	<a href="#">Subset2</a> : subsets stc2_coordinates:coords.BasicCoordValue.coordFrame				

**Property RedshiftValue::cval**

owner	<a href="#">RedshiftValue</a>						
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::redshift::RedshiftValue::cval</td> </tr> <tr> <td>multiplicity</td> <td>1</td> </tr> <tr> <td>type</td> <td><a href="#">RealQuantity</a></td> </tr> </table>	qualified name	stc2_coordinates::domain::redshift::RedshiftValue::cval	multiplicity	1	type	<a href="#">RealQuantity</a>
qualified name	stc2_coordinates::domain::redshift::RedshiftValue::cval						
multiplicity	1						
type	<a href="#">RealQuantity</a>						
constraints	<a href="#">Subset1</a> : subsets stc2_coordinates:coords.PhysicalCoordValue.cval						

**4.6.4 DataType Redshift**

owner	<a href="#">redshift</a>				
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::redshift::Redshift</td> </tr> <tr> <td>abstract</td> <td>false</td> </tr> </table>	qualified name	stc2_coordinates::domain::redshift::Redshift	abstract	false
qualified name	stc2_coordinates::domain::redshift::Redshift				
abstract	false				
general	<a href="#">RedshiftValue</a>				
shown on diagram	<a href="#">RedshiftDomain</a>				
documentation	The RedshiftValue expressed as a unitless redshift (z)				

**4.6.5 DataType DopplerVelocity**

owner	<a href="#">redshift</a>				
properties	<table border="0"> <tr> <td>qualified name</td> <td>stc2_coordinates::domain::redshift::DopplerVelocity</td> </tr> <tr> <td>abstract</td> <td>false</td> </tr> </table>	qualified name	stc2_coordinates::domain::redshift::DopplerVelocity	abstract	false
qualified name	stc2_coordinates::domain::redshift::DopplerVelocity				
abstract	false				
ownedMember	<a href="#">dopplerDefinition</a>				
general	<a href="#">RedshiftValue</a>				
shown on diagram	<a href="#">RedshiftDomain</a>				

documentation	The RedshiftValue expressed as a DopplerVelocity, requiring a DopplerDefinition (default: optical).
---------------	---

Property **DopplerVelocity::dopplerDefinition**

owner	<a href="#">DopplerVelocity</a>
properties	qualified name <code>stc2_coordinates::domain::redshift::DopplerVelocity::dopplerDefinition</code> multiplicity <code>0..1</code> type <a href="#">DopplerDefinition</a> default <code>optical</code>

### 4.6.6 Enumeration DopplerDefinition

diagram	
owner	<a href="#">redshift</a>
properties	qualified name <code>stc2_coordinates::domain::redshift::DopplerDefinition</code> abstract <code>false</code>
ownedMember	<b>optical radio redshift relativistic</b>
typedElements	DataType <a href="#">DopplerVelocity</a> Property <a href="#">dopplerDefinition</a>
shown on diagram	<a href="#">RedshiftDomain</a>
documentation	Doppler/redshift definition

This parameter specifies what the definition of redshift is and how it should be translated to Doppler velocity. Allowed values are `optical`, `radio`, `relativistic`, and `redshift`.

optical: 
$$V_{opt} = c \cdot \frac{\Delta\lambda}{\lambda_0} = -c \cdot \frac{\Delta v}{v}$$

radio: 
$$V_{rad} = -c \cdot \frac{\Delta v}{v_0} = c \cdot \frac{\Delta\lambda}{\lambda}$$

relativistic: 
$$V_{rel} = c \cdot \frac{\lambda^2 - \lambda_0^2}{\lambda^2 + \lambda_0^2} = -c \cdot \frac{v^2 - v_0^2}{v^2 + v_0^2}$$

redshift: 
$$z = \frac{\Delta\lambda}{\lambda_0} = \frac{\Delta v}{v}$$

It should be emphasized that Doppler velocities are formal velocities; i.e., defined by a formalism and not necessarily physical in nature. Also note here that the label `relativistic` is not strictly correct.

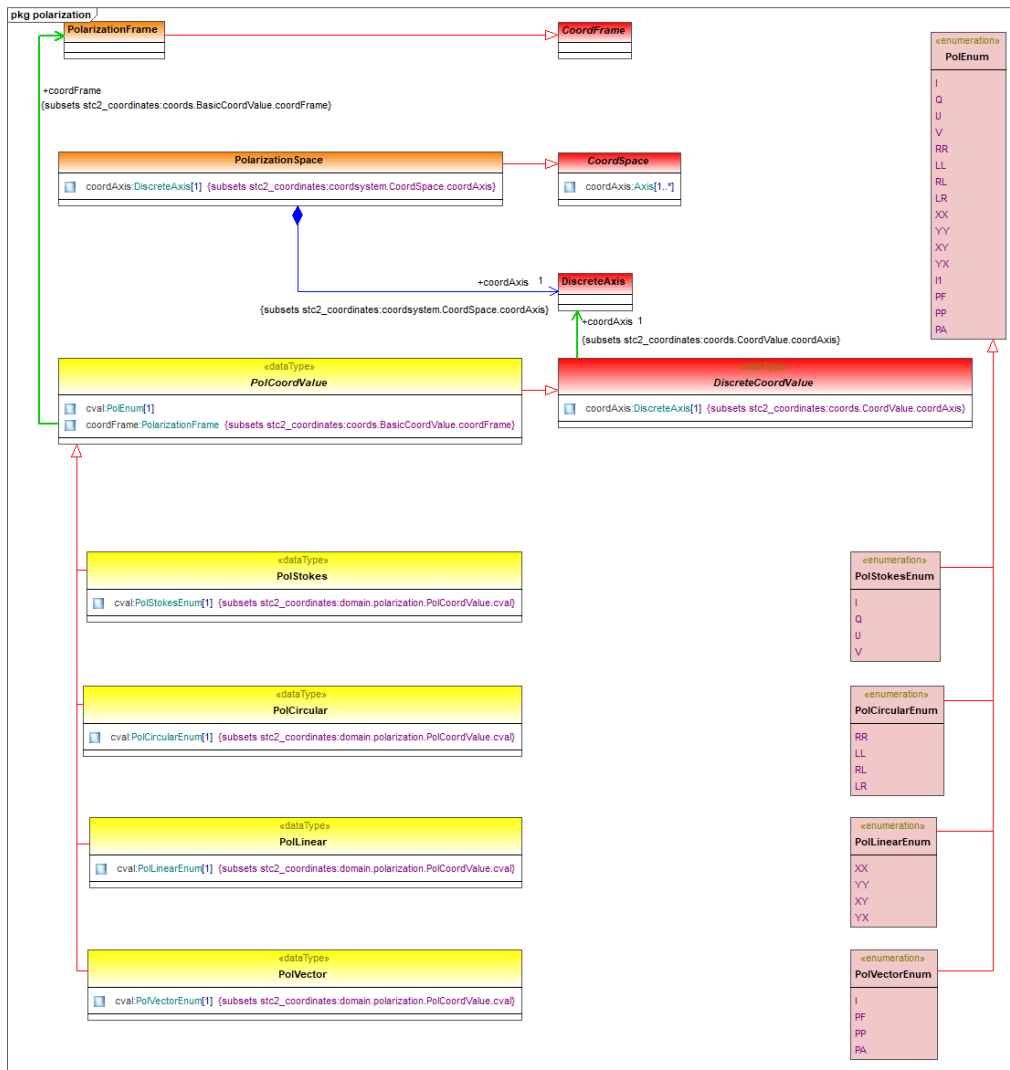
### 4.7 Polarization Domain

The Polarization domain intrinsically contains enumerated coordinate values, which may be expressed in four different ways (flavors): Stokes parameters, circular polarization, linear polarization, or polarization vectors.


Package **polarization**

owner	<a href="#">domain</a>
properties	qualified name <code>stc2_coordinates::domain::polarization</code>
ownedDiagrams	<a href="#">PolarizationDomain</a>
ownedMember	<a href="#">PolarizationFrame</a> <a href="#">PolarizationSpace</a> <a href="#">PolCircular</a> <a href="#">PolCircularEnum</a> <a href="#">PolCoordValue</a> <a href="#">PolEnum</a> <a href="#">PolLinear</a> <a href="#">PolLinearEnum</a> <a href="#">PolStokes</a> <a href="#">PolStokesEnum</a> <a href="#">PolVector</a> <a href="#">PolVectorEnum</a>


Class Diagram **PolarizationDomain** ([polarization](#))



### 4.7.1 Class PolarizationFrame

diagram	
owner	<a href="#">polarization</a>
properties	qualified name stc2_coordinates::domain::polarization::PolarizationFrame abstract false
ownedMember	<a href="#">Subset</a>
general	<a href="#">CoordFrame</a>
typedElements	Class <a href="#">AstroCoordSystem</a> Property <a href="#">polarizationFrame</a> DataType <a href="#">PolCoordValue</a> Property <a href="#">coordFrame</a>
shown on diagram	<a href="#">CoordSystems</a> <a href="#">PolarizationDomain</a>
documentation	A Polarization Frame SHALL contain one PolarizationSpace object. It may need a reference to a spatial frame (TBD).

### 4.7.2 Class PolarizationSpace

diagram	
owner	<a href="#">polarization</a>
properties	qualified name stc2_coordinates::domain::polarization::PolarizationSpace abstract false
ownedMember	<a href="#">coordAxis</a> <a href="#">Subset</a>
general	<a href="#">CoordSpace</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	The PolarizationSpace SHALL include one DiscreteAxis object, since polarization is an enumerated coordinate.

#### Constraint PolarizationSpace::Subset

owner	<a href="#">PolarizationSpace</a>
properties	qualified name stc2_coordinates::domain::polarization::PolarizationSpace::Subset specification subsets stc2_coordinates::coordsystem::CoordSpace.coordAxis constrained elements <a href="#">coordAxis</a>

#### Property PolarizationSpace::coordAxis

owner	<a href="#">PolarizationSpace</a>
properties	qualified name stc2_coordinates::domain::polarization::PolarizationSpace::coordAxis multiplicity 1 type <a href="#">DiscreteAxis</a> aggregation composite
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coordsystem::CoordSpace.coordAxis

### 4.7.3 DataType PolCoordValue

owner	<a href="#">polarization</a>
properties	qualified name stc2_coordinates::domain::polarization::PolCoordValue abstract true

ownedMember	<a href="#">coordFrame</a> <a href="#">cval</a> <a href="#">Subset</a>
general	<a href="#">DiscreteCoordValue</a>
specific	<a href="#">PolCircular</a> <a href="#">PolLinear</a> <a href="#">PolStokes</a> <a href="#">PolVector</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	The abstract data type PolCoordValue contains a reference to a PolarizationFrame and to a PolarizationSpace object through a DiscreteAxis, It allows the equivalent data types PolStokes, PolCircular, PolLinear, and PolVector coordinate values to be specified as derived data types.

**Constraint PolCoordValue::Subset**

owner	<a href="#">PolCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolCoordValue::Subset</code> specification <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code> constrained elements <a href="#">coordFrame</a>

**Property PolCoordValue::coordFrame**

owner	<a href="#">PolCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolCoordValue::coordFrame</code> type <a href="#">PolarizationFrame</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:coords.BasicCoordValue.coordFrame</code>

**Property PolCoordValue::cval**

owner	<a href="#">PolCoordValue</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolCoordValue::cval</code> multiplicity 1 type <a href="#">PolEnum</a>

### 4.7.4 Enumeration PolEnum

diagram	<pre> classDiagram     class PolEnum {         I         Q         U         V         RR         LL         RL         LR         XX         YY         XY         YX         II         PF         PP         PA     }                 </pre>
owner	<a href="#">polarization</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolEnum</code> abstract false



ownedMember	I I1 LL LR PA PF PP Q RL RR U V XX XY YX YY
specific	<a href="#">PolCircularEnum</a> <a href="#">PolLinearEnum</a> <a href="#">PolStokesEnum</a> <a href="#">PolVectorEnum</a>
typedElements	DataType <a href="#">PolCoordValue</a> Property <a href="#">cval</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	PolEnum contains all polarization coordinate values. The values for each of the four types of polarization coordinates are contained in separate derived (by restriction) enumeration types.

### 4.7.5 DataType PolStokes

owner	<a href="#">polarization</a>
properties	qualified name stc2_coordinates::domain::polarization::PolStokes abstract false
ownedMember	<a href="#">cval</a> <a href="#">Subset</a>
general	<a href="#">PolCoordValue</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	Stokes polarization coordinate value.

#### Constraint PolStokes::Subset

owner	<a href="#">PolStokes</a>
properties	qualified name stc2_coordinates::domain::polarization::PolStokes::Subset specification subsets stc2_coordinates:domain.polarization.PolCoordValue.cval constrained elements <a href="#">cval</a>

#### Property PolStokes::cval

owner	<a href="#">PolStokes</a>
properties	qualified name stc2_coordinates::domain::polarization::PolStokes::cval multiplicity 1 type <a href="#">PolStokesEnum</a>
constraints	<a href="#">Subset</a> : subsets stc2_coordinates:domain.polarization.PolCoordValue.cval

#### Enumeration PolStokesEnum

diagram	<pre> classDiagram     class PolStokesEnum {         I         Q         U         V     }             </pre>
owner	<a href="#">polarization</a>
properties	qualified name stc2_coordinates::domain::polarization::PolStokesEnum abstract false
ownedMember	I Q U V
general	<a href="#">PolEnum</a>
typedElements	DataType <a href="#">PolStokes</a> Property <a href="#">cval</a>
shown on	<a href="#">PolarizationDomain</a>

diagram	
documentation	The four Stokes coordinate values for a polarization coordinate.

### 4.7.6 DataType PolCircular

owner	<a href="#">polarization</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolCircular</code> abstract <code>false</code>
ownedMember	<a href="#">cval Subset</a>
general	<a href="#">PolCoordValue</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	Circular polarization coordinate value.

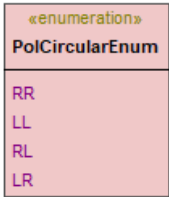
#### Constraint PolCircular::Subset

owner	<a href="#">PolCircular</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolCircular::Subset</code> specification <code>subsets stc2_coordinates:domain.polarization.PolCoordValue.cval</code> constrained elements <a href="#">cval</a>

#### Property PolCircular::cval

owner	<a href="#">PolCircular</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolCircular::cval</code> multiplicity <code>1</code> type <a href="#">PolCircularEnum</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:domain.polarization.PolCoordValue.cval</code>

#### Enumeration PolCircularEnum

diagram	 <pre> classDiagram     class PolCircularEnum {         RR         LL         RL         LR     }         </pre>
owner	<a href="#">polarization</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolCircularEnum</code> abstract <code>false</code>
ownedMember	<b>LL LR RL RR</b>
general	<a href="#">PolEnum</a>
typedElements	DataType <a href="#">PolCircular</a> Property <a href="#">cval</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	The four circular polarization coordinate values for a polarization coordinate.

### 4.7.7 DataType PolLinear

owner	<a href="#">polarization</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolLinear</code> abstract <code>false</code>
ownedMember	<a href="#">cval Subset</a>
general	<a href="#">PolCoordValue</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	Linear polarization coordinate value.

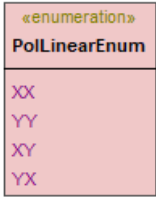
#### Constraint PolLinear::Subset

owner	<a href="#">PolLinear</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolLinear::Subset</code> specification <code>subsets stc2_coordinates:domain.polarization.PolCoordValue.cval</code> constrained elements <a href="#">cval</a>

#### Property PolLinear::cval

owner	<a href="#">PolLinear</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolLinear::cval</code> multiplicity <code>1</code> type <a href="#">PolLinearEnum</a>
constraints	<a href="#">Subset</a> : <code>subsets stc2_coordinates:domain.polarization.PolCoordValue.cval</code>

#### Enumeration PolLinearEnum

diagram	 <pre> classDiagram     class PolLinearEnum {         XX         YY         XY         YX     }   </pre>
owner	<a href="#">polarization</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolLinearEnum</code> abstract <code>false</code>
ownedMember	<b>XX XY YX YY</b>
general	<a href="#">PolEnum</a>
typedElements	DataType <a href="#">PolLinear</a> Property <a href="#">cval</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	The four linear polarization coordinate values for a polarization coordinate.

### 4.7.8 DataType PolVector

owner	<a href="#">polarization</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolVector</code> abstract <code>false</code>
ownedMember	<a href="#">cval Subset</a>

general	<a href="#">PolCoordValue</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	Vector polarization coordinate value.

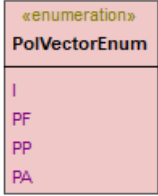
**Constraint PolVector::Subset**

owner	<a href="#">PolVector</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolVector::Subset</code> specification <code>subsets stc2_coordinates:domain.polarization.PolCoordValue.cval</code> constrained elements <a href="#">cval</a>

**Property PolVector::cval**

owner	<a href="#">PolVector</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolVector::cval</code> multiplicity 1 type <a href="#">PolVectorEnum</a>
constraints	<a href="#">Subset</a> : subsets <code>stc2_coordinates:domain.polarization.PolCoordValue.cval</code>

**Enumeration PolVectorEnum**

diagram	
owner	<a href="#">polarization</a>
properties	qualified name <code>stc2_coordinates::domain::polarization::PolVectorEnum</code> abstract false
ownedMember	<b>I PA PF PP</b>
general	<a href="#">PolEnum</a>
typedElements	DataType <a href="#">PolVector</a> Property <a href="#">cval</a>
shown on diagram	<a href="#">PolarizationDomain</a>
documentation	The four polarization vector coordinate values for a polarization coordinate.

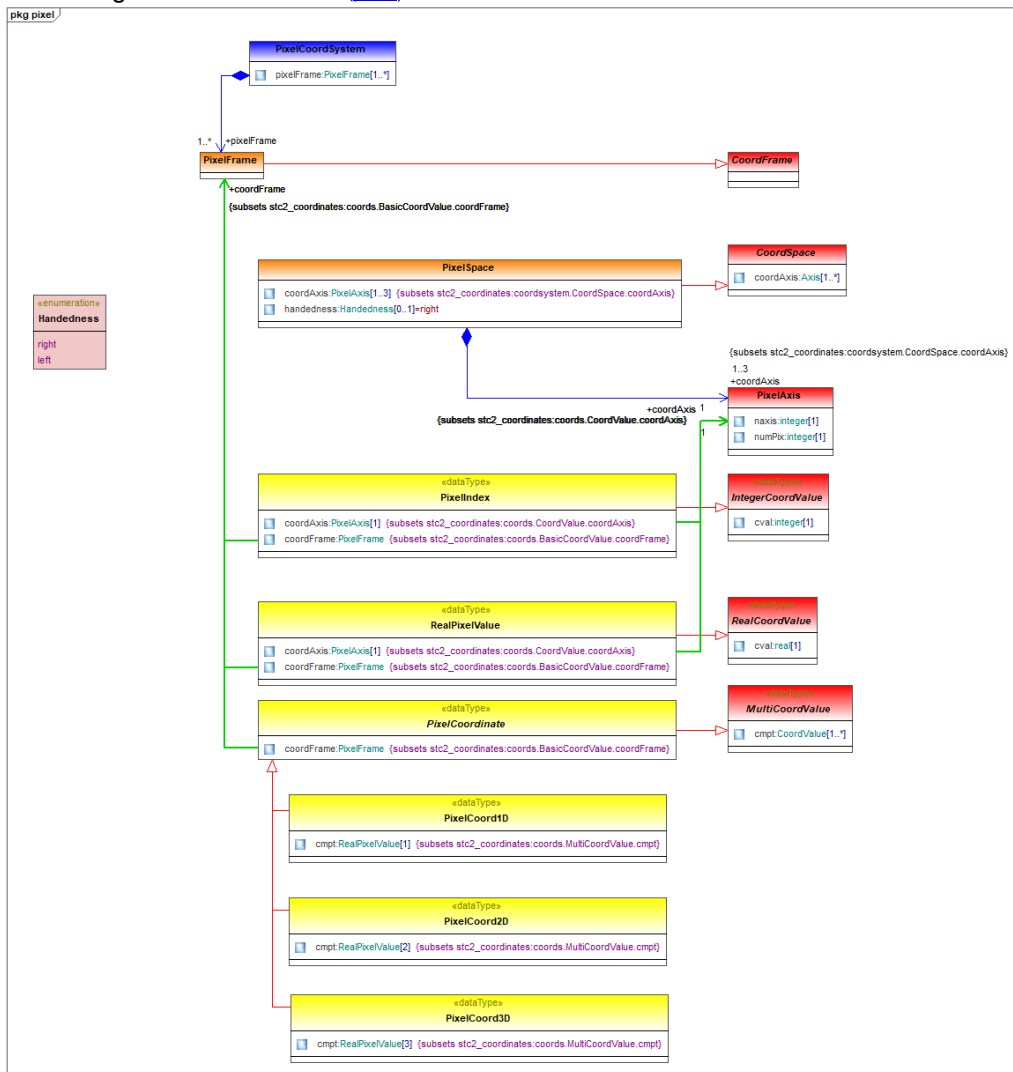
## 4.8 Pixel Domain

In a Pixel Space values of a dependent variable are indexed on a regular 1-, 2-, or 3-dimensional grid. Mathematically, it can be described as a continuous 1-, 2-, or 3-dimensional function, usually convolved with a point spread function (PSF), multiplied by (or sampled at) a bed-of-nails function. If the PSF is a tophat function sized to the grid elements, the pixels will appear as bins. The Pixel Space is unitless, although the pseudo unit `pixel` may be used for clarity. Although the pixel space indexing, by its nature uses integers for its coordinates, operations performed in Pixel Space, such as calculating distribution moments, require allowing the use of reals for pixel coordinates.

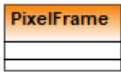
Package **pixel**

owner	<a href="#">domain</a>
properties	qualified name <code>stc2_coordinates::domain::pixel</code> visibility <code>public</code> «modelelement» <code>false</code> «modelimport» <code>false</code>
ownedDiagrams	<a href="#">PixelDomain</a>
ownedMember	<a href="#">Handedness</a> <a href="#">PixelCoord1D</a> <a href="#">PixelCoord2D</a> <a href="#">PixelCoord3D</a> <a href="#">PixelCoordinate</a> <a href="#">PixelFrame</a> <a href="#">PixelIndex</a> <a href="#">PixelSpace</a> <a href="#">RealPixelValue</a>

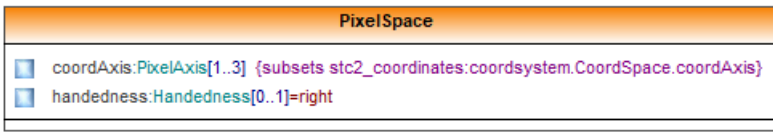
Class Diagram **PixelDomain** ([pixel](#))



### 4.8.1 Class PixelFrame

diagram																	
owner	<a href="#">pixel</a>																
properties	qualified name stc2_coordinates::domain::pixel::PixelFrame abstract false																
ownedMember	<a href="#">Subset</a>																
general	<a href="#">CoordFrame</a>																
typedElements	<table border="0"> <tr> <td>DataType</td> <td><a href="#">PixelCoordinate</a></td> <td>Property</td> <td><a href="#">coordFrame</a></td> </tr> <tr> <td>Class</td> <td><a href="#">PixelCoordSystem</a></td> <td>Property</td> <td><a href="#">pixelFrame</a></td> </tr> <tr> <td>DataType</td> <td><a href="#">PixelIndex</a></td> <td>Property</td> <td><a href="#">coordFrame</a></td> </tr> <tr> <td>DataType</td> <td><a href="#">RealPixelValue</a></td> <td>Property</td> <td><a href="#">coordFrame</a></td> </tr> </table>	DataType	<a href="#">PixelCoordinate</a>	Property	<a href="#">coordFrame</a>	Class	<a href="#">PixelCoordSystem</a>	Property	<a href="#">pixelFrame</a>	DataType	<a href="#">PixelIndex</a>	Property	<a href="#">coordFrame</a>	DataType	<a href="#">RealPixelValue</a>	Property	<a href="#">coordFrame</a>
DataType	<a href="#">PixelCoordinate</a>	Property	<a href="#">coordFrame</a>														
Class	<a href="#">PixelCoordSystem</a>	Property	<a href="#">pixelFrame</a>														
DataType	<a href="#">PixelIndex</a>	Property	<a href="#">coordFrame</a>														
DataType	<a href="#">RealPixelValue</a>	Property	<a href="#">coordFrame</a>														
shown on diagram	<a href="#">CoordSystems</a> <a href="#">PixelDomain</a>																
documentation	A PixelFrame forms the Coordinate Frame for a 1-D, 2-D, or 3-D PixelSpace, and is included in the PixelCoordSystem. It SHALL contain one PixelSpace object. Note that, unlike other coordinate frames, a Pixel Frame SHOULD have at least one Frame Transform mapping: pixels without reference to the wider world are meaningless.																

### 4.8.2 Class PixelSpace

diagram	
owner	<a href="#">pixel</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelSpace abstract false
ownedMember	<a href="#">coordAxis</a> <a href="#">handedness</a> <a href="#">Subset</a>
general	<a href="#">CoordSpace</a>
shown on diagram	<a href="#">PixelDomain</a>
documentation	A PixelSpace SHALL include one, two, or three ordered PixelAxis objects, defining a pixel object (sub)set that can be the native or target space for a FrameTransform mapping object.

#### Constraint PixelSpace::Subset

owner	<a href="#">PixelSpace</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelSpace::Subset specification subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis constrained elements <a href="#">coordAxis</a>


#### Property PixelSpace::coordAxis

owner	<a href="#">PixelSpace</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelSpace::coordAxis multiplicity 1..3 type <a href="#">PixelAxis</a> aggregation composite
constraints	<a href="#">Subset</a> : subsets stc2_coordinates:coordsystem.CoordSpace.coordAxis

Property **PixelSpace::handedness**

owner	<a href="#">PixelSpace</a>
properties	qualified name <code>stc2_coordinates::domain::pixel::PixelSpace::handedness</code> multiplicity <code>0..1</code> type <a href="#">Handedness</a>

### 4.8.3 Enumeration Handedness

diagram	
owner	<a href="#">pixel</a>
properties	qualified name <code>stc2_coordinates::domain::pixel::Handedness</code> abstract <code>false</code>
ownedMember	<b>left right</b>
typedElements	Class <a href="#">PixelSpace</a> Property <a href="#">handedness</a>
shown on diagram	<a href="#">PixelDomain</a>
documentation	Handedness of a coordinate frame.

### 4.8.4 DataType PixelIndex

owner	<a href="#">pixel</a>
properties	qualified name <code>stc2_coordinates::domain::pixel::PixelIndex</code> abstract <code>false</code>
ownedMember	<a href="#">coordAxis</a> <a href="#">coordFrame</a> <a href="#">Subset1</a> <a href="#">Subset2</a>
general	<a href="#">IntegerCoordValue</a>
shown on diagram	<a href="#">PixelDomain</a>
documentation	The 1-dimensional Pixel Index value.

Constraint **PixelIndex::Subset1**

owner	<a href="#">PixelIndex</a>
properties	qualified name <code>stc2_coordinates::domain::pixel::PixelIndex::Subset1</code> specification <code>subsets stc2_coordinates::coords::CoordValue::coordAxis</code> constrained elements <a href="#">coordAxis</a>

Constraint **PixelIndex::Subset2**

owner	<a href="#">PixelIndex</a>
properties	qualified name <code>stc2_coordinates::domain::pixel::PixelIndex::Subset2</code> specification <code>subsets stc2_coordinates::coords::BasicCoordValue::coordFrame</code> constrained elements <a href="#">coordFrame</a>

Property **PixelIndex::coordAxis**

owner	<a href="#">PixelIndex</a>
properties	qualified name <code>stc2_coordinates::domain::pixel::PixelIndex::coordAxis</code> multiplicity <code>1</code>

	type <a href="#">PixelAxis</a>
constraints	<a href="#">Subset1</a> : subsets stc2_coordinates:coords.CoordValue.coordAxis

Property **PixelIndex::coordFrame**

owner	<a href="#">PixelIndex</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelIndex::coordFrame type <a href="#">PixelFrame</a>
constraints	<a href="#">Subset2</a> : subsets stc2_coordinates:coords.BasicCoordValue.coordFrame

4.8.5 DataType **RealPixelValue**

owner	<a href="#">pixel</a>
properties	qualified name stc2_coordinates::domain::pixel::RealPixelValue abstract false
ownedMember	<a href="#">coordAxis</a> <a href="#">coordFrame</a> <a href="#">Subset1</a> <a href="#">Subset2</a>
general	<a href="#">RealCoordValue</a>
typedElements	DataType <a href="#">PixelCoord1D</a> Property <a href="#">cmpt</a> DataType <a href="#">PixelCoord2D</a> Property <a href="#">cmpt</a> DataType <a href="#">PixelCoord3D</a> Property <a href="#">cmpt</a>
shown on diagram	<a href="#">PixelDomain</a>
documentation	The 1-dimensional pixel value, to be used for non-index (non-integer) pixel coordinate values, such as may result from analysis calculations.

Constraint **RealPixelValue::Subset1**

owner	<a href="#">RealPixelValue</a>
properties	qualified name stc2_coordinates::domain::pixel::RealPixelValue::Subset1 specification subsets stc2_coordinates:coords.CoordValue.coordAxis constrained elements <a href="#">coordAxis</a>

Constraint **RealPixelValue::Subset2**

owner	<a href="#">RealPixelValue</a>
properties	qualified name stc2_coordinates::domain::pixel::RealPixelValue::Subset2 specification subsets stc2_coordinates:coords.BasicCoordValue.coordFrame constrained elements <a href="#">coordFrame</a>

Property **RealPixelValue::coordAxis**

owner	<a href="#">RealPixelValue</a>
properties	qualified name stc2_coordinates::domain::pixel::RealPixelValue::coordAxis multiplicity 1 type <a href="#">PixelAxis</a>
constraints	<a href="#">Subset1</a> : subsets stc2_coordinates:coords.CoordValue.coordAxis

Property **RealPixelValue::coordFrame**

owner	<a href="#">RealPixelValue</a>
properties	qualified name stc2_coordinates::domain::pixel::RealPixelValue::coordFrame type <a href="#">PixelFrame</a>
constraints	<a href="#">Subset2</a> : subsets stc2_coordinates:coords.BasicCoordValue.coordFrame



## 4.8.6 DataType PixelCoordinate

owner	<a href="#">pixel</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoordinate abstract true
ownedMember	<a href="#">coordFrame Subset</a>
general	<a href="#">MultiCoordValue</a>
specific	<a href="#">PixelCoord1D</a> <a href="#">PixelCoord2D</a> <a href="#">PixelCoord3D</a>
shown on diagram	<a href="#">PixelDomain</a>
documentation	A PixelCoordinate is a real-valued coordinate in 1-, 2-, or 3-dimensional pixel space, with each component referenced to its pixel frame through its associated Pixel Axis. This data type specifies an arbitrary location in the Pixel Frame, rather than identifying a specific pixel. The vector is built up from RealPixelValue scalar components.

### Constraint PixelCoordinate::Subset

owner	<a href="#">PixelCoordinate</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoordinate::Subset specification subsets stc2_coordinates::coords::BasicCoordValue::coordFrame constrained elements <a href="#">coordFrame</a>

### Property PixelCoordinate::coordFrame

owner	<a href="#">PixelCoordinate</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoordinate::coordFrame type <a href="#">PixelFrame</a>
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coords::BasicCoordValue::coordFrame

## 4.8.7 DataType PixelCoord1D

owner	<a href="#">pixel</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord1D abstract false
ownedMember	<a href="#">cmpt Subset</a>
general	<a href="#">PixelCoordinate</a>
shown on diagram	<a href="#">PixelDomain</a>
documentation	The 1-dimensional Pixel Coordinate value.

### Constraint PixelCoord1D::Subset

owner	<a href="#">PixelCoord1D</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord1D::Subset specification subsets stc2_coordinates::coords::MultiCoordValue::cmpt constrained elements <a href="#">cmpt</a>

### Property PixelCoord1D::cmpt

owner	<a href="#">PixelCoord1D</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord1D::cmpt multiplicity 1 type <a href="#">RealPixelValue</a>
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coords::MultiCoordValue::cmpt

### 4.8.8 DataType PixelCoord2D

owner	<a href="#">pixel</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord2D abstract false
ownedMember	<a href="#">cmpt Subset</a>
general	<a href="#">PixelCoordinate</a>
shown on diagram	<a href="#">PixelDomain</a>
documentation	The 2-dimensional Pixel Coordinate value.

#### Constraint PixelCoord2D::Subset

owner	<a href="#">PixelCoord2D</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord2D::Subset specification subsets stc2_coordinates::coords.MultiCoordValue.cmpt constrained elements <a href="#">cmpt</a>

#### Property PixelCoord2D::cmpt

owner	<a href="#">PixelCoord2D</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord2D::cmpt multiplicity 2 type <a href="#">RealPixelValue</a>
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coords.MultiCoordValue.cmpt

### 4.8.9 DataType PixelCoord3D

owner	<a href="#">pixel</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord3D abstract false
ownedMember	<a href="#">cmpt Subset</a>
general	<a href="#">PixelCoordinate</a>
shown on diagram	<a href="#">PixelDomain</a>
documentation	The 3-dimensional Pixel Coordinate value.

#### Constraint PixelCoord3D::Subset

owner	<a href="#">PixelCoord3D</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord3D::Subset specification subsets stc2_coordinates::coords.MultiCoordValue.cmpt constrained elements <a href="#">cmpt</a>

#### Property PixelCoord3D::cmpt

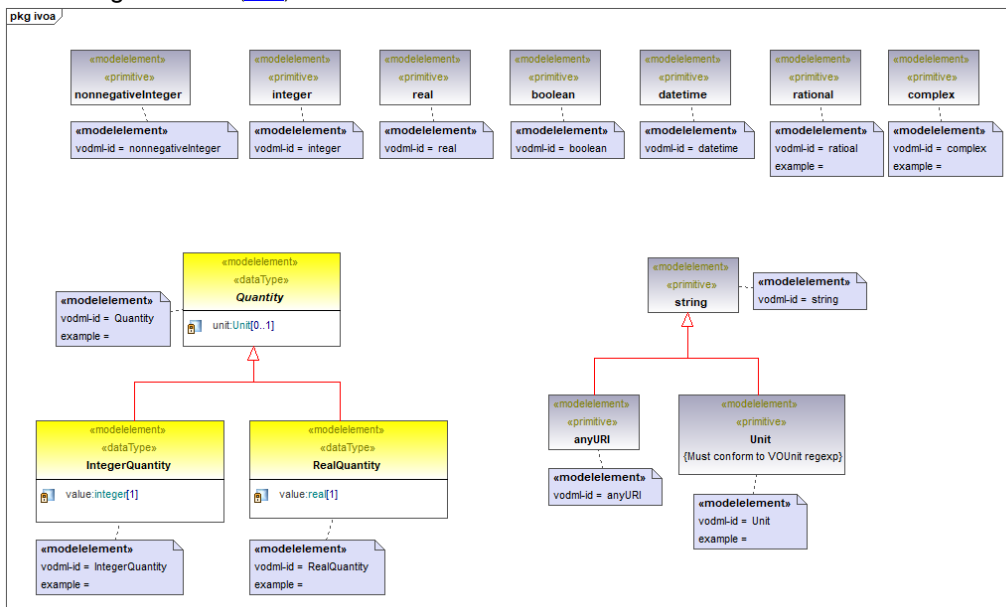
owner	<a href="#">PixelCoord3D</a>
properties	qualified name stc2_coordinates::domain::pixel::PixelCoord3D::cmpt multiplicity 3 type <a href="#">RealPixelValue</a>
constraints	<a href="#">Subset</a> : subsets stc2_coordinates::coords.MultiCoordValue.cmpt

## 5 Imported Models: ivoa Model and IVOA UML Profile

### Model ivoa

owner	<a href="#">stc2_coordinates</a>	
properties	qualified name «modelimport» <a href="#">url</a> <a href="#">documentationURL</a>	stc2_coordinates::ivoa true http://volute.g-vo.org/svn/trunk/projects/dm/vo-dml/models/ivoa/vo-dml/IVOA-v1.0.vo-dml.xml http://volute.g-vo.org/svn/trunk/projects/dm/vo-dml/models/ivoa/vo-dml/IVOA.html
ownedDiagrams	<a href="#">ivoa</a>	
ownedMember	anyURI boolean complex datetime integer <a href="#">IntegerQuantity</a> nonnegativeInteger <a href="#">Quantity</a> rational real <a href="#">RealQuantity</a> string Unit	
source of relation	Dependency ProfileApplication	IVOA_UML_Profile IVOA_UML_Profile
target of relation	Dependency	<a href="#">coordsystem</a>

### Class Diagram ivoa ([ivoa](#))



### Data Type IntegerQuantity

owner	<a href="#">ivoa</a>	
properties	qualified name abstract <a href="#">vodml-id</a>	stc2_coordinates::ivoa::IntegerQuantity false IntegerQuantity
ownedMember	<a href="#">value</a>	
general	<a href="#">Quantity</a>	
shown on diagram	<a href="#">ivoa</a>	

Property **IntegerQuantity::value**

owner	<a href="#">IntegerQuantity</a>
properties	qualified name stc2_coordinates::ivoa::IntegerQuantity::value multiplicity 1 type integer

Data Type **Quantity**

owner	<a href="#">ivoa</a>
properties	qualified name stc2_coordinates::ivoa::Quantity abstract true <a href="#">vodml-id</a> Quantity
ownedMember	<a href="#">unit</a>
specific	<a href="#">IntegerQuantity</a> <a href="#">RealQuantity</a>
typedElements	Class <a href="#">CoordAxis</a> Property <a href="#">domainMax</a> <a href="#">domainMin</a> DataType <a href="#">PhysicalCoordValue</a> Property <a href="#">cval</a>
shown on diagram	<a href="#">ivoa</a>
documentation	Meant to represent the value of a numerical physical quantity. May be integer, what units can apply there?

Property **Quantity::unit**

owner	<a href="#">Quantity</a>
properties	qualified name stc2_coordinates::ivoa::Quantity::unit multiplicity 0..1 type Unit

Data Type **RealQuantity**

owner	<a href="#">ivoa</a>																								
properties	qualified name stc2_coordinates::ivoa::RealQuantity abstract false <a href="#">vodml-id</a> RealQuantity																								
ownedMember	<a href="#">value</a>																								
general	<a href="#">Quantity</a>																								
typedElements	<table border="0"> <tr> <td>DataType <a href="#">Cartesian</a></td> <td>Property <a href="#">x</a> <a href="#">y</a> <a href="#">z</a></td> </tr> <tr> <td>DataType <a href="#">Ecliptic</a></td> <td>Property <a href="#">elat</a> <a href="#">elong</a> <a href="#">r</a></td> </tr> <tr> <td>DataType <a href="#">Equatorial</a></td> <td>Property <a href="#">dec</a> <a href="#">r</a> <a href="#">ra</a></td> </tr> <tr> <td>DataType <a href="#">Galactic</a></td> <td>Property <a href="#">b</a> <a href="#">l</a> <a href="#">r</a></td> </tr> <tr> <td>DataType <a href="#">GenericCoordValue</a></td> <td>Property <a href="#">cval</a></td> </tr> <tr> <td>DataType <a href="#">LongLat</a></td> <td>Property <a href="#">lat</a> <a href="#">long</a> <a href="#">r</a></td> </tr> <tr> <td>DataType <a href="#">RedshiftValue</a></td> <td>Property <a href="#">cval</a></td> </tr> <tr> <td>DataType <a href="#">SpatialCoordValue</a></td> <td>Property <a href="#">cval</a></td> </tr> <tr> <td>DataType <a href="#">SpectralValue</a></td> <td>Property <a href="#">cval</a></td> </tr> <tr> <td>DataType <a href="#">TimeCoordinateValue</a></td> <td>Property <a href="#">cval</a></td> </tr> <tr> <td>Class <a href="#">TimeFrame</a></td> <td>Property <a href="#">period</a></td> </tr> <tr> <td>DataType <a href="#">TimeOffset</a></td> <td>Property <a href="#">time</a></td> </tr> </table>	DataType <a href="#">Cartesian</a>	Property <a href="#">x</a> <a href="#">y</a> <a href="#">z</a>	DataType <a href="#">Ecliptic</a>	Property <a href="#">elat</a> <a href="#">elong</a> <a href="#">r</a>	DataType <a href="#">Equatorial</a>	Property <a href="#">dec</a> <a href="#">r</a> <a href="#">ra</a>	DataType <a href="#">Galactic</a>	Property <a href="#">b</a> <a href="#">l</a> <a href="#">r</a>	DataType <a href="#">GenericCoordValue</a>	Property <a href="#">cval</a>	DataType <a href="#">LongLat</a>	Property <a href="#">lat</a> <a href="#">long</a> <a href="#">r</a>	DataType <a href="#">RedshiftValue</a>	Property <a href="#">cval</a>	DataType <a href="#">SpatialCoordValue</a>	Property <a href="#">cval</a>	DataType <a href="#">SpectralValue</a>	Property <a href="#">cval</a>	DataType <a href="#">TimeCoordinateValue</a>	Property <a href="#">cval</a>	Class <a href="#">TimeFrame</a>	Property <a href="#">period</a>	DataType <a href="#">TimeOffset</a>	Property <a href="#">time</a>
DataType <a href="#">Cartesian</a>	Property <a href="#">x</a> <a href="#">y</a> <a href="#">z</a>																								
DataType <a href="#">Ecliptic</a>	Property <a href="#">elat</a> <a href="#">elong</a> <a href="#">r</a>																								
DataType <a href="#">Equatorial</a>	Property <a href="#">dec</a> <a href="#">r</a> <a href="#">ra</a>																								
DataType <a href="#">Galactic</a>	Property <a href="#">b</a> <a href="#">l</a> <a href="#">r</a>																								
DataType <a href="#">GenericCoordValue</a>	Property <a href="#">cval</a>																								
DataType <a href="#">LongLat</a>	Property <a href="#">lat</a> <a href="#">long</a> <a href="#">r</a>																								
DataType <a href="#">RedshiftValue</a>	Property <a href="#">cval</a>																								
DataType <a href="#">SpatialCoordValue</a>	Property <a href="#">cval</a>																								
DataType <a href="#">SpectralValue</a>	Property <a href="#">cval</a>																								
DataType <a href="#">TimeCoordinateValue</a>	Property <a href="#">cval</a>																								
Class <a href="#">TimeFrame</a>	Property <a href="#">period</a>																								
DataType <a href="#">TimeOffset</a>	Property <a href="#">time</a>																								
shown on diagram	<a href="#">ivoa</a>																								

Property **RealQuantity::value**

owner	<a href="#">RealQuantity</a>
properties	qualified name stc2_coordinates::ivoa::RealQuantity::value multiplicity 1 type real

**Constraint Unit::regexp**

owner	<b>Unit</b>	
properties	qualified name specification constrained elements	stc2_coordinates::ivoa::Unit::regexp Must conform to VOUnit regexp Unit

**Class Diagram IVOA\_UML\_Profile** (IVOA\_UML\_Profile)

