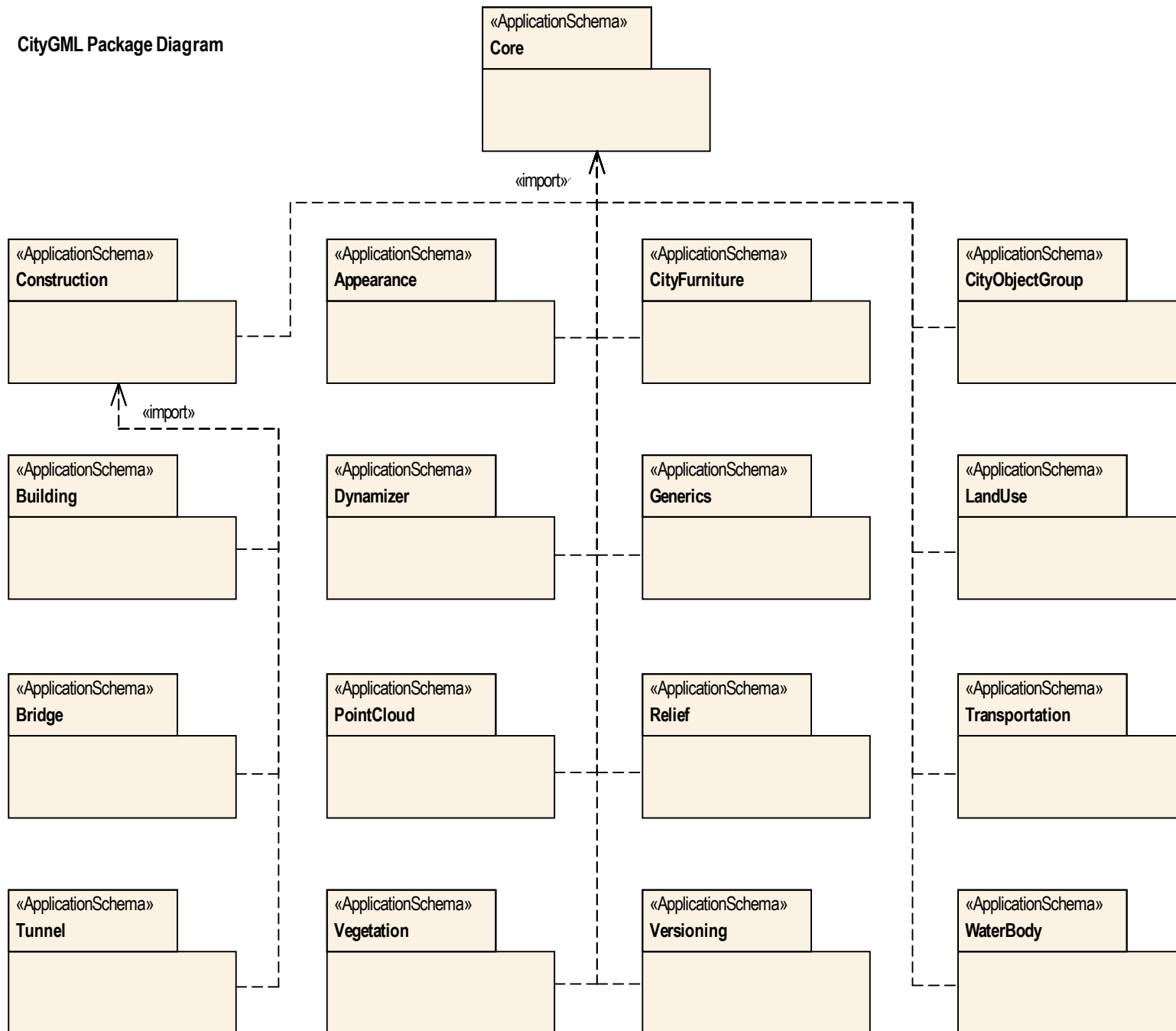
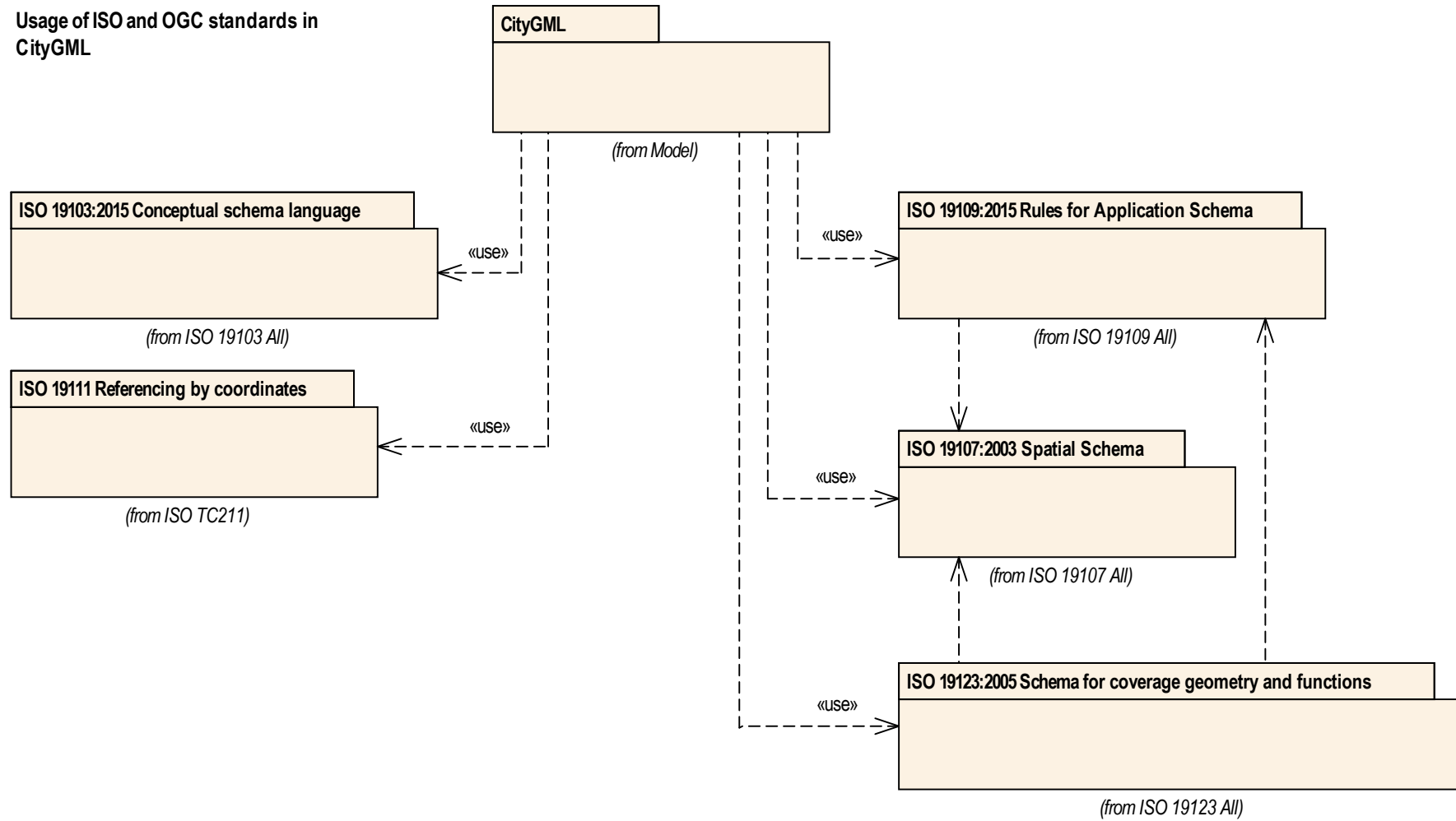


CityGML Package Diagram



Usage of ISO and OGC standards in CityGML



Color Scheme

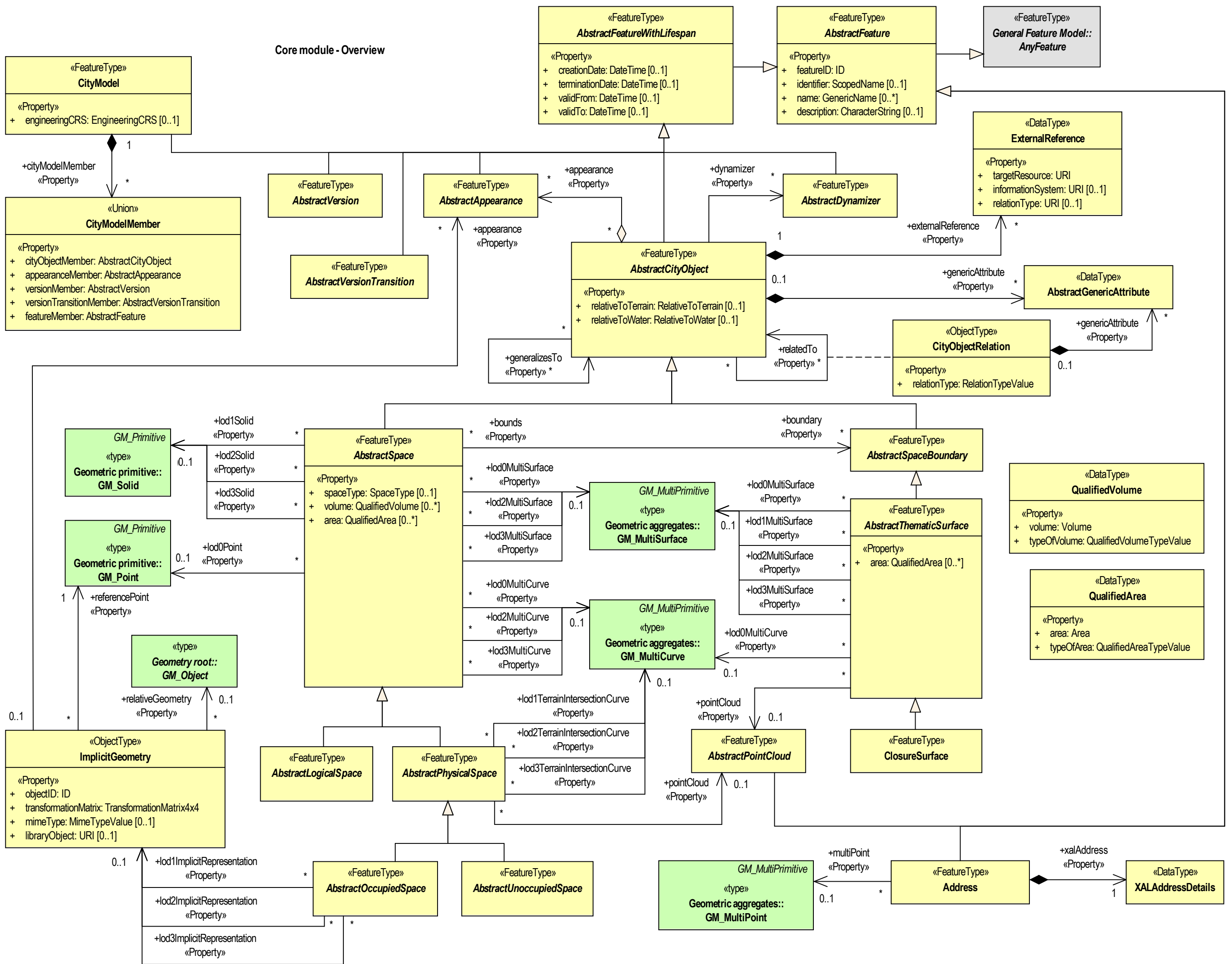
Class defined in this
Requirements Class

Class defined in another
Requirements Class

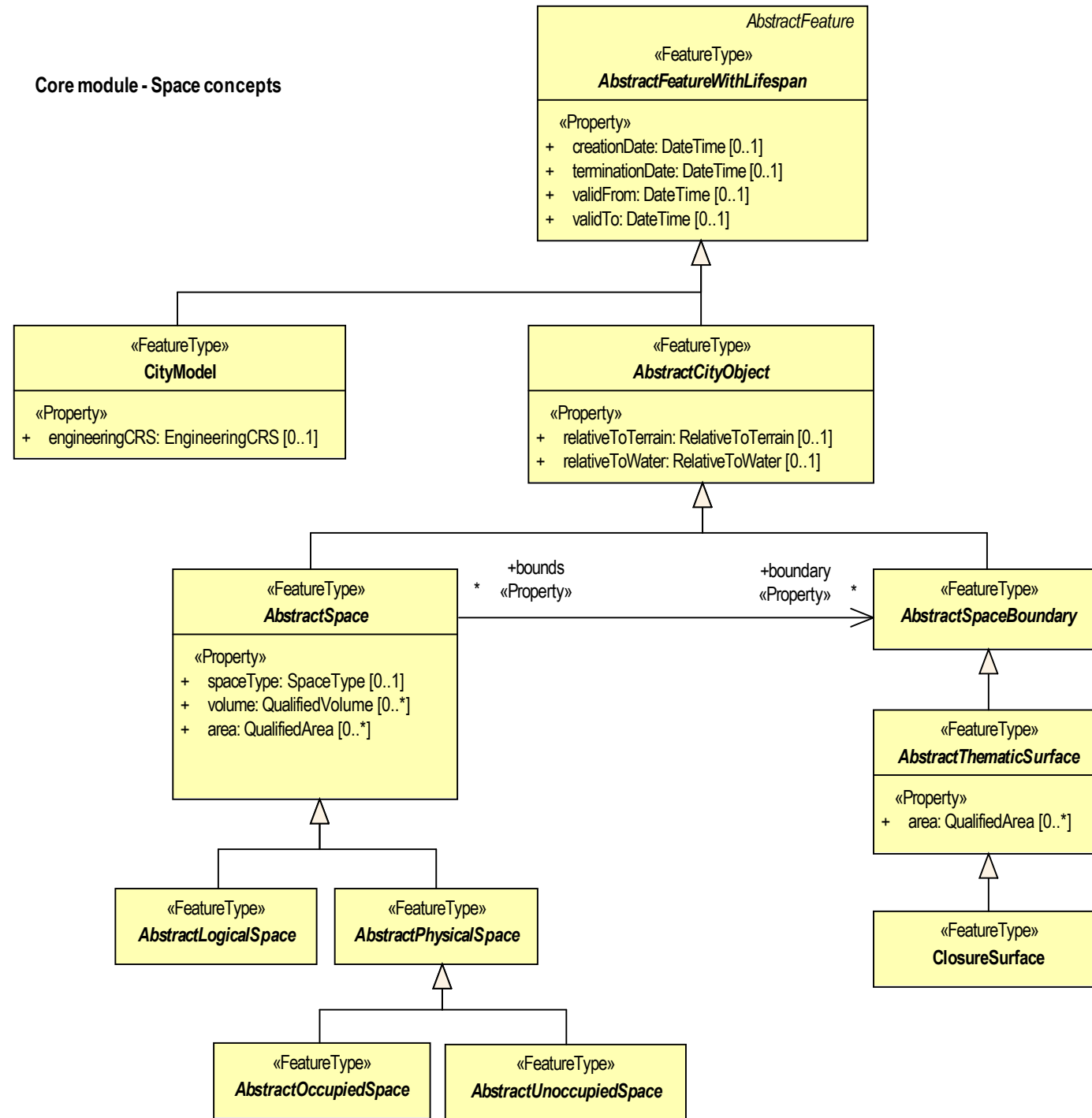
Class defined in ISO 19107,
ISO 19111 or ISO 19123

Class defined in ISO 19109

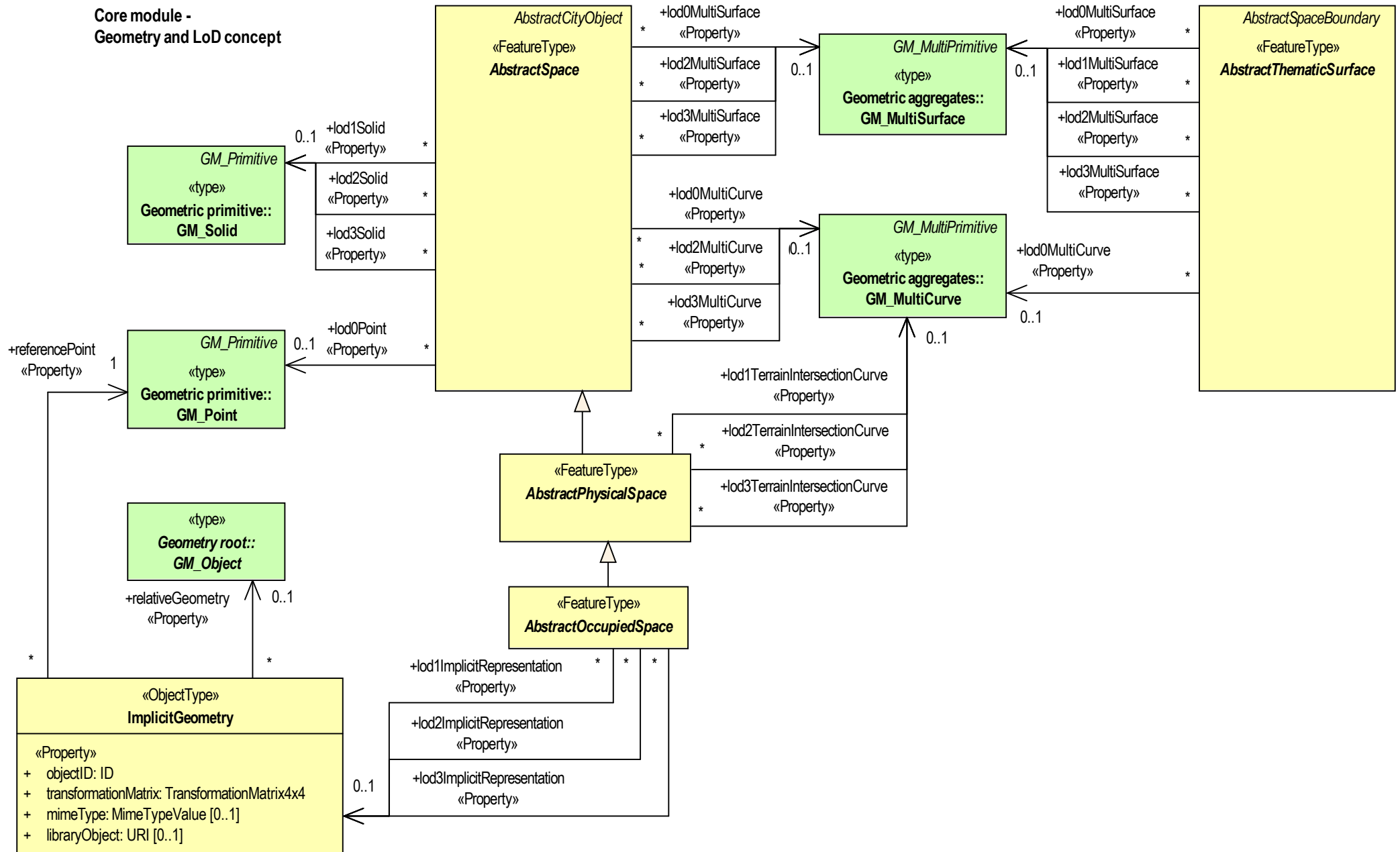
Notes and OCL constraints



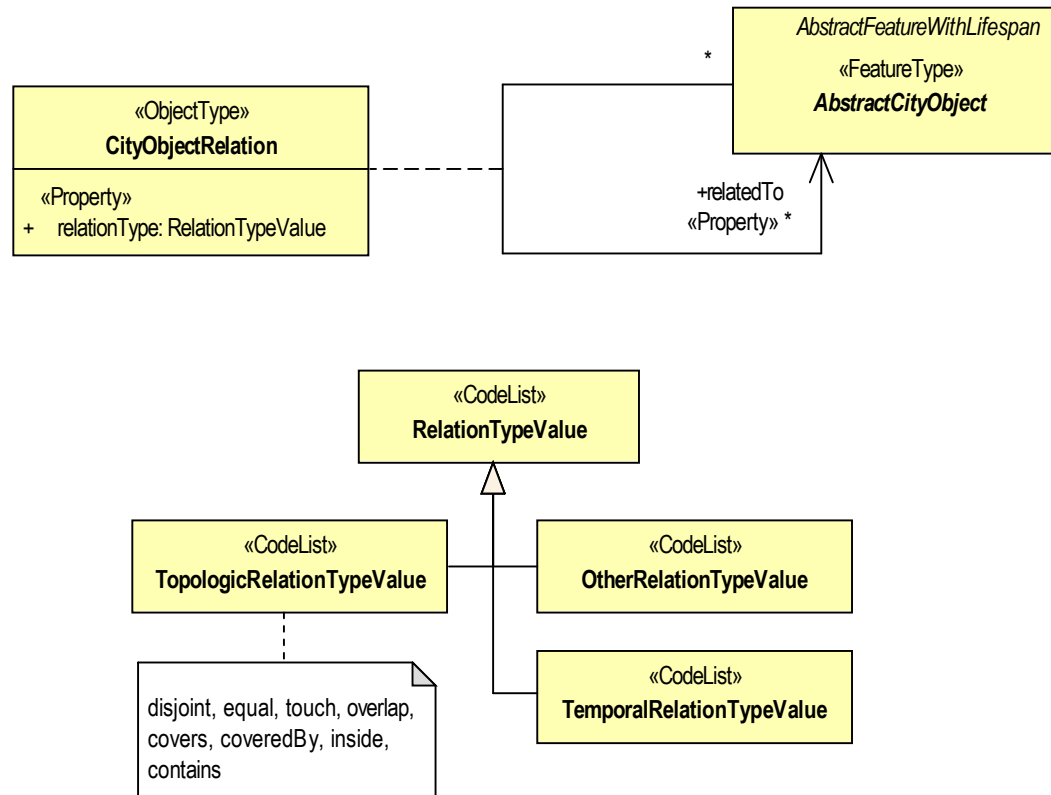
Core module - Space concepts



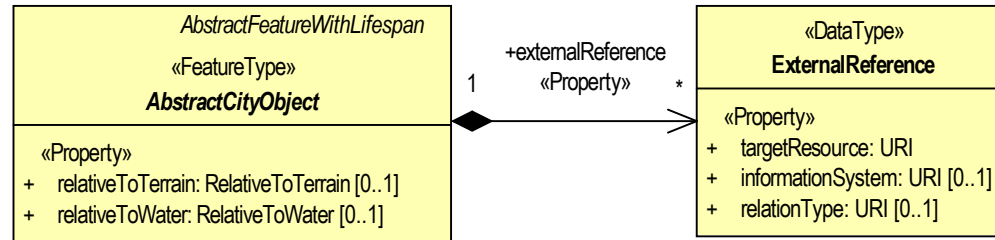
Core module - Geometry and LoD concept



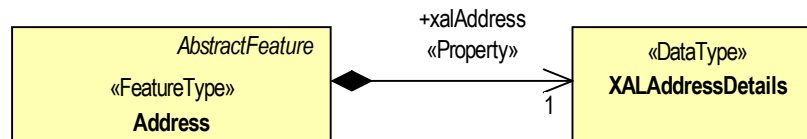
Core module - City object relations



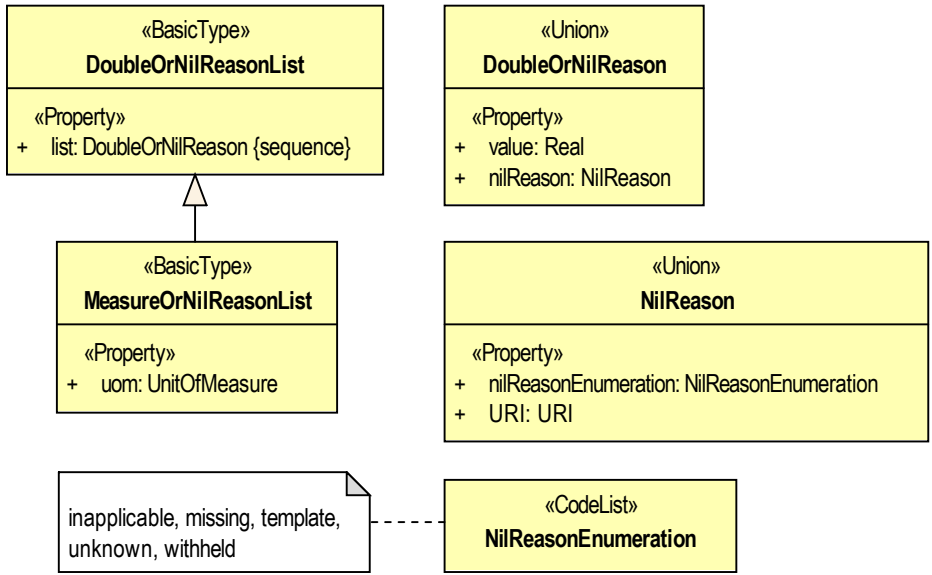
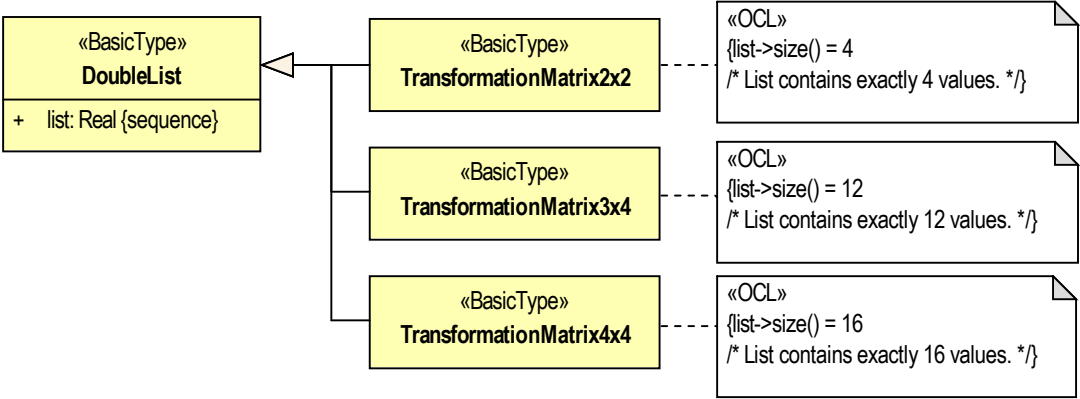
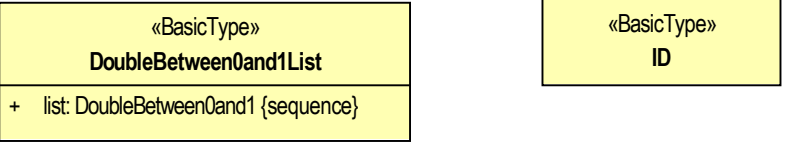
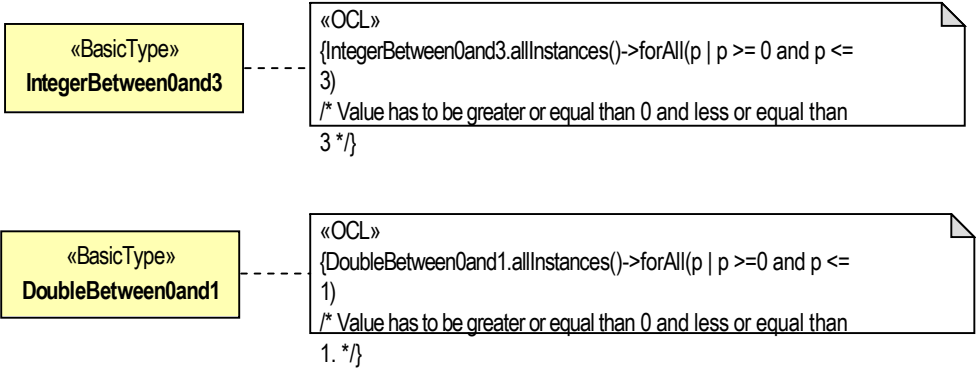
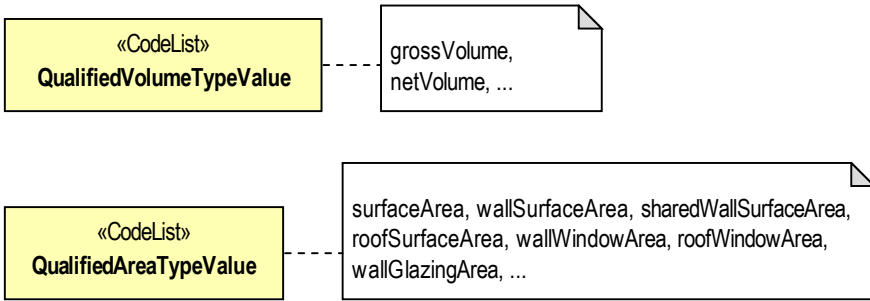
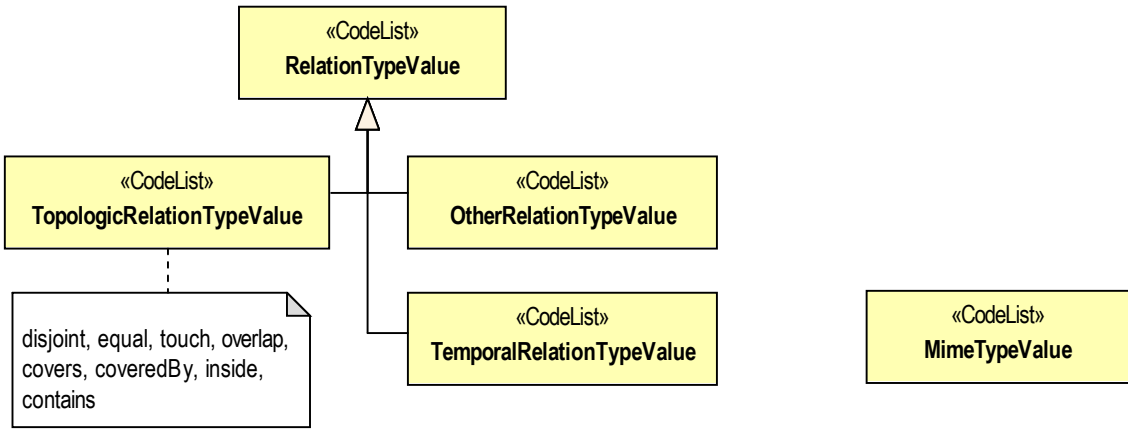
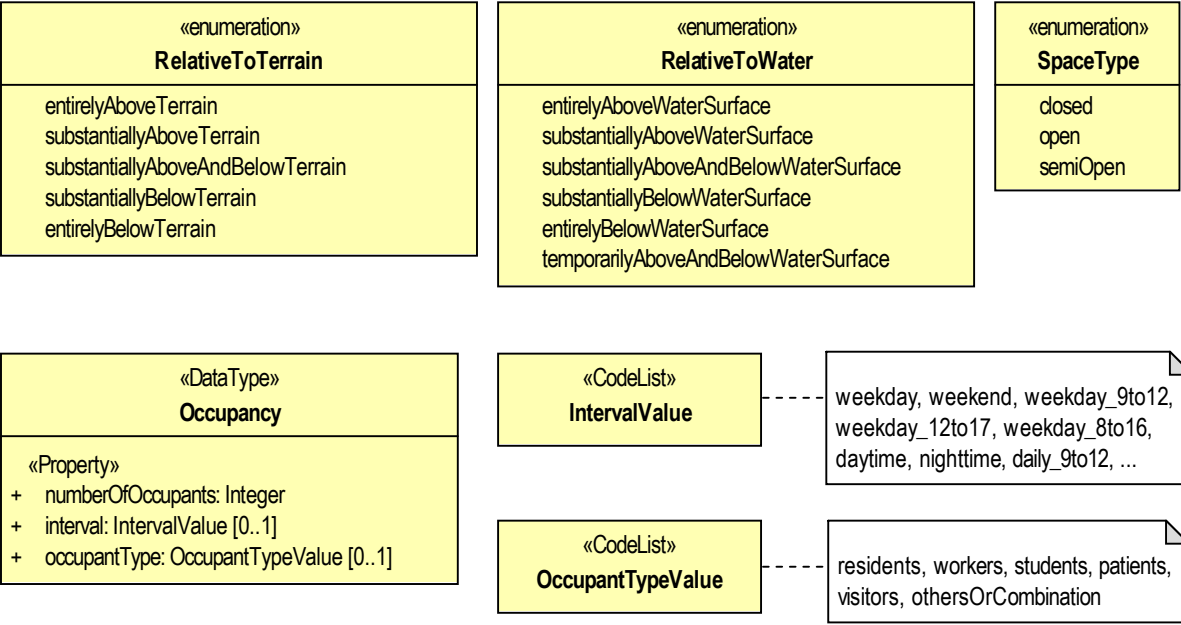
Core module - Miscellaneous

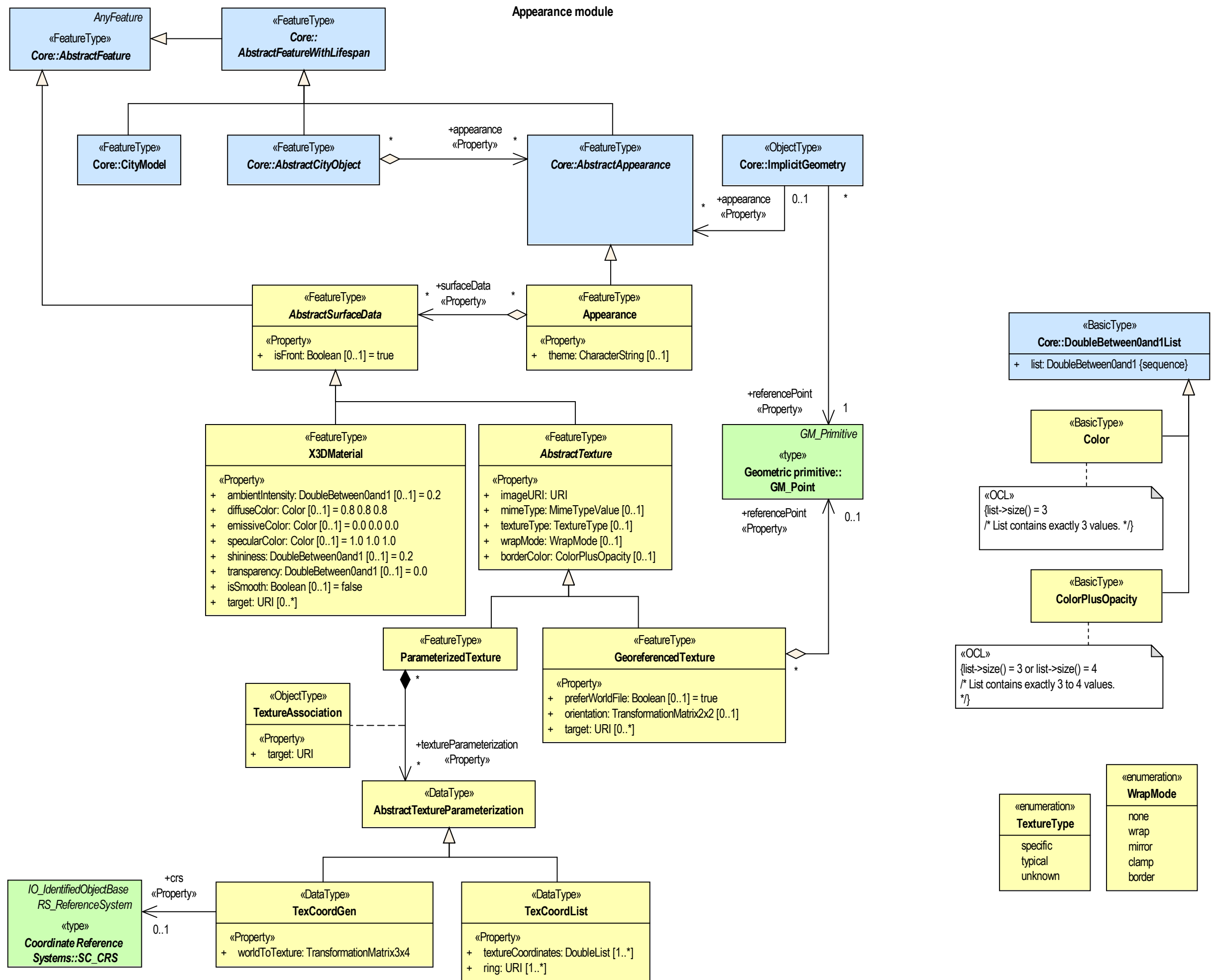


ExternalReference is now extended by an optional **relation Type** which can link to some external definition of the type of relation (e.g. the *sameAs* relation from OWL). Hence, **ExternalReferences** can now be used to express relations similar to RDF.

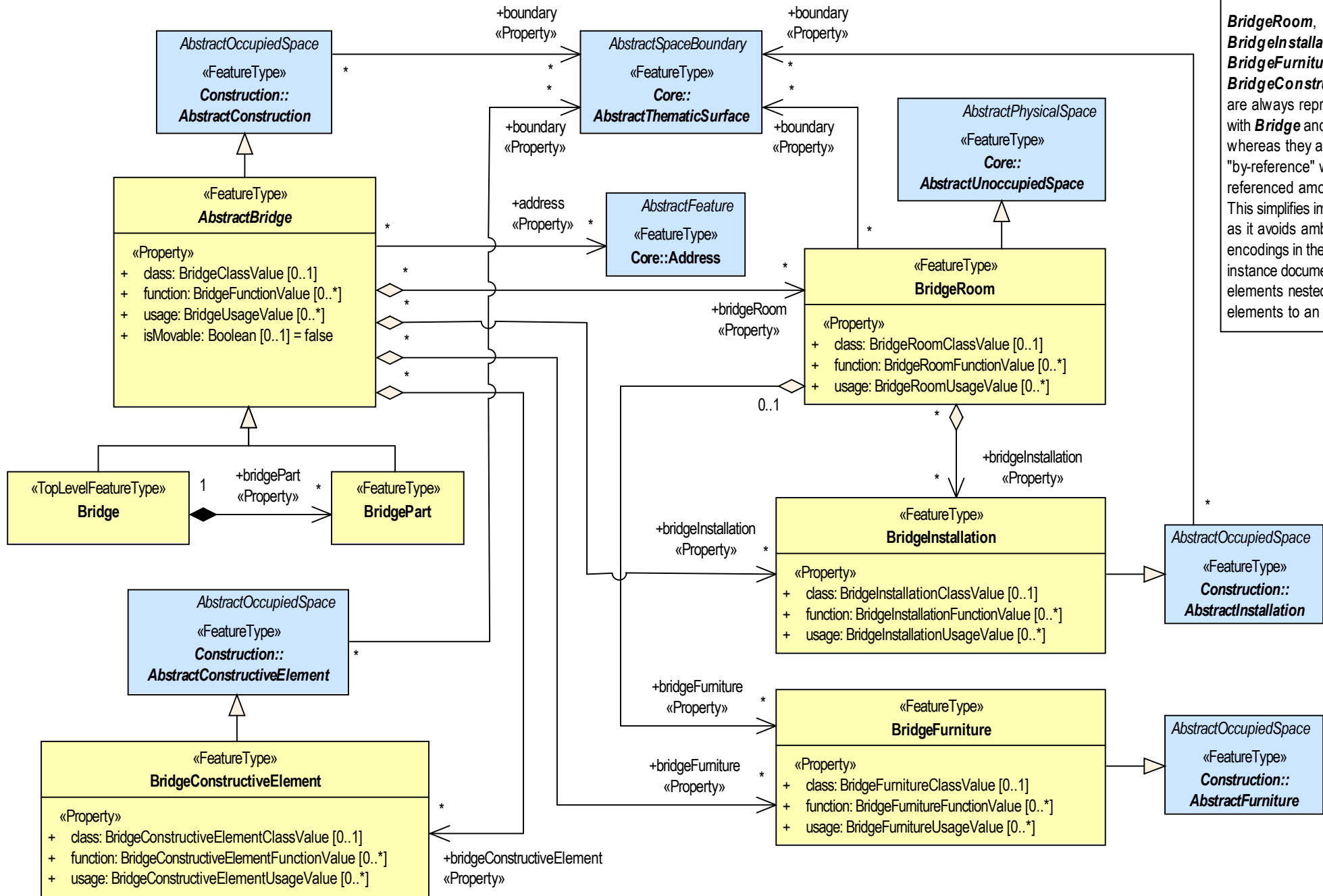


Core module - Basic Types, Enumerations, and Code lists



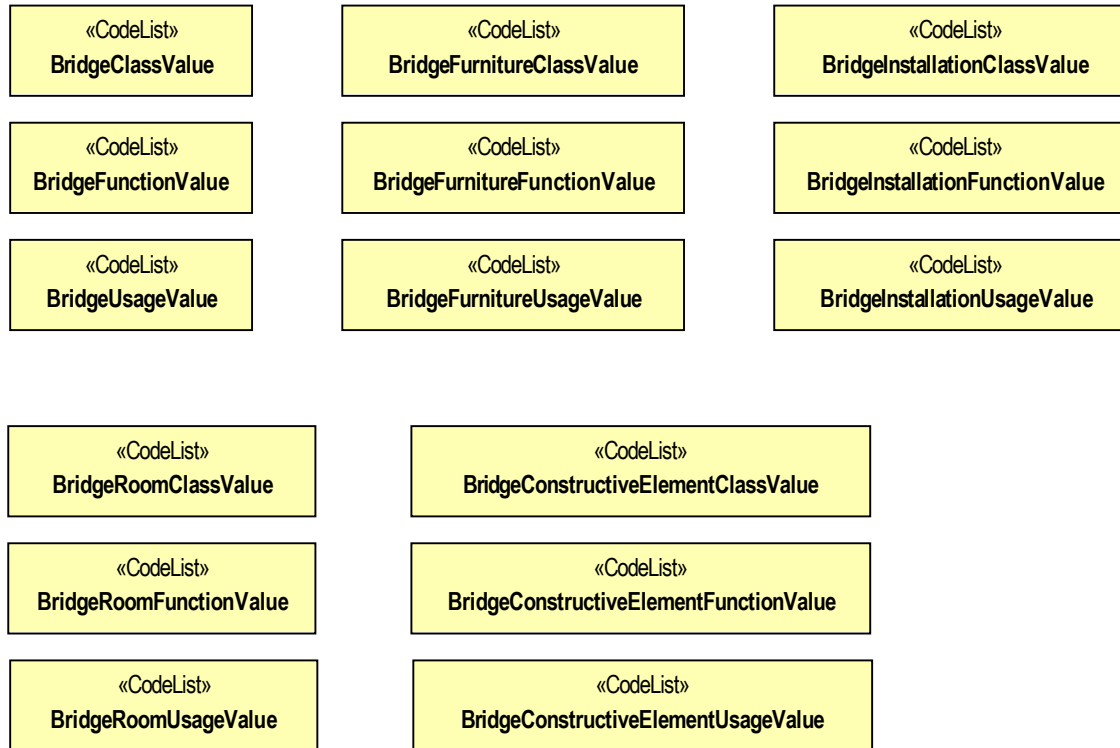


Bridge module

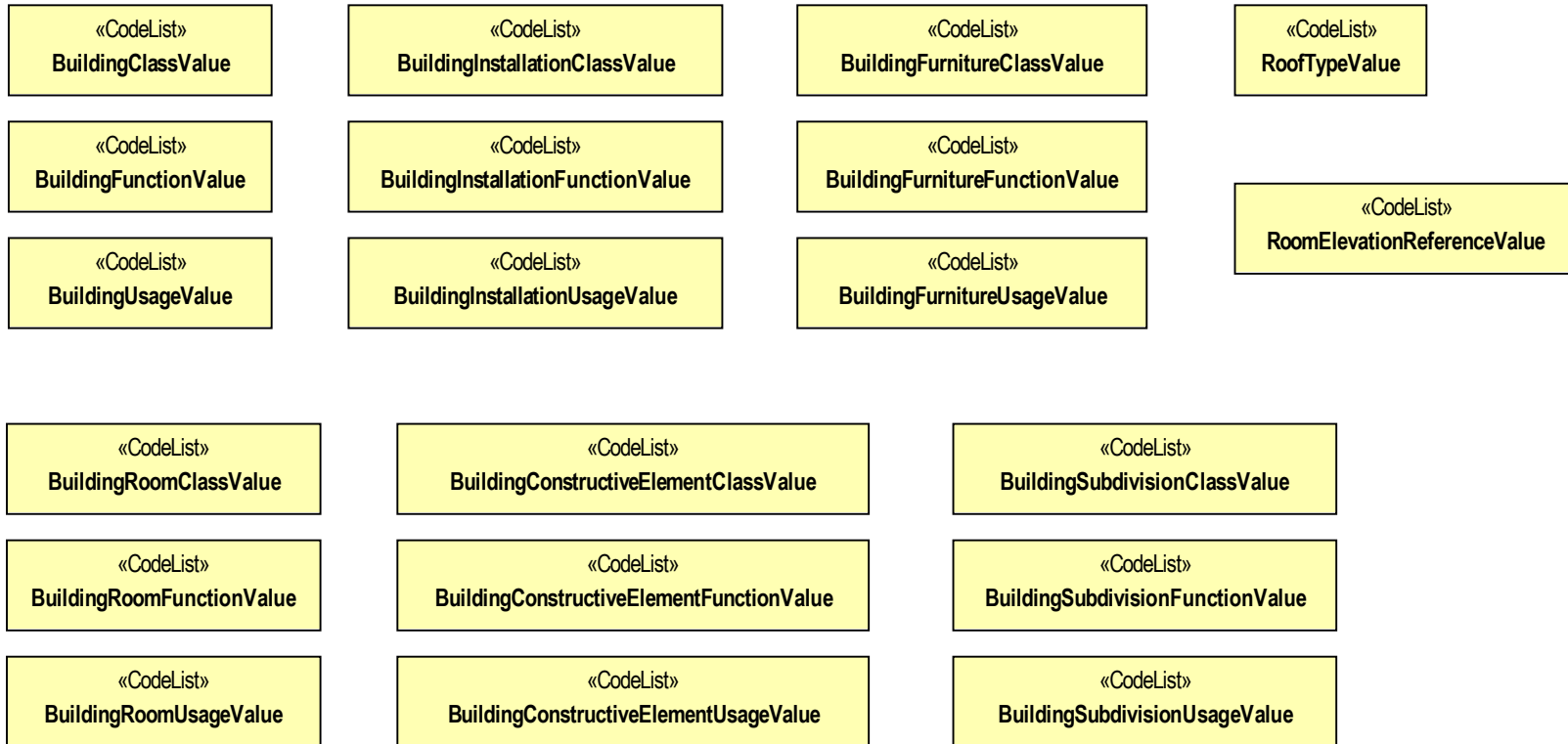


BridgeRoom, **BridgeInstallation**, **BridgeFurniture**, and **BridgeConstructiveElement** are always represented "inline" with **Bridge** and **BridgePart**, whereas they are represented "by-reference" when referenced amongst each other. This simplifies implementations as it avoids ambiguous encodings in the form of instance documents with elements nested inline of other elements to an arbitrary depth.

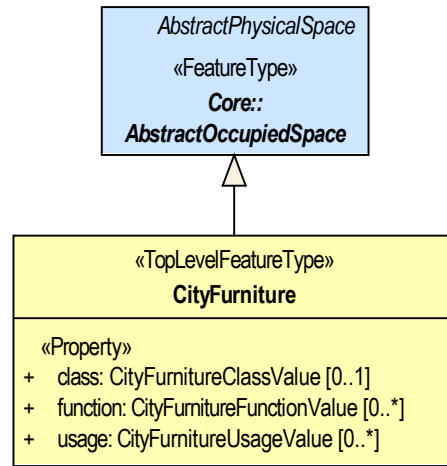
Bridge module - Code lists



Building module - Code lists



CityFurniture module



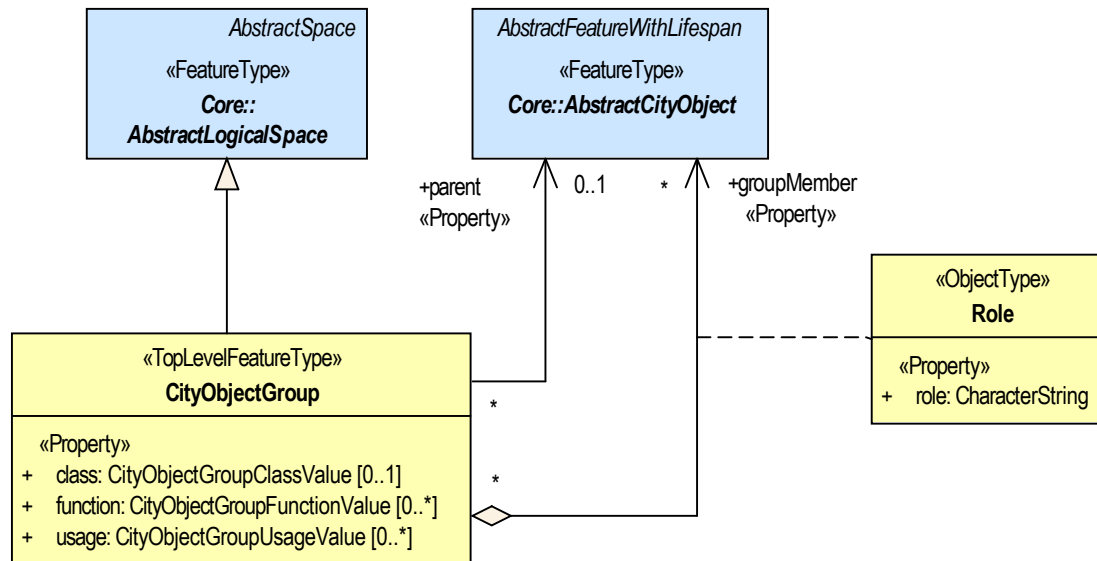
CityFurniture module - Code lists

«CodeList»
CityFurnitureClassValue

«CodeList»
CityFurnitureFunctionValue

«CodeList»
CityFurnitureUsageValue

CityObjectGroup module



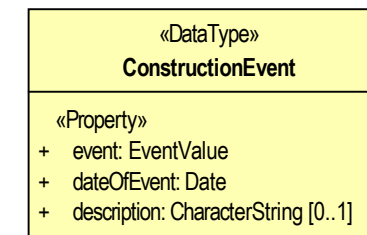
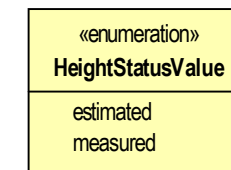
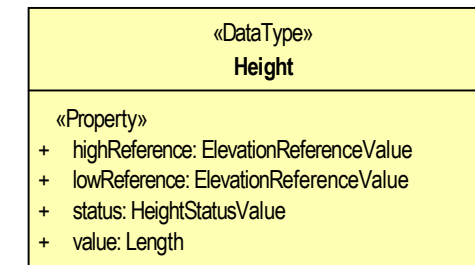
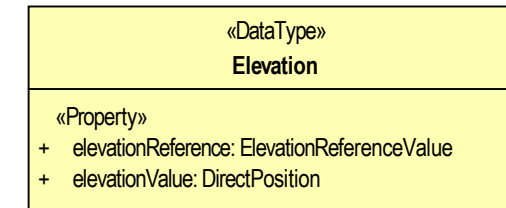
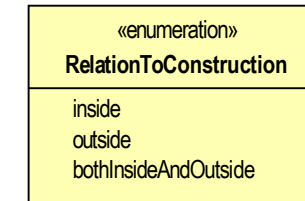
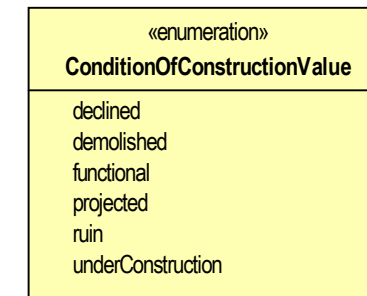
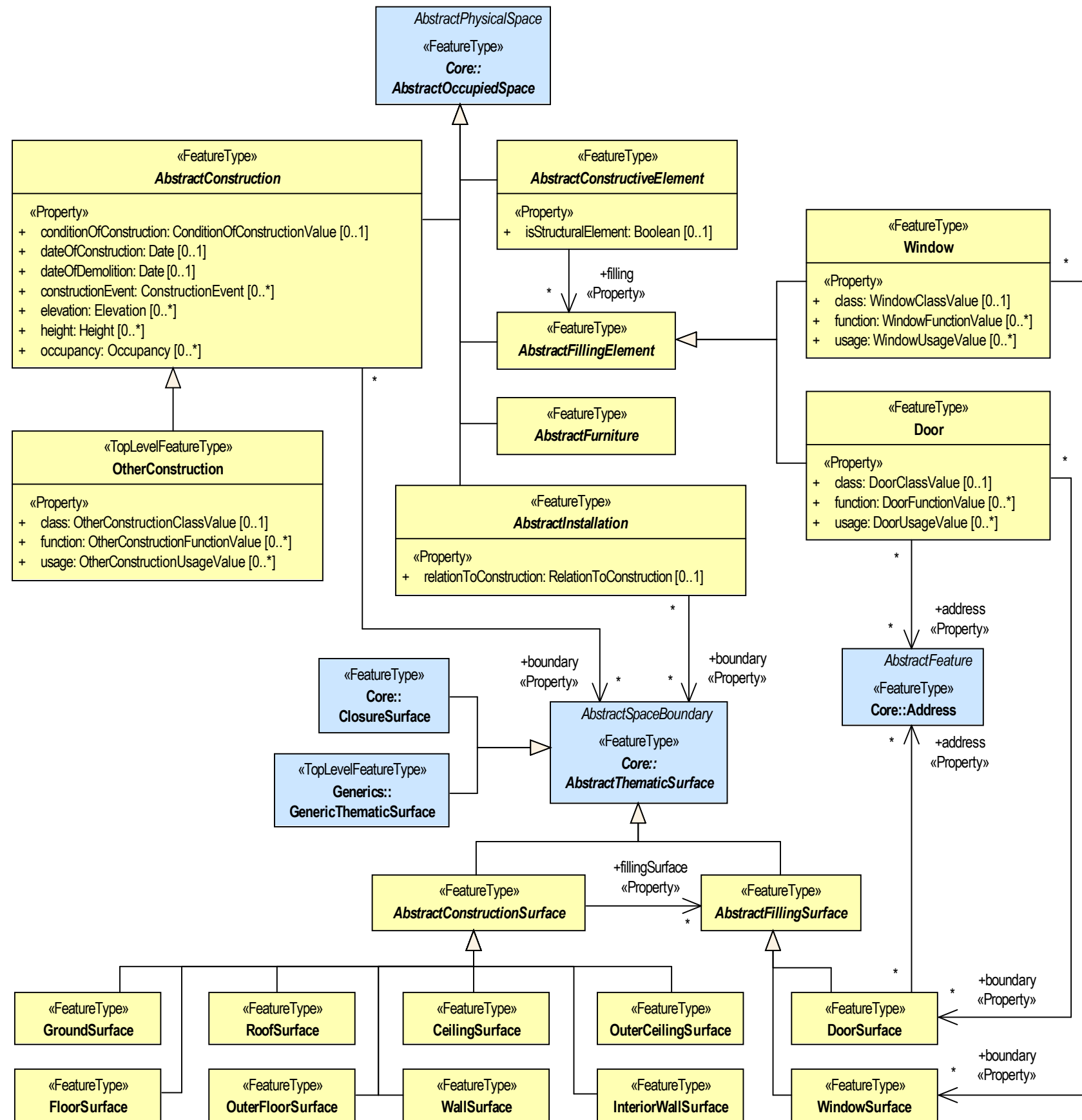
CityObjectGroup module - Code lists

«CodeList»
CityObjectGroupClassValue

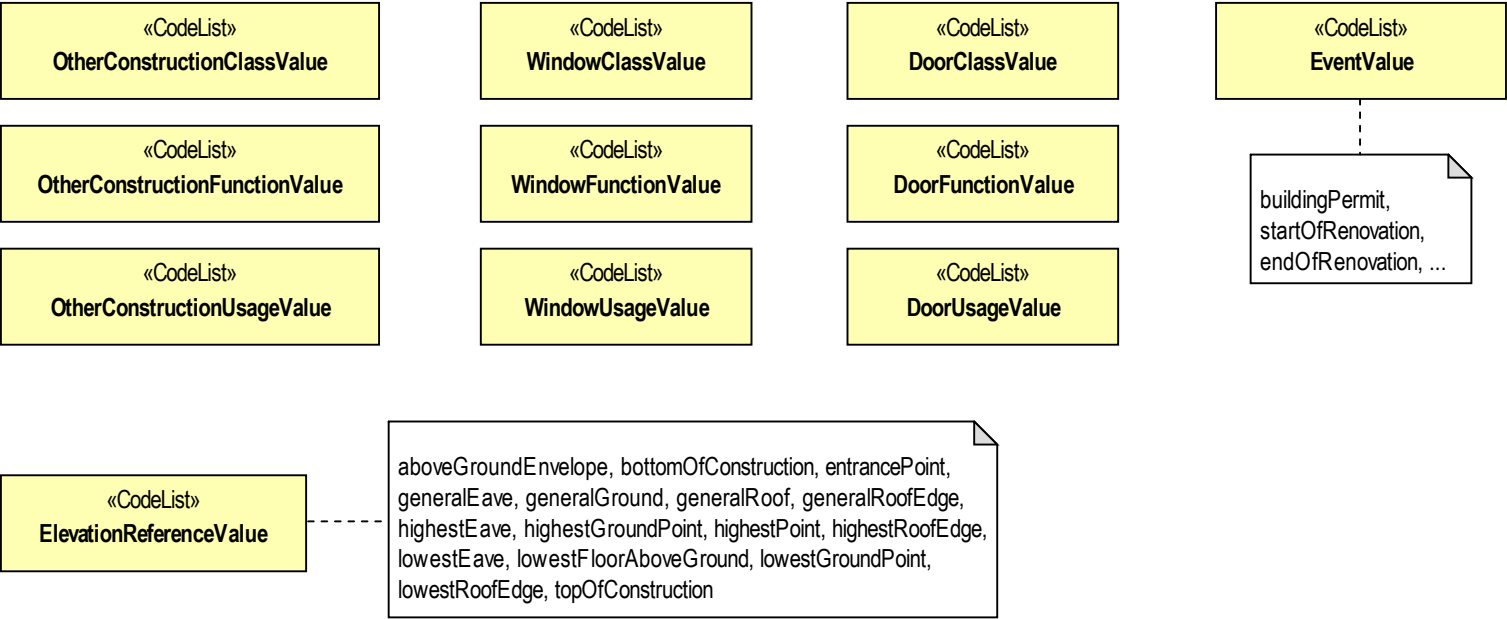
«CodeList»
CityObjectGroupFunctionValue

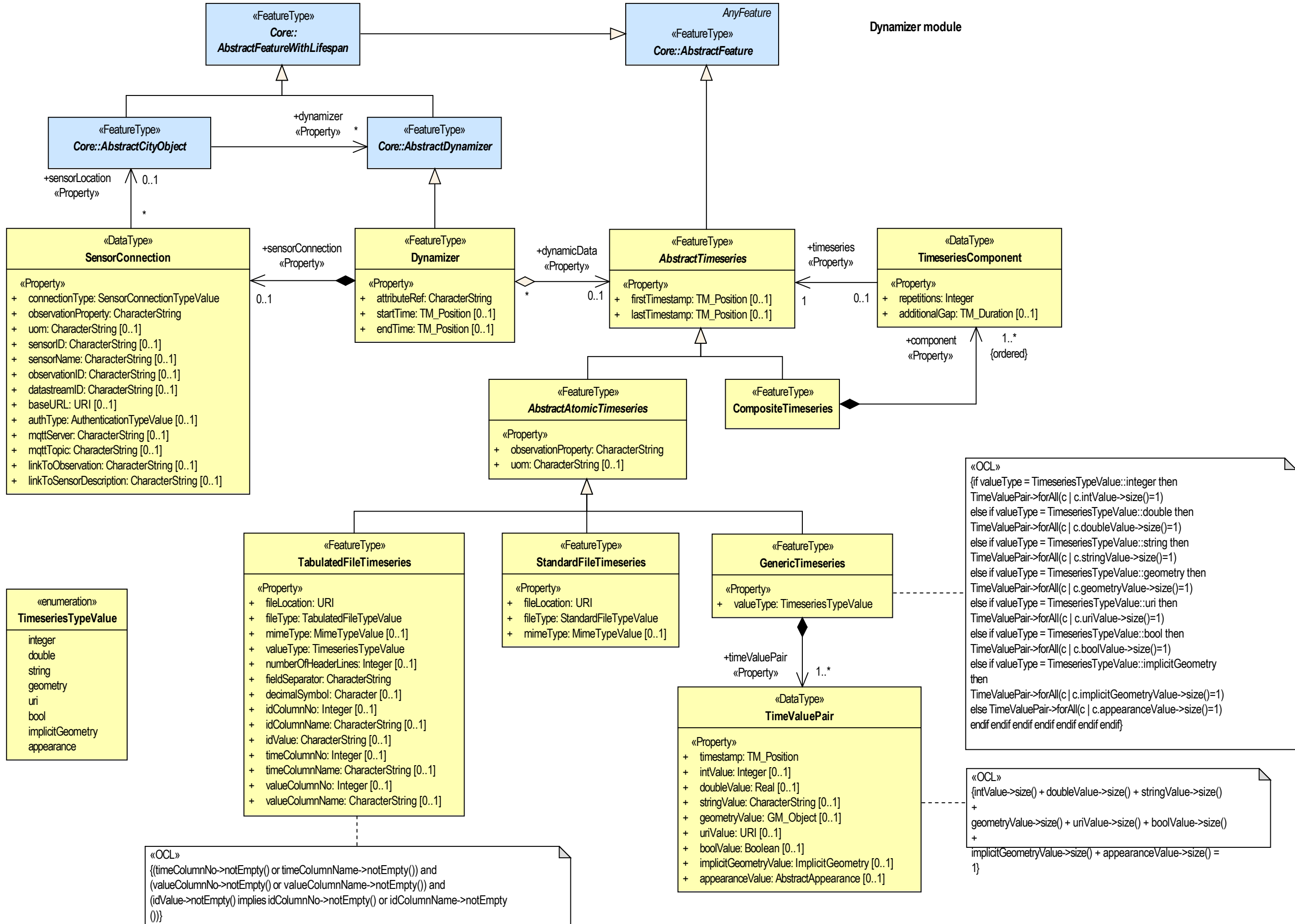
«CodeList»
CityObjectGroupUsageValue

Construction module

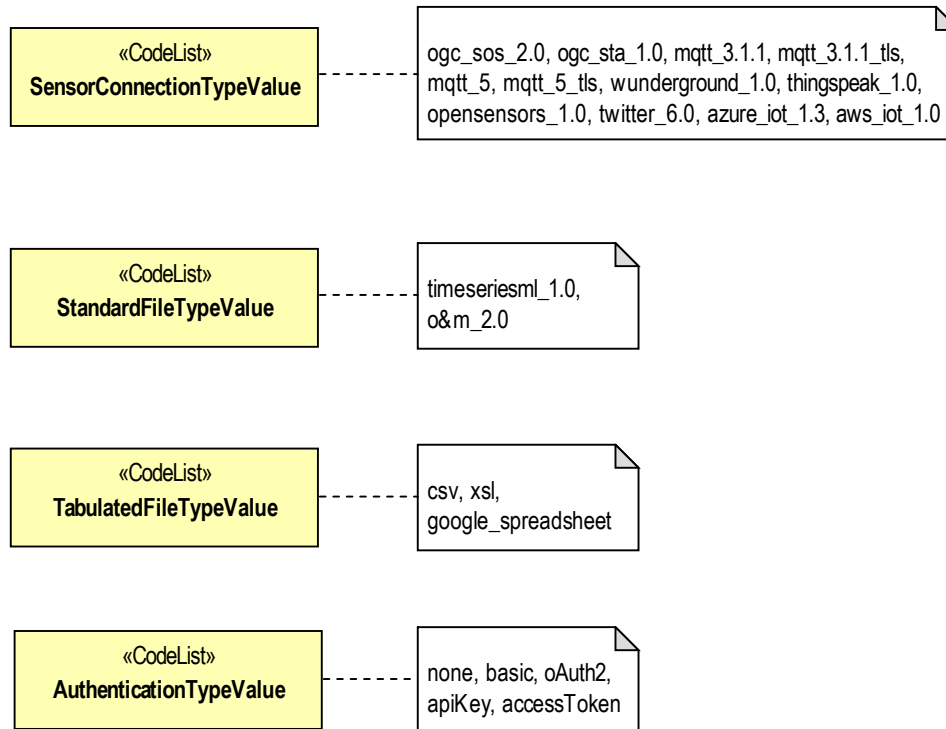


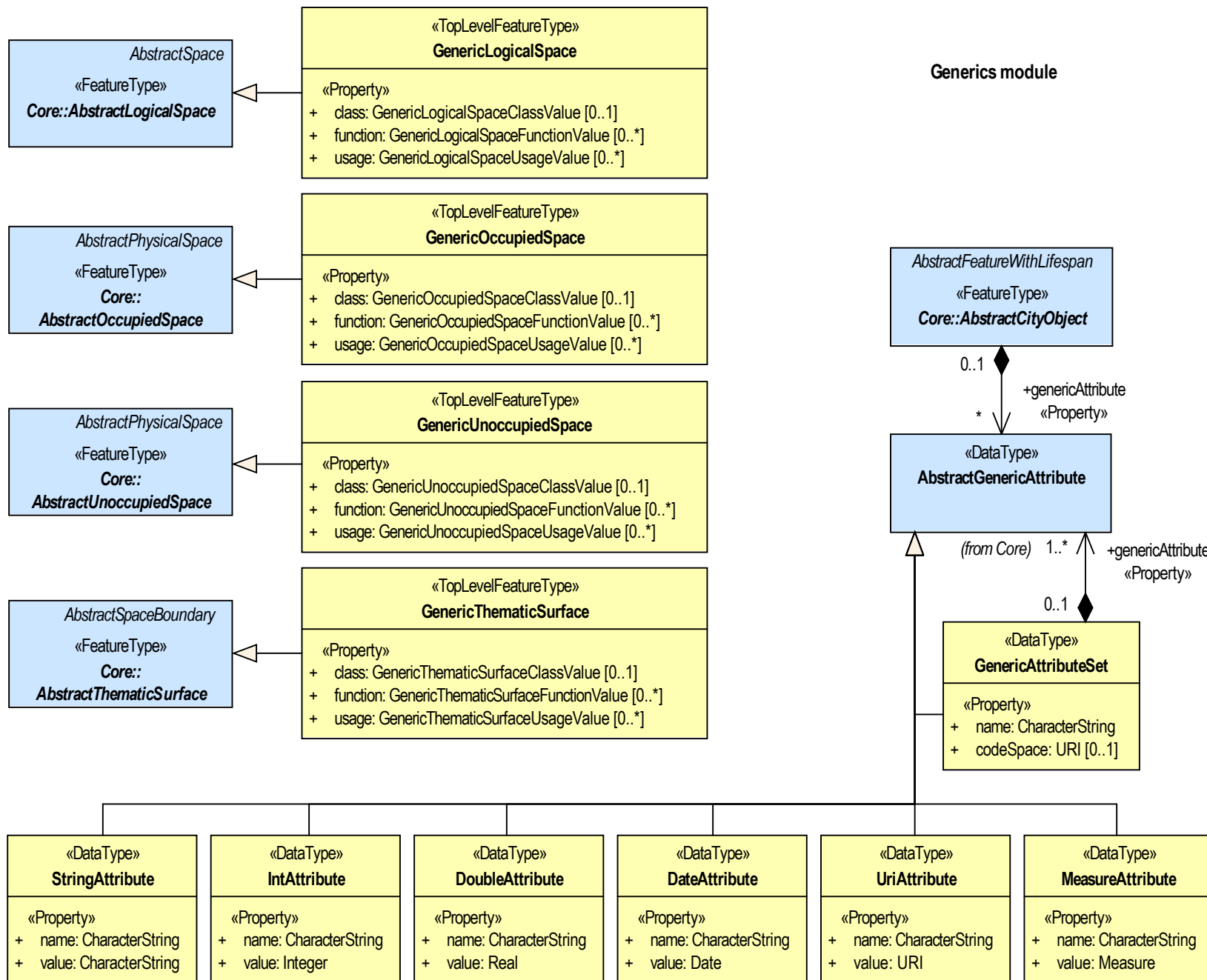
Construction module - Code lists





Dynamizer module - Code lists





Generics module - Code lists

«CodeList»
GenericLogicalSpaceClassValue

«CodeList»
GenericOccupiedSpaceClassValue

«CodeList»
GenericUnoccupiedSpaceClassValue

«CodeList»
GenericThematicSurfaceClassValue

«CodeList»
GenericLogicalSpaceFunctionValue

«CodeList»
GenericOccupiedSpaceFunctionValue

«CodeList»
GenericUnoccupiedSpaceFunctionValue

«CodeList»
GenericThematicSurfaceFunctionValue

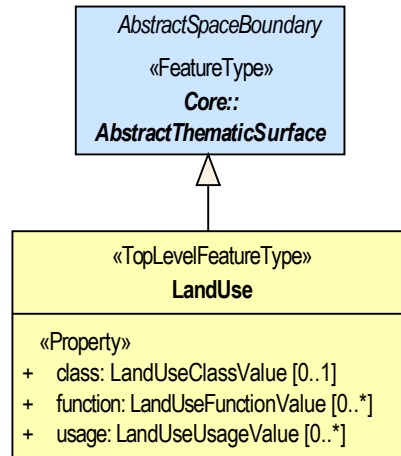
«CodeList»
GenericLogicalSpaceUsageValue

«CodeList»
GenericOccupiedSpaceUsageValue

«CodeList»
GenericUnoccupiedSpaceUsageValue

«CodeList»
GenericThematicSurfaceUsageValue

LandUse module



LandUse module - Code lists

«CodeList»

LandUseClassValue

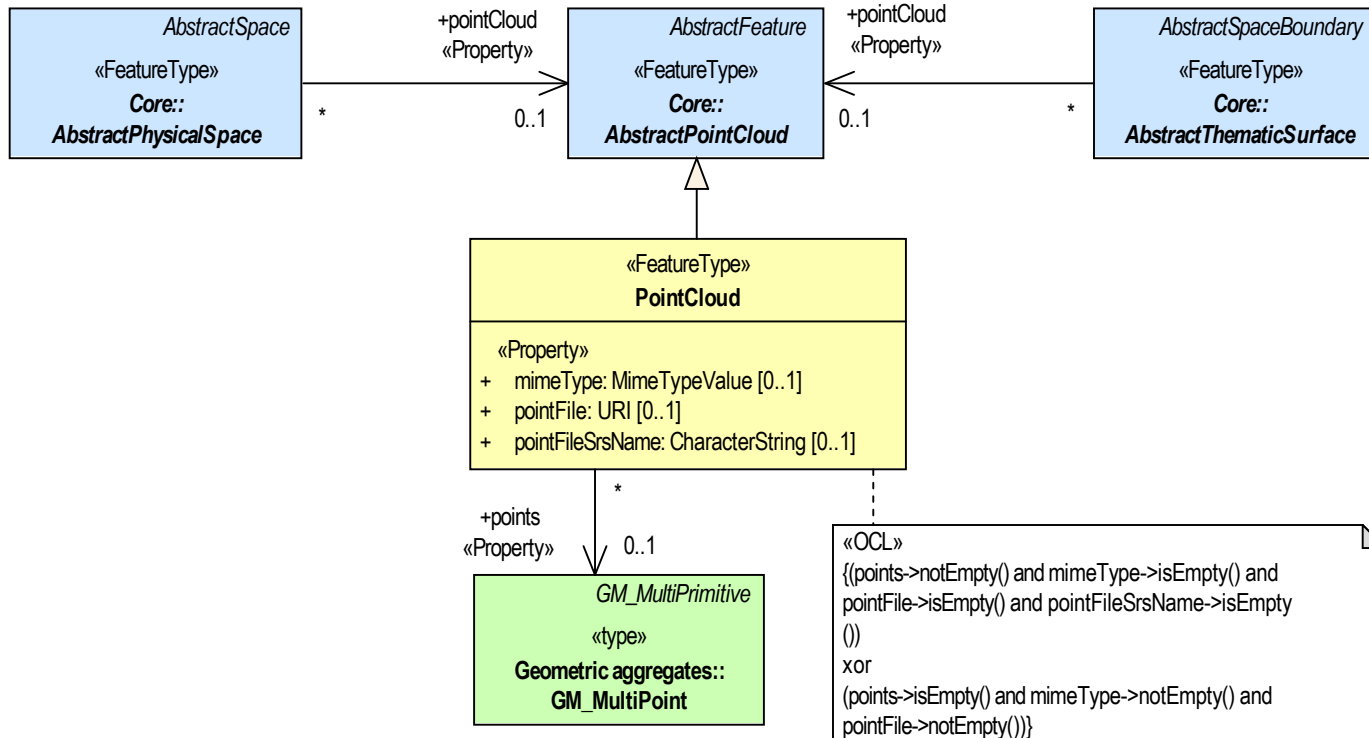
«CodeList»

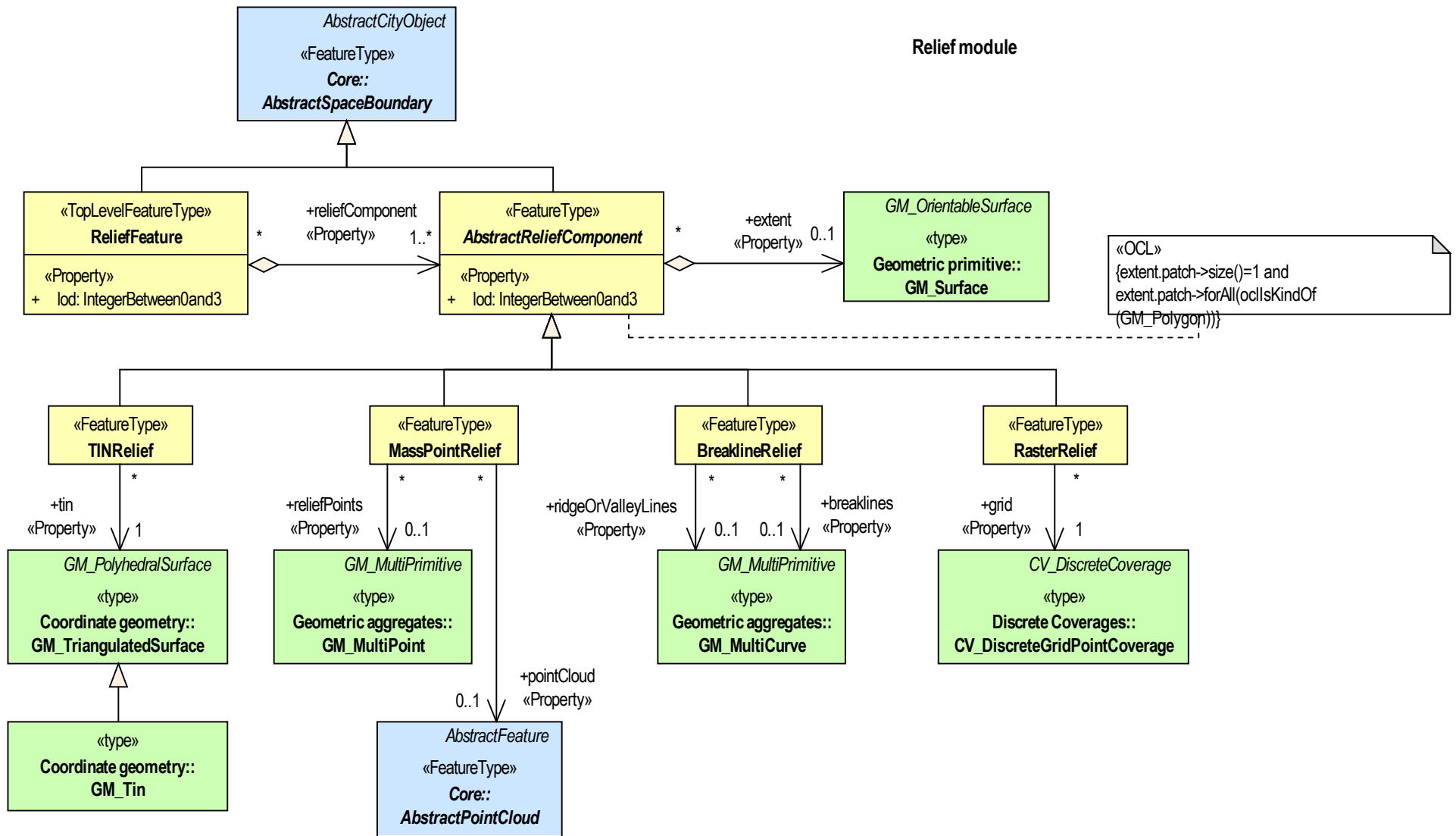
LandUseFunctionValue

«CodeList»

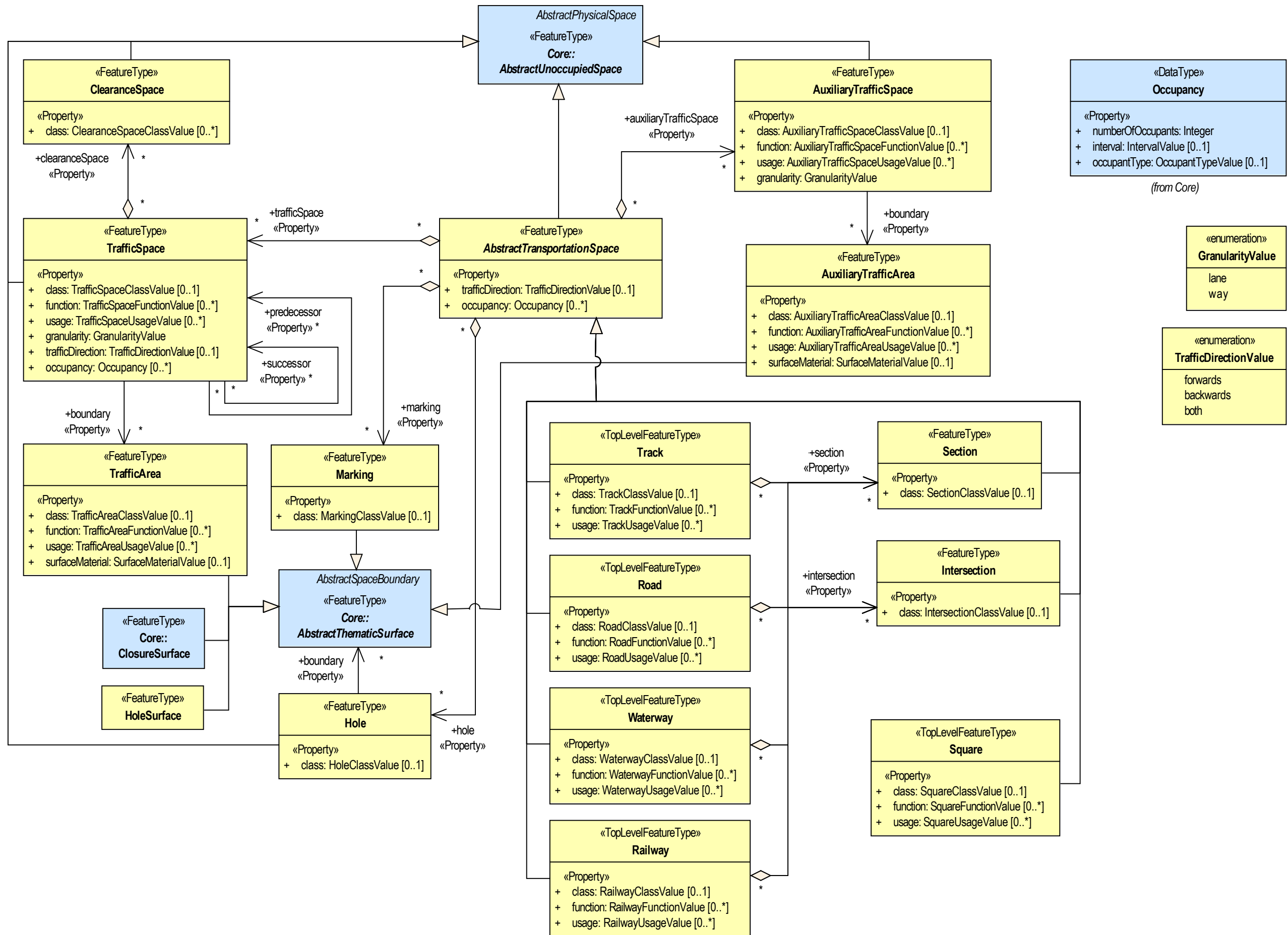
LandUseUsageValue

PointCloud module

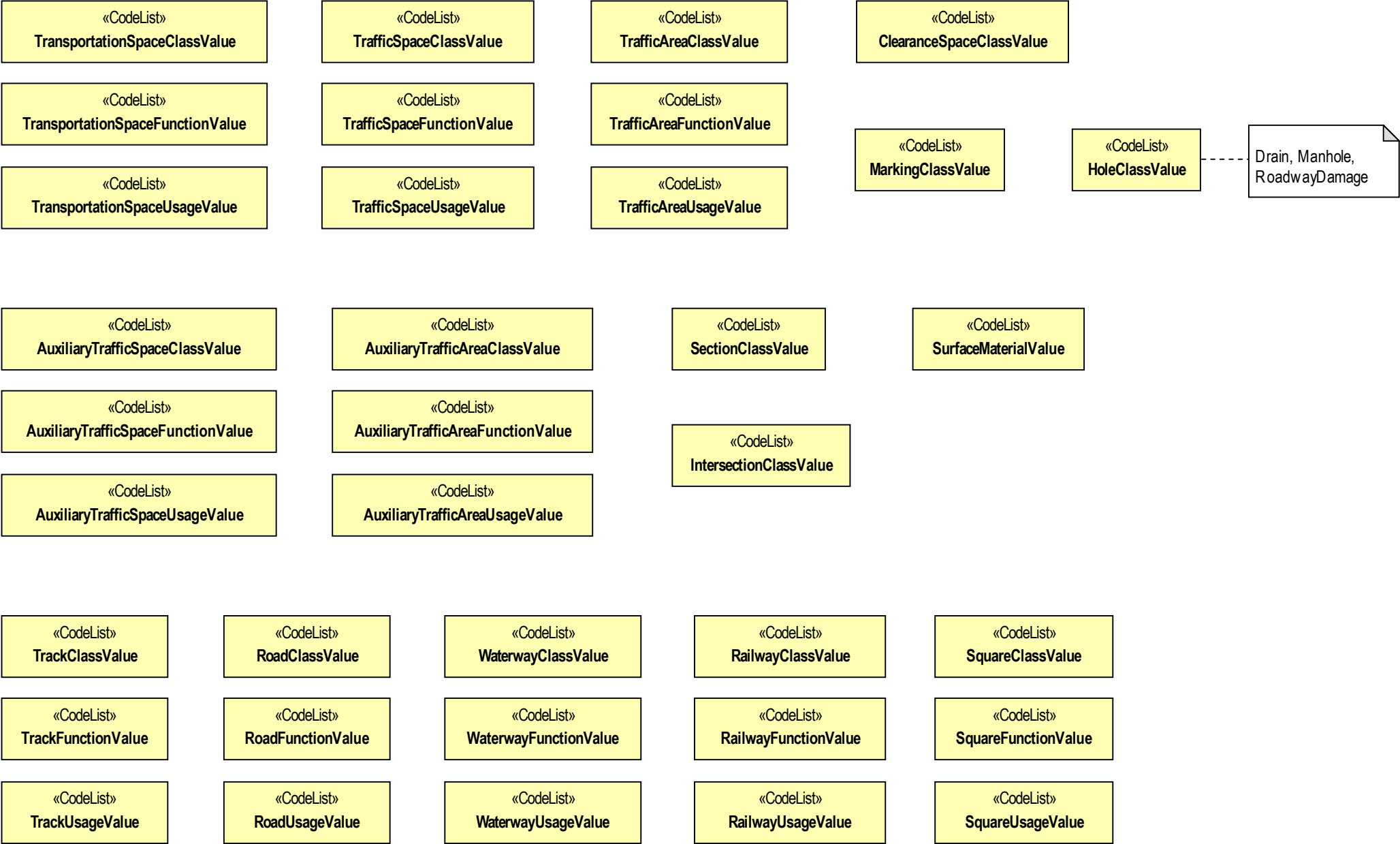




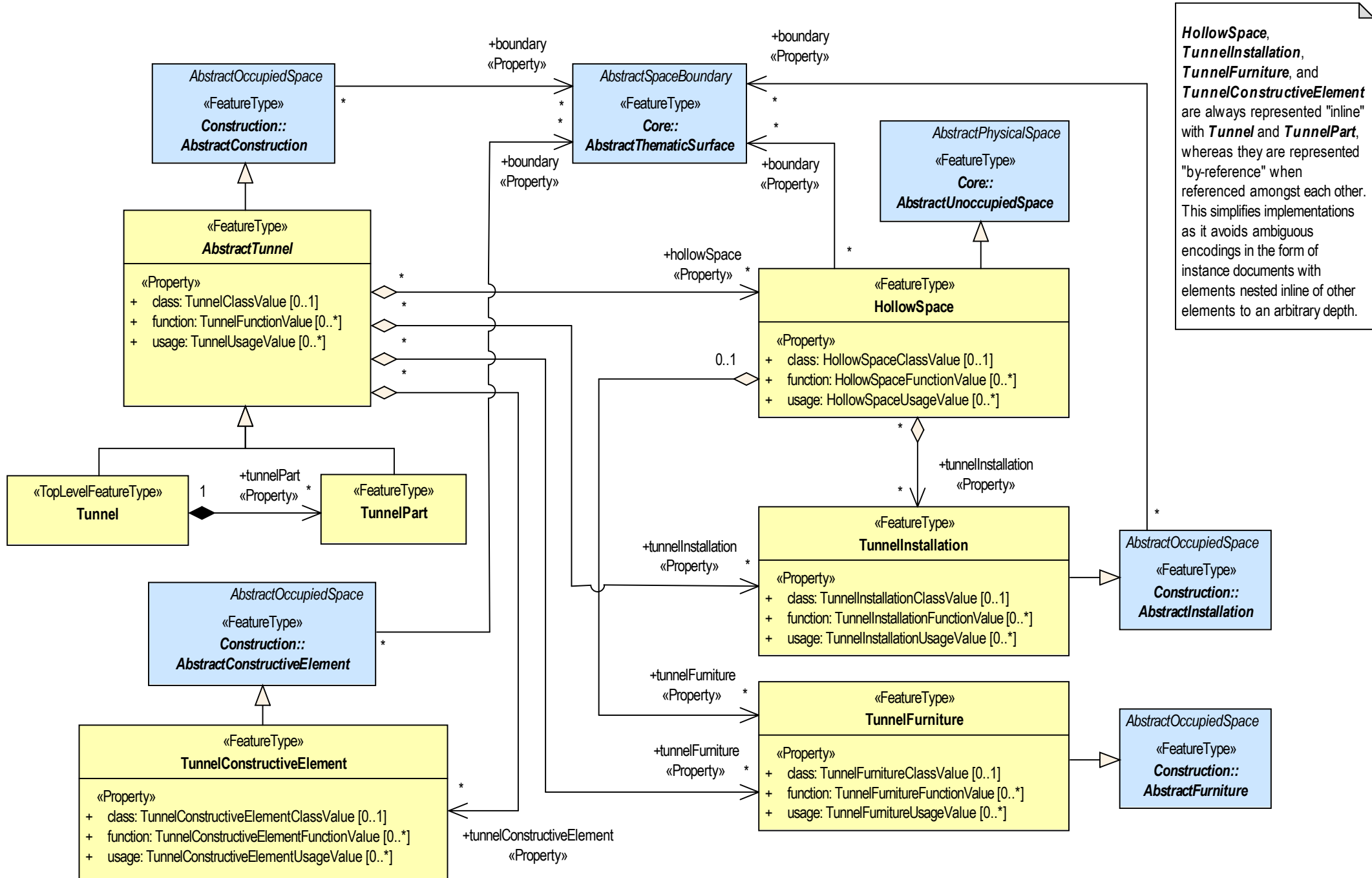
Transportation module



Transportation module - Code lists

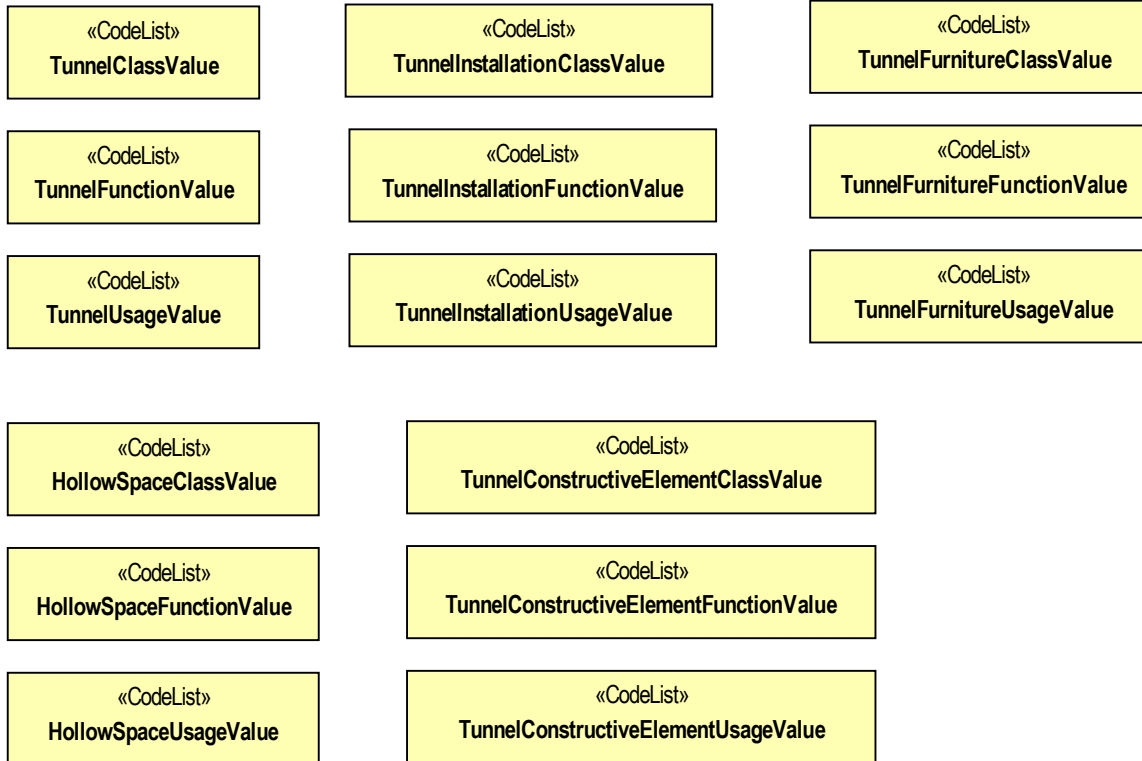


Tunnel module

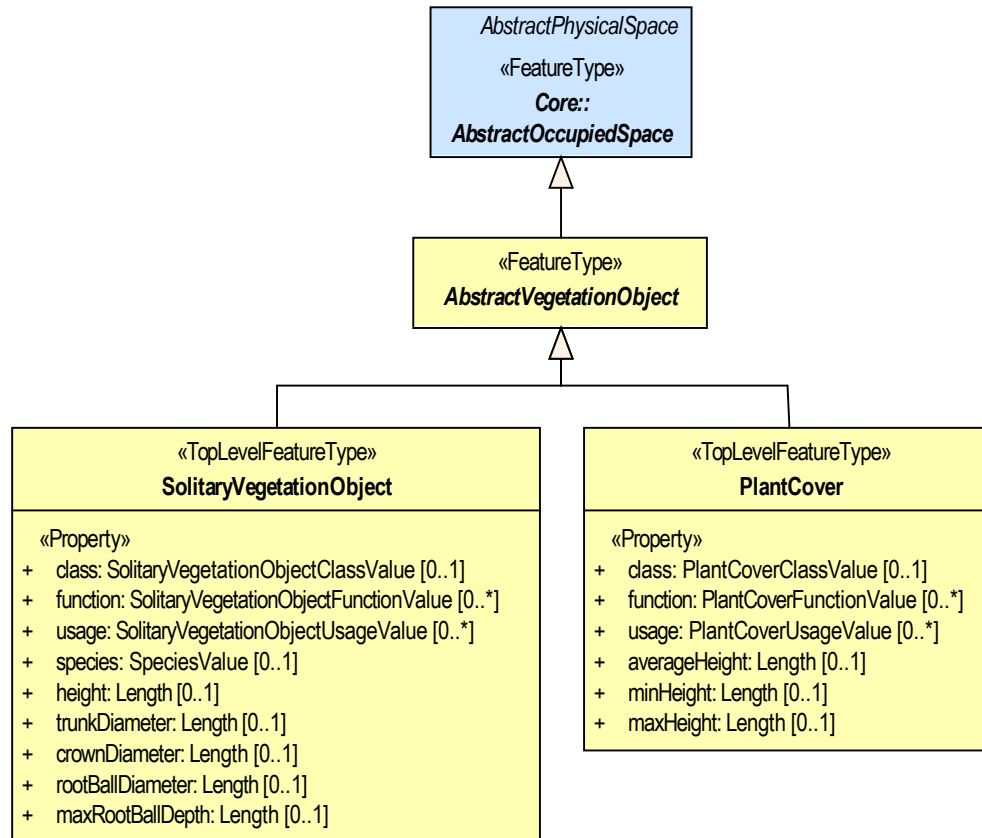


HollowSpace, **TunnellInstallation**, **TunnelFurniture**, and **TunnelConstructiveElement** are always represented "inline" with **Tunnel** and **TunnelPart**, whereas they are represented "by-reference" when referenced amongst each other. This simplifies implementations as it avoids ambiguous encodings in the form of instance documents with elements nested inline of other elements to an arbitrary depth.

Tunnel module - Code lists



Vegetation module



Vegetation module - Code lists

«CodeList»
SolitaryVegetationObjectClassValue

«CodeList»
SolitaryVegetationObjectFunctionValue

«CodeList»
SolitaryVegetationObjectUsageValue

«CodeList»
PlantCoverClassValue

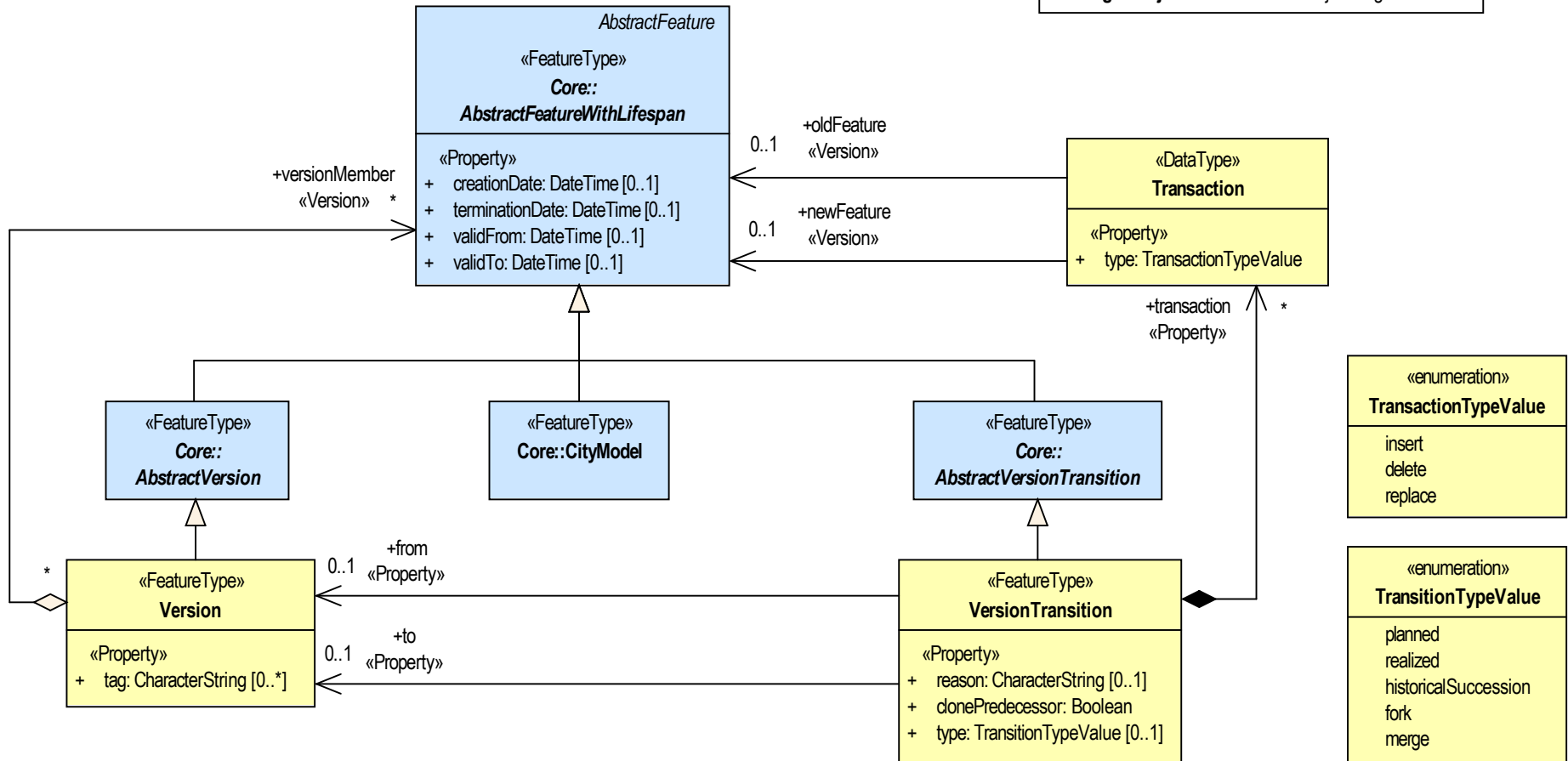
«CodeList»
PlantCoverFunctionValue

«CodeList»
PlantCoverUsageValue

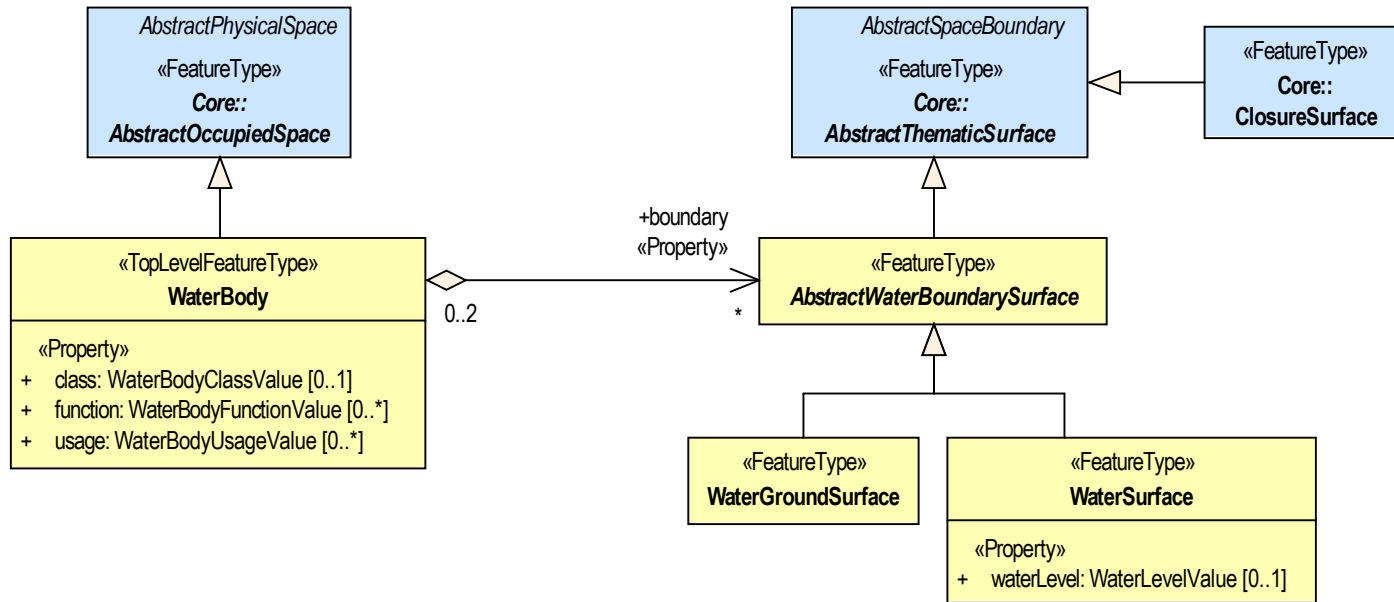
«CodeList»
SpeciesValue

Versioning module

The stereotype **«Version»** is adopted from INSPIRE. The stereotype is used for association roles to express that the **association refers to a specific version of the target object** and not to the object in general.



WaterBody module



WaterBody module - Code lists

«CodeList»
WaterBodyClassValue

«CodeList»
WaterBodyFunctionValue

«CodeList»
WaterBodyUsageValue

«CodeList»
WaterLevelValue