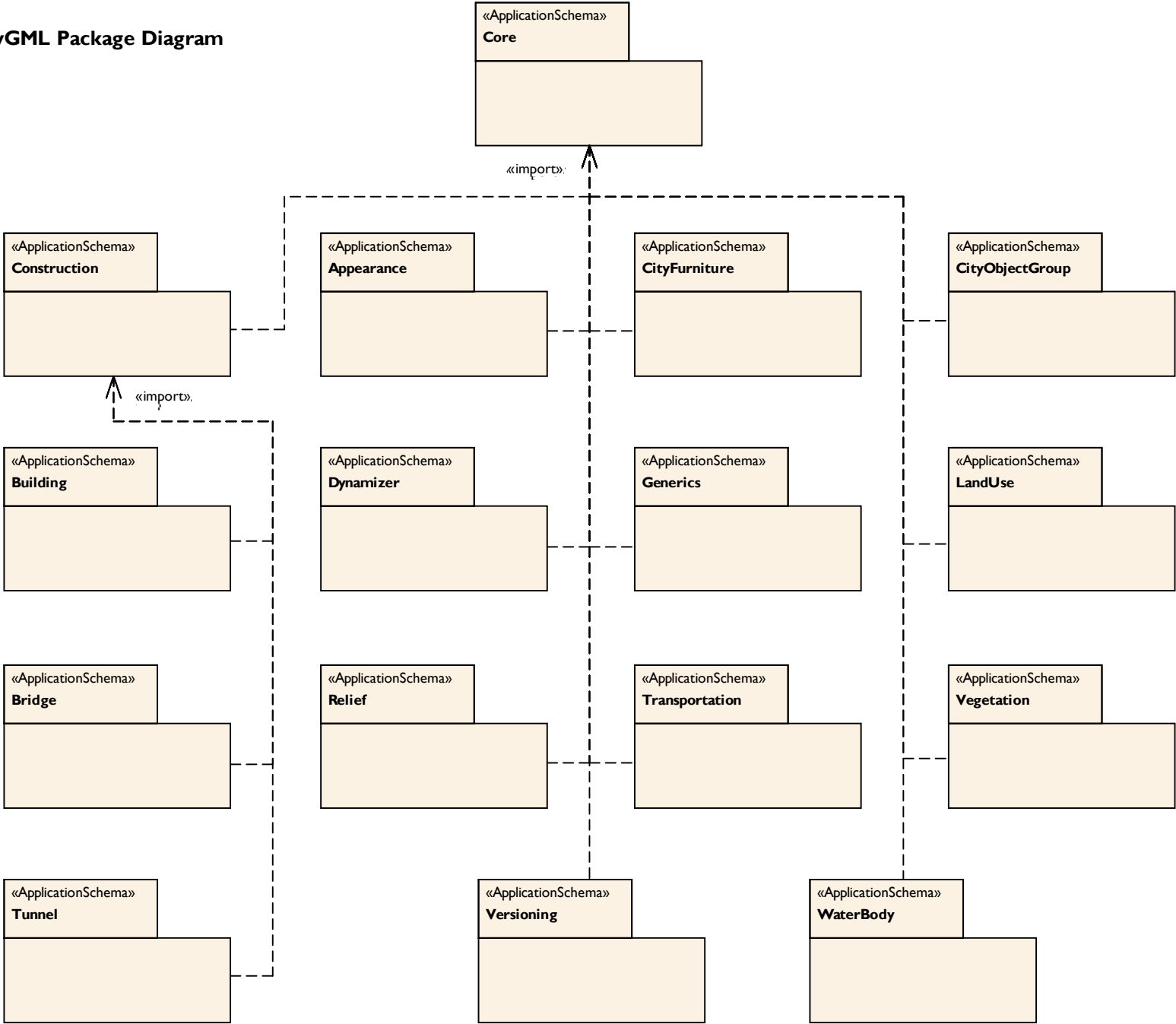
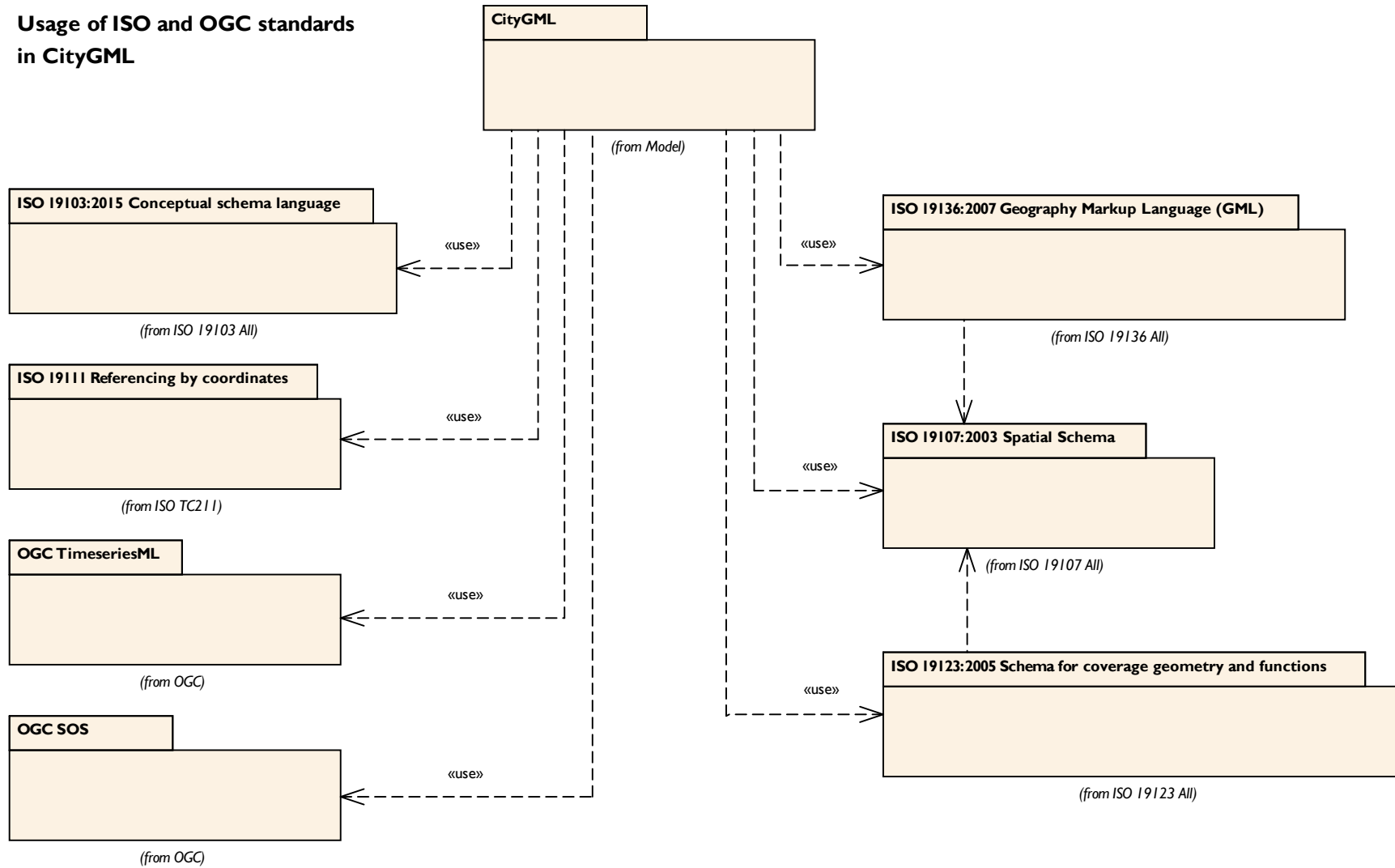


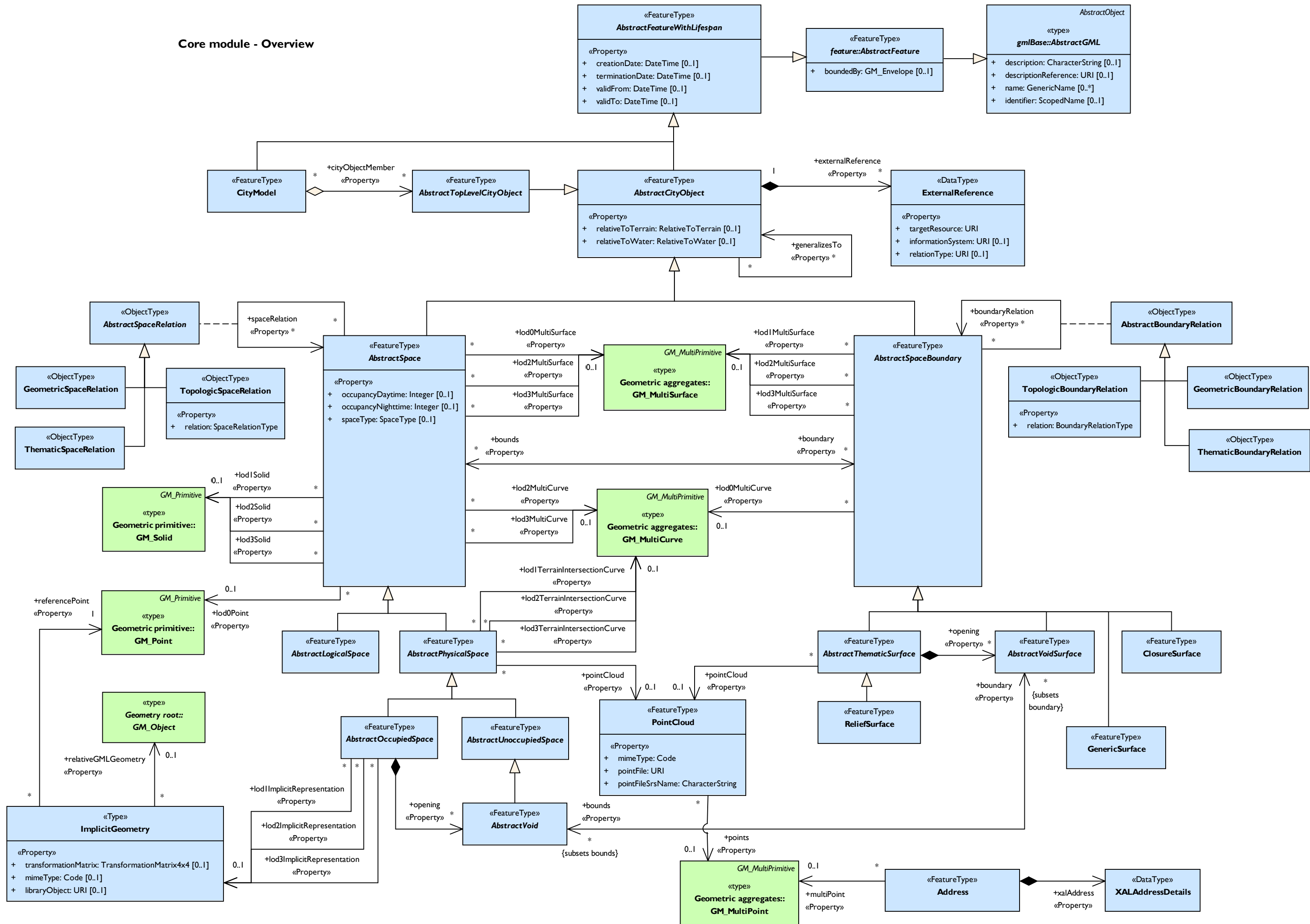
CityGML Package Diagram



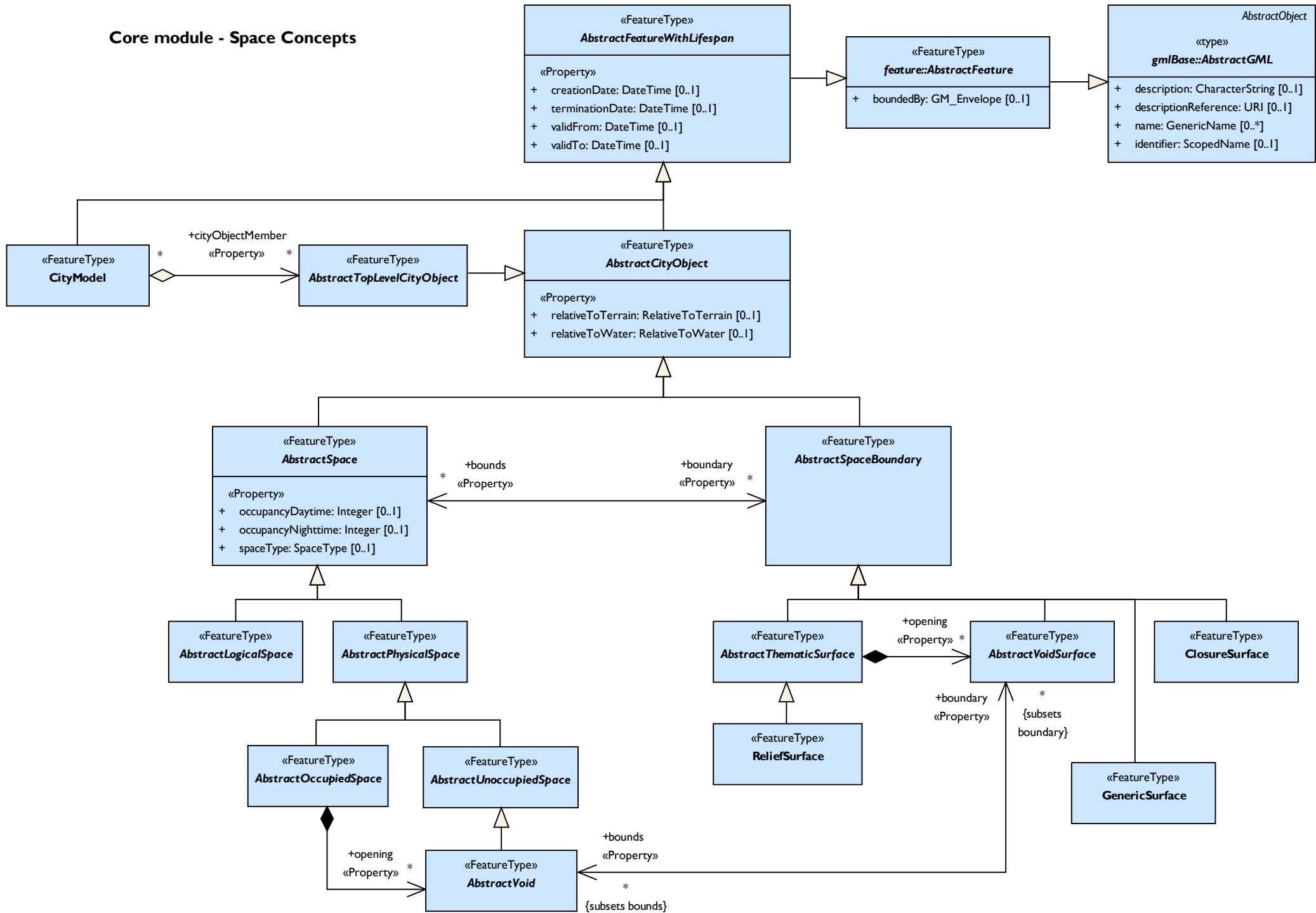
Usage of ISO and OGC standards in CityGML



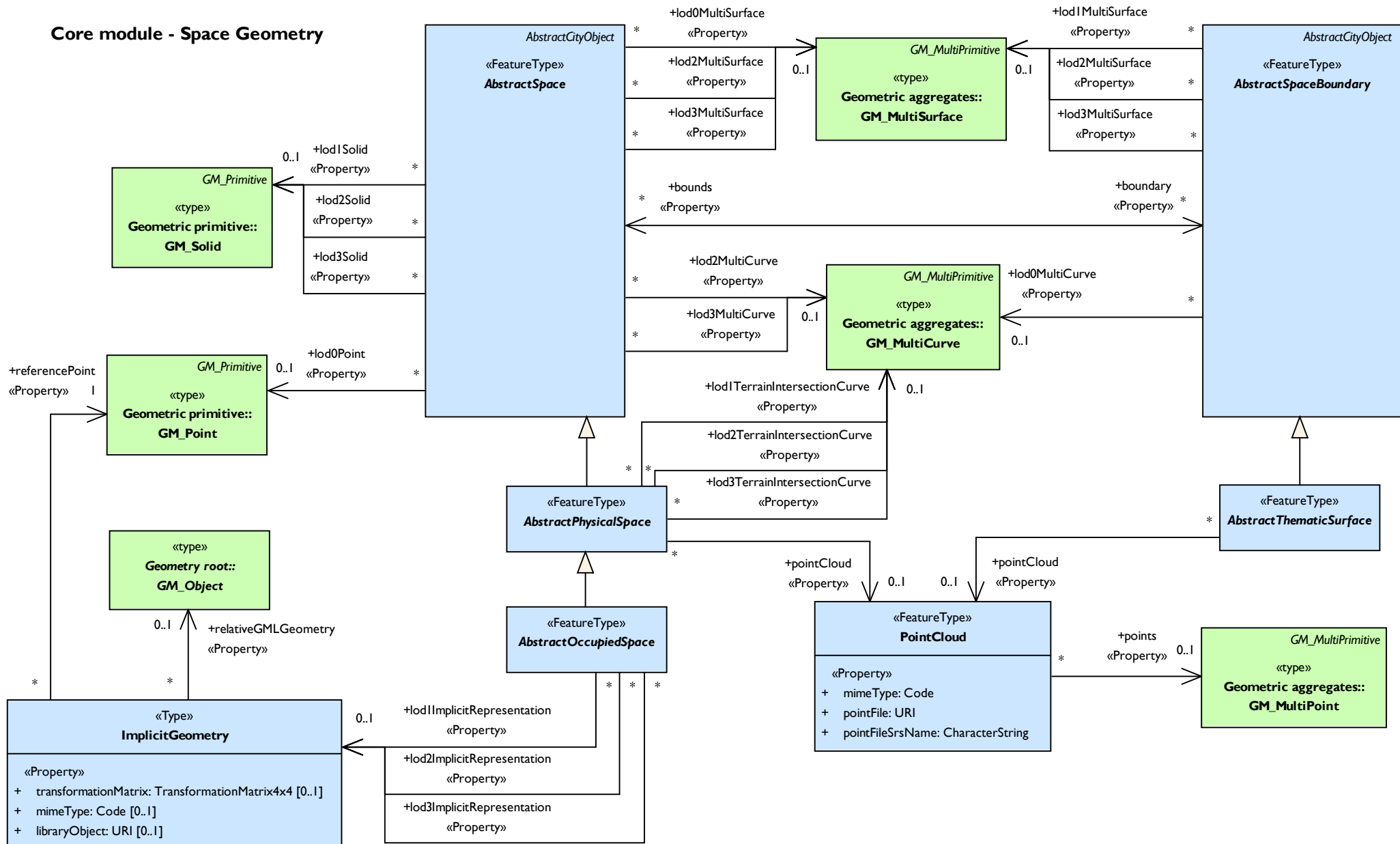
Core module - Overview



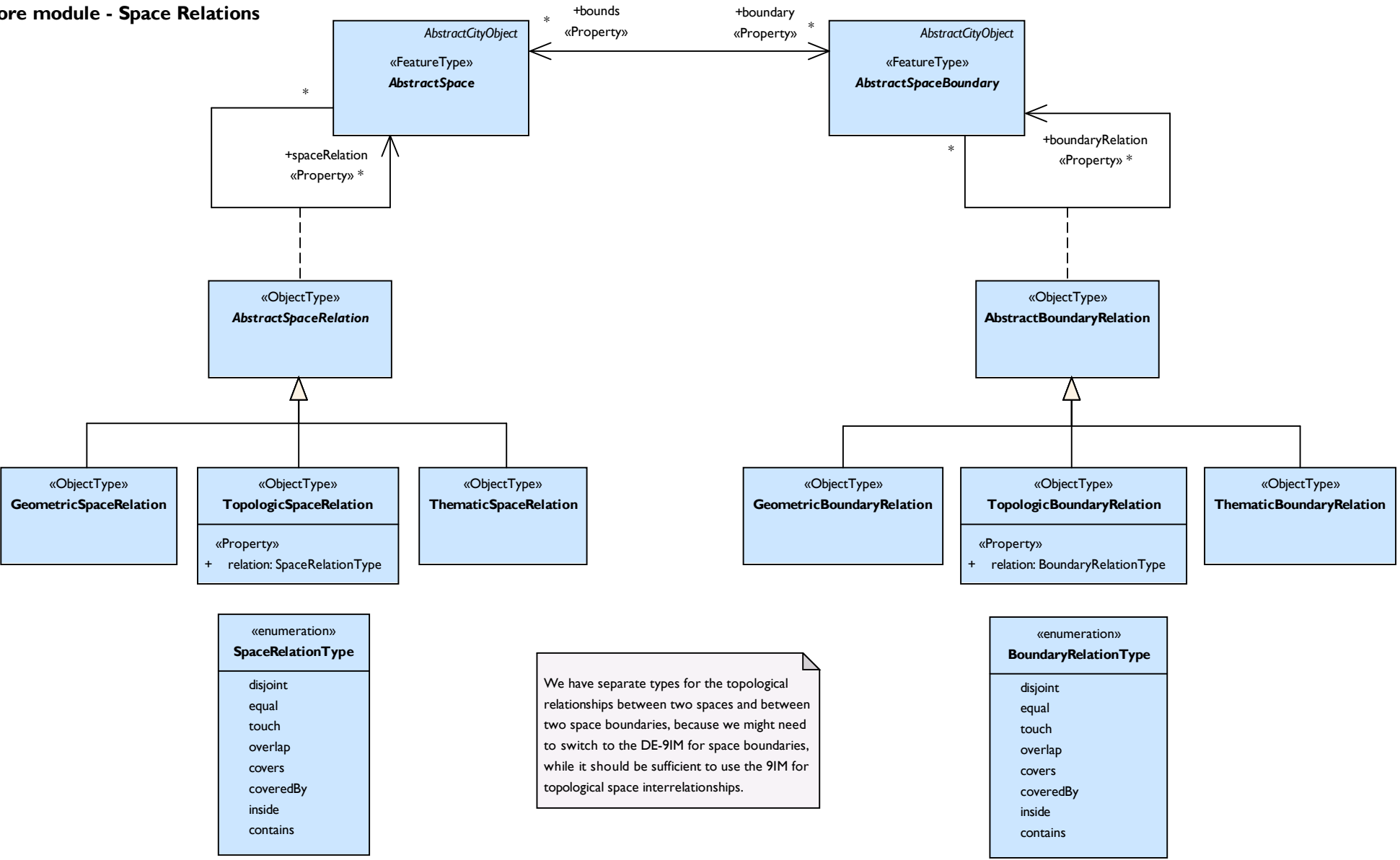
Core module - Space Concepts



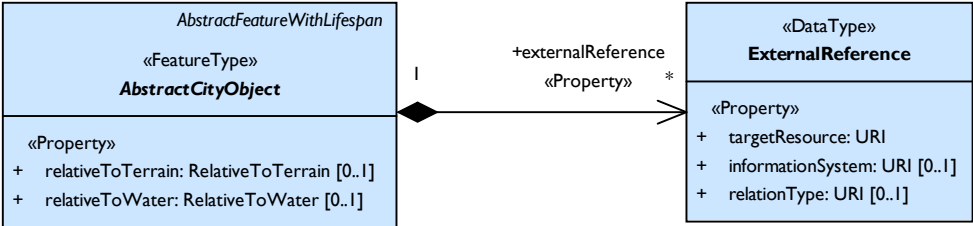
Core module - Space Geometry



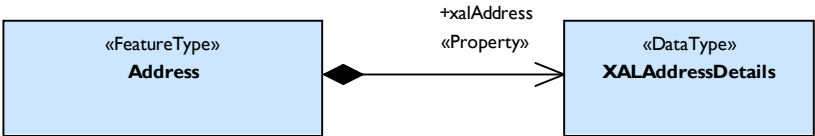
Core module - Space Relations



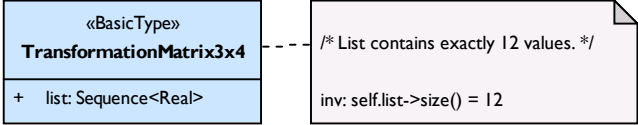
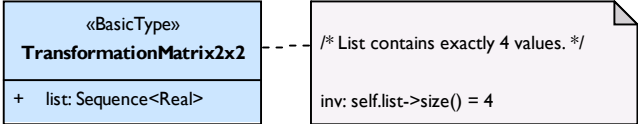
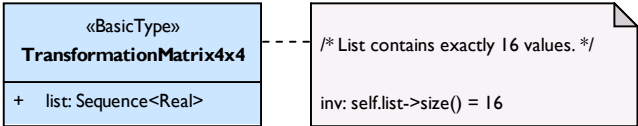
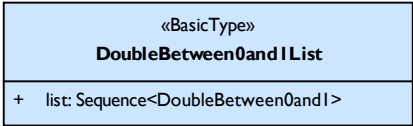
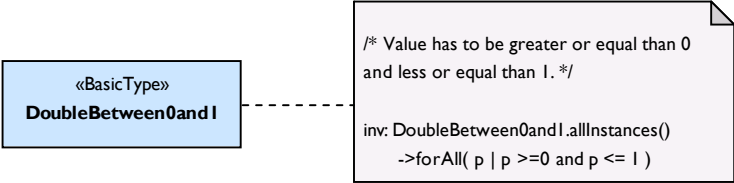
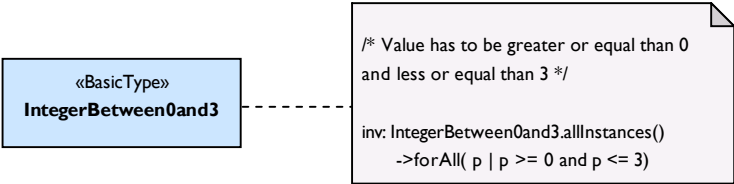
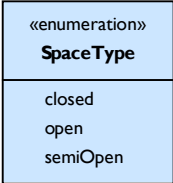
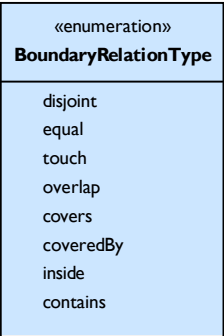
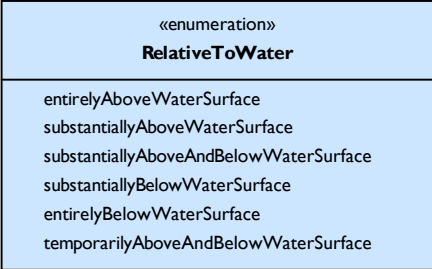
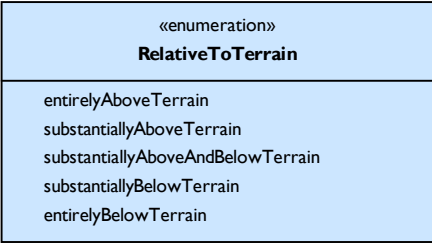
Core module - Miscellaneous



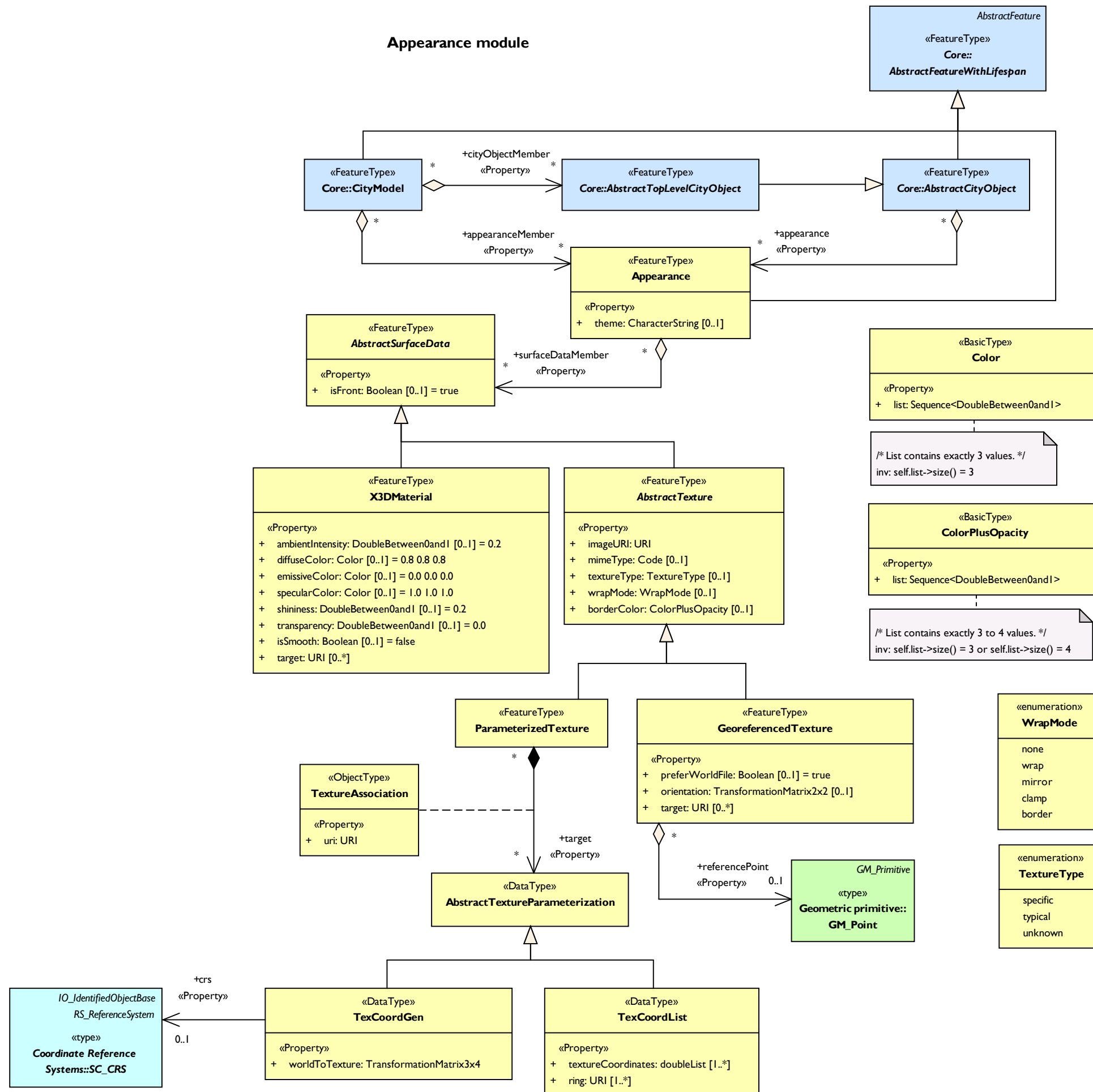
ExternalReference is now extended by an optional **relationType** which can link to some external definition of the type of relation (e.g. the *sameAs* relation from OWL). Hence, **ExternalReferences** can now be used to express relations similar to RDF.



Core module - Basic Types and Enumerations

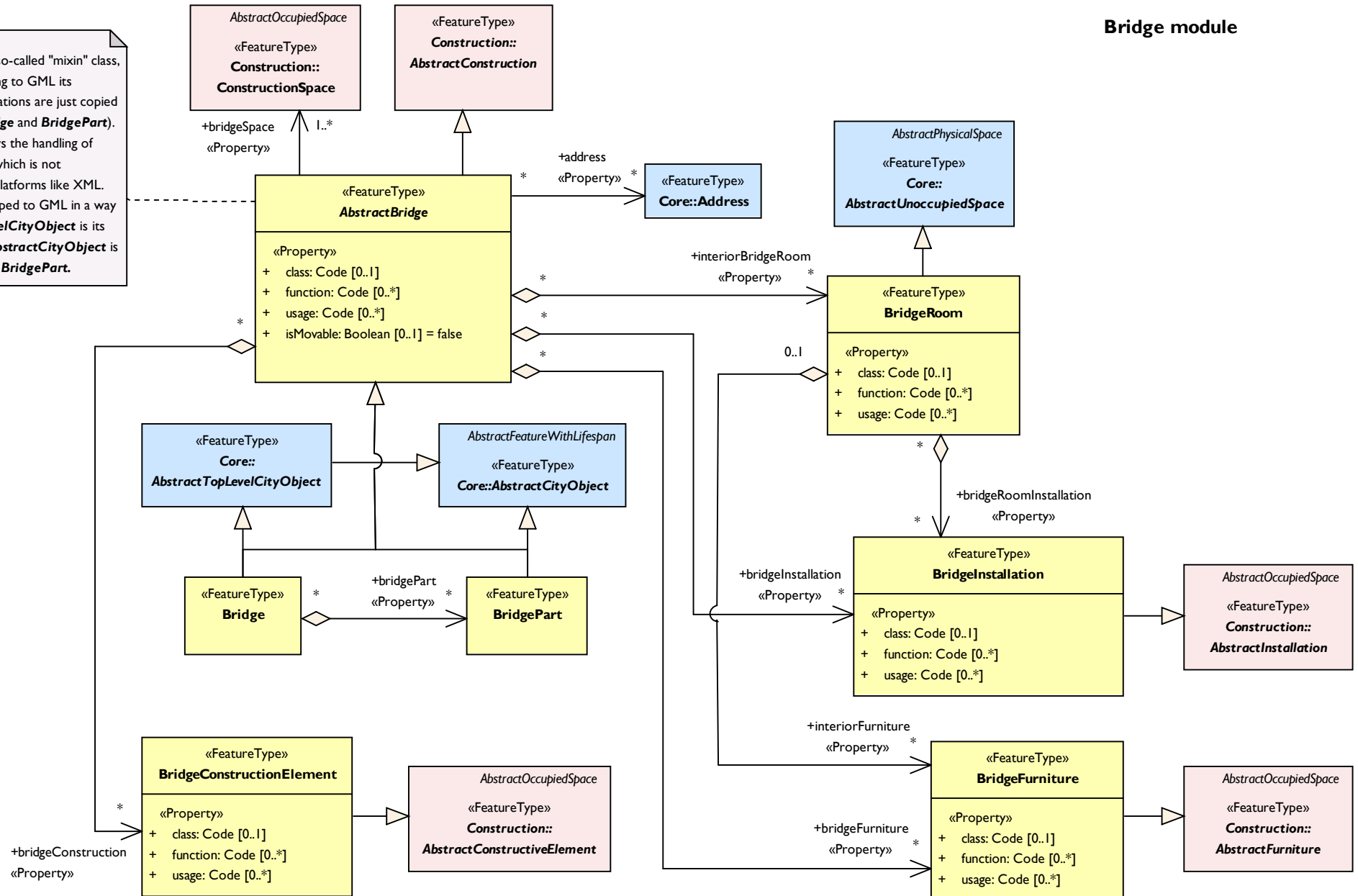


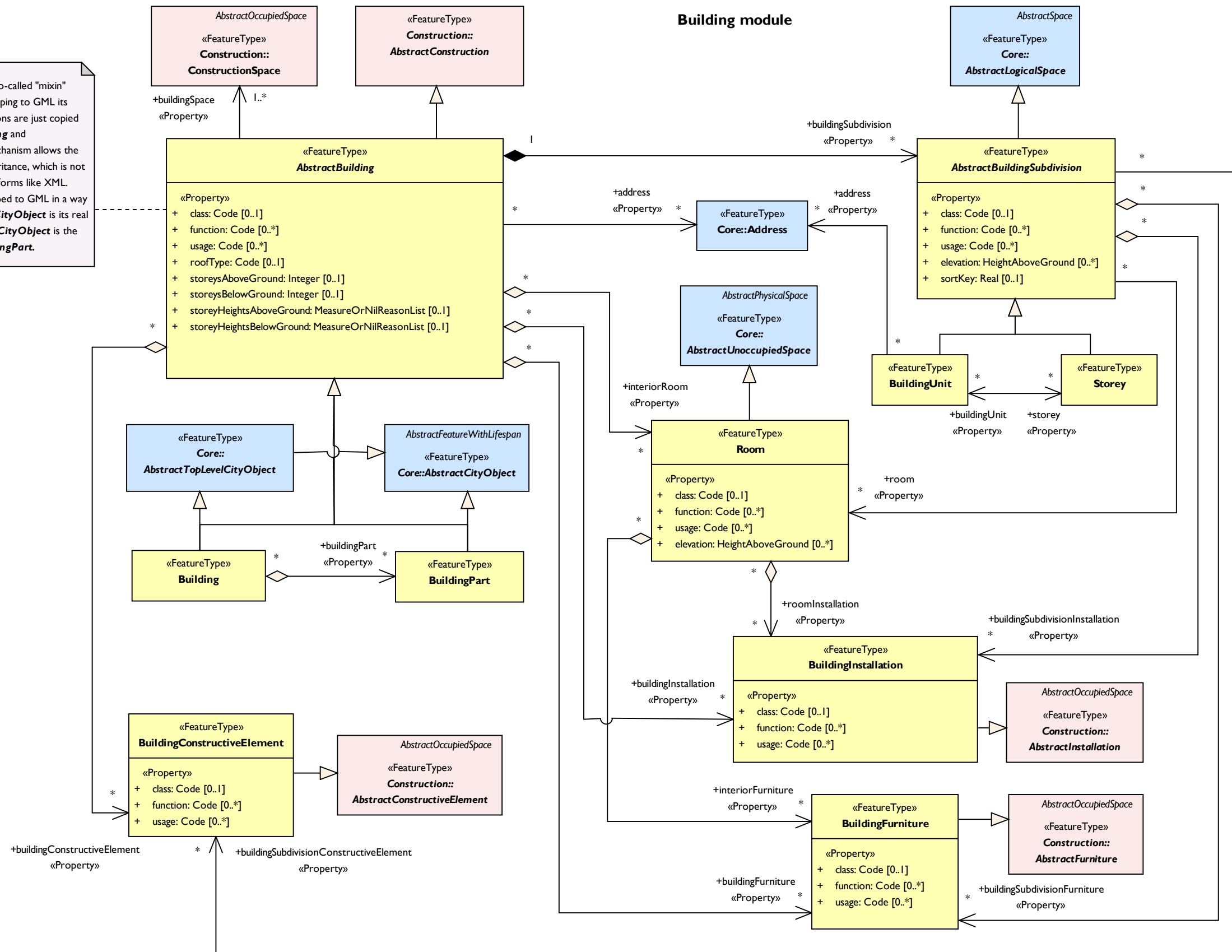
Appearance module



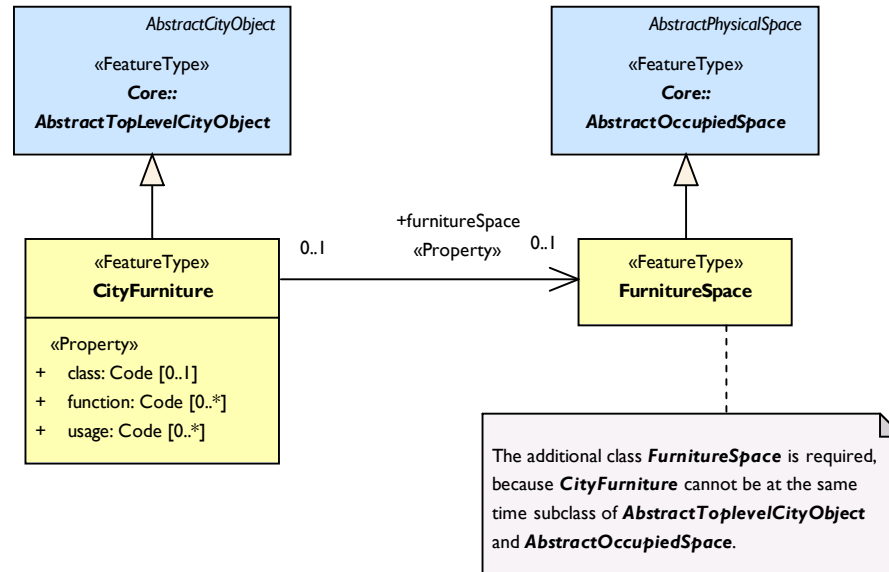
Bridge module

AbstractBridge is a so-called "mixin" class, i. e. during the mapping to GML its properties and associations are just copied to its subclasses (**Bridge** and **BridgePart**). This mechanism allows the handling of multiple inheritance, which is not supported by some platforms like XML. Hence, **Bridge** is mapped to GML in a way that **AbstractTopLevelCityObject** is its real superclass and **AbstractCityObject** is the real superclass of **BridgePart**.

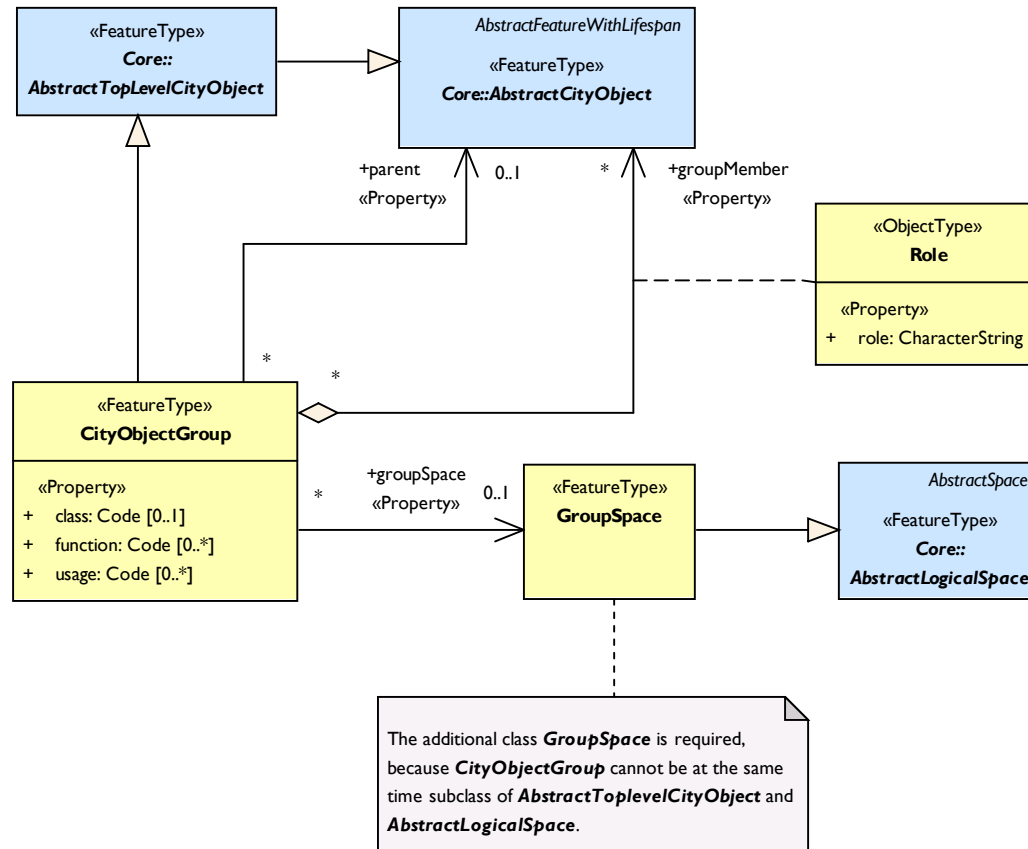




CityFurniture module

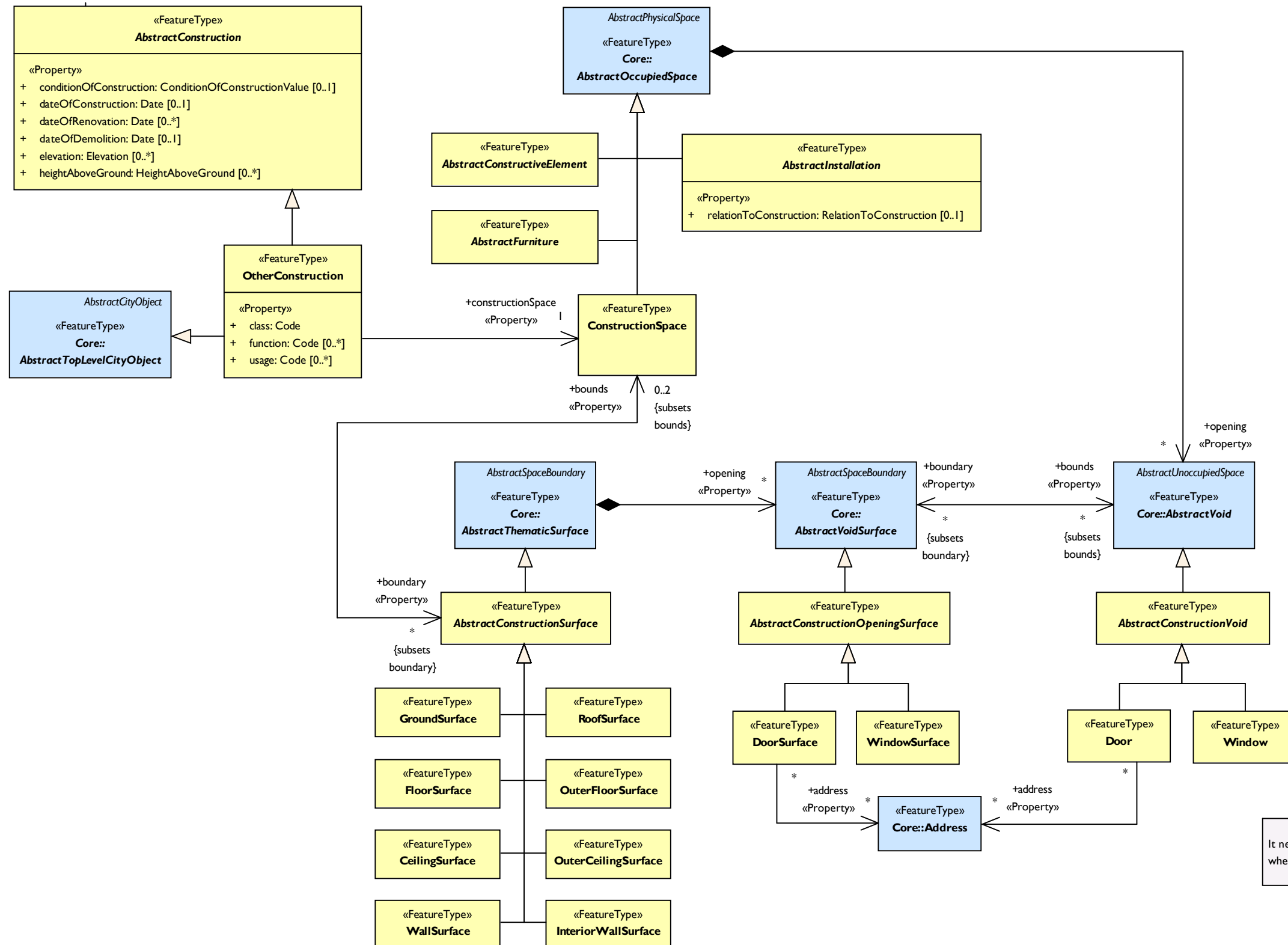


CityObjectGroup module



AbstractConstruction is a so-called "mixin" class, i. e. during the mapping to GML its properties and associations are just copied to its subclass **OtherConstruction**. This mechanism allows the handling of multiple inheritance, which is not supported by some platforms like XML. Hence, **OtherConstruction** is mapped to GML in a way that **AbstractTopLevelCityObject** is its real superclass.

Construction module



«CodeList» ConditionOfConstructionValue
+ declined + demolished + functional + projected + ruin + underConstruction

«enumeration» RelationToConstruction
inside outside bothInsideAndOutside

«DataType» Elevation
+ elevationReference: ElevationReferenceValue + elevationValue: DirectPosition

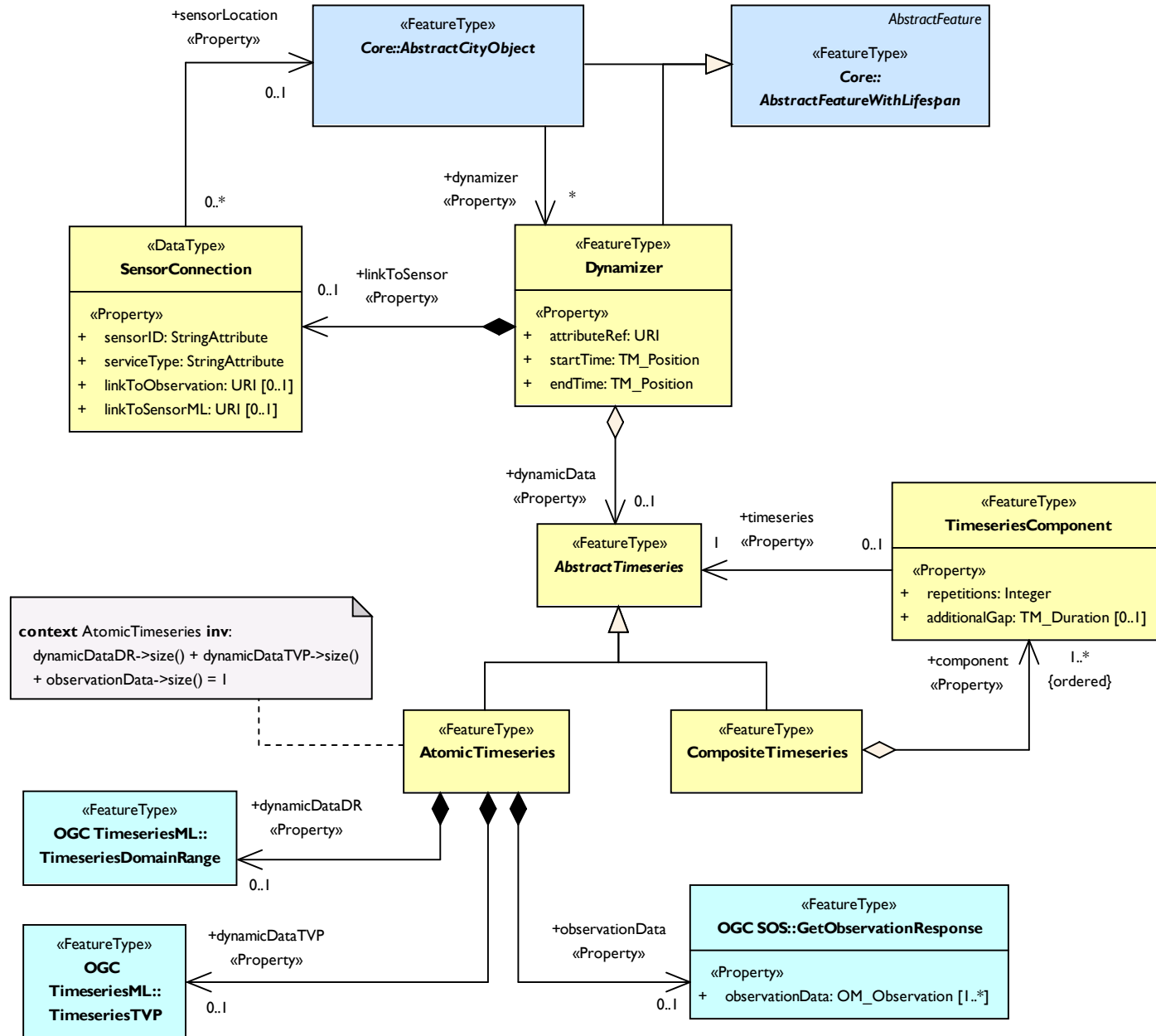
«DataType» HeightAboveGround
+ heightReference: ElevationReferenceValue + lowReference: ElevationReferenceValue + status: HeightStatusValue + value: Length

«CodeList» ElevationReferenceValue
+ aboveGroundEnvelope + bottomOfConstruction + entrancePoint + generalEave + generalGround + generalRoof + generalRoofEdge + highestEave + highestGroundPoint + highestPoint + highestRoofEdge + lowestEave + lowestFloorAboveGround + lowestGroundPoint + lowestRoofEdge + topOfConstruction

«CodeList» HeightStatusValue
+ estimated + measured

It needs to be discussed further within the SWG, whether a **Door** is a **Void** or an **Opening**,

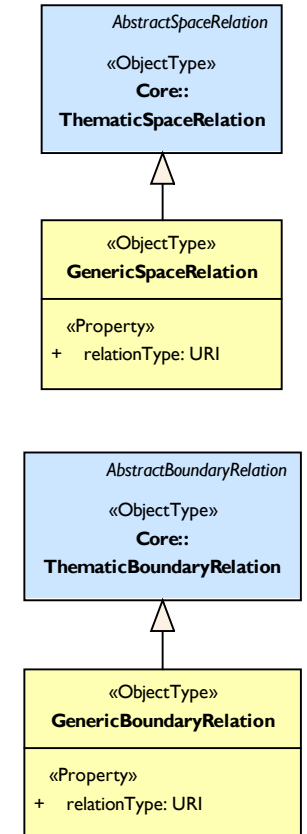
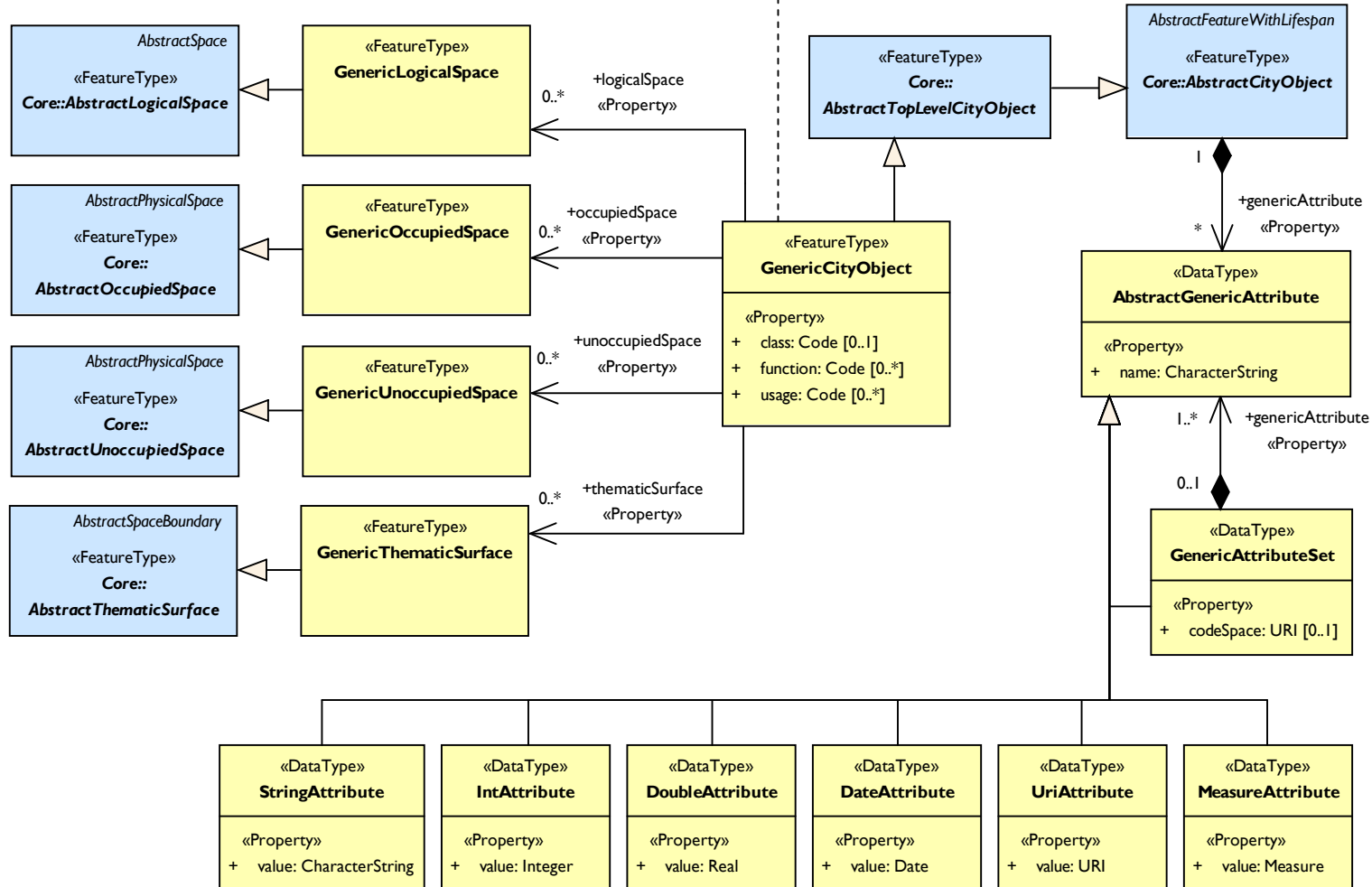
Dynamizer module



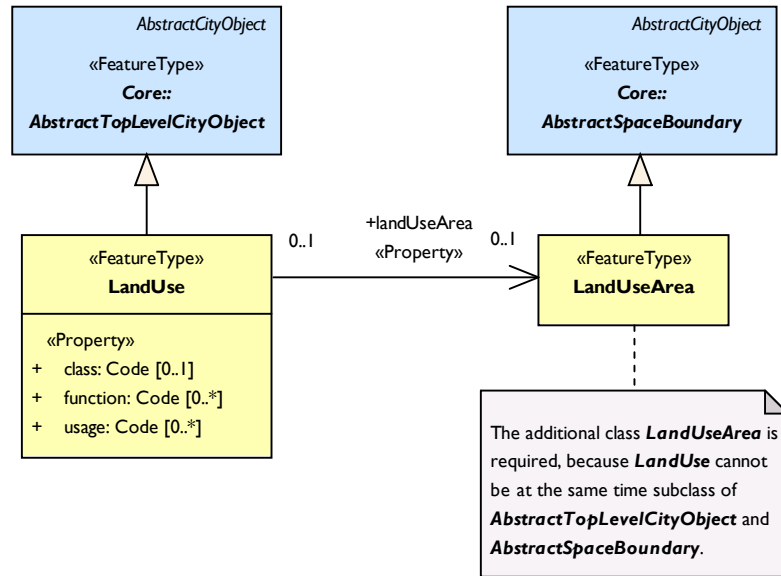
An instance of **GenericCityObject** can only be associated either with one or several instances of one of the Space classes or with one or several instances of **GenericThematicSurface**.

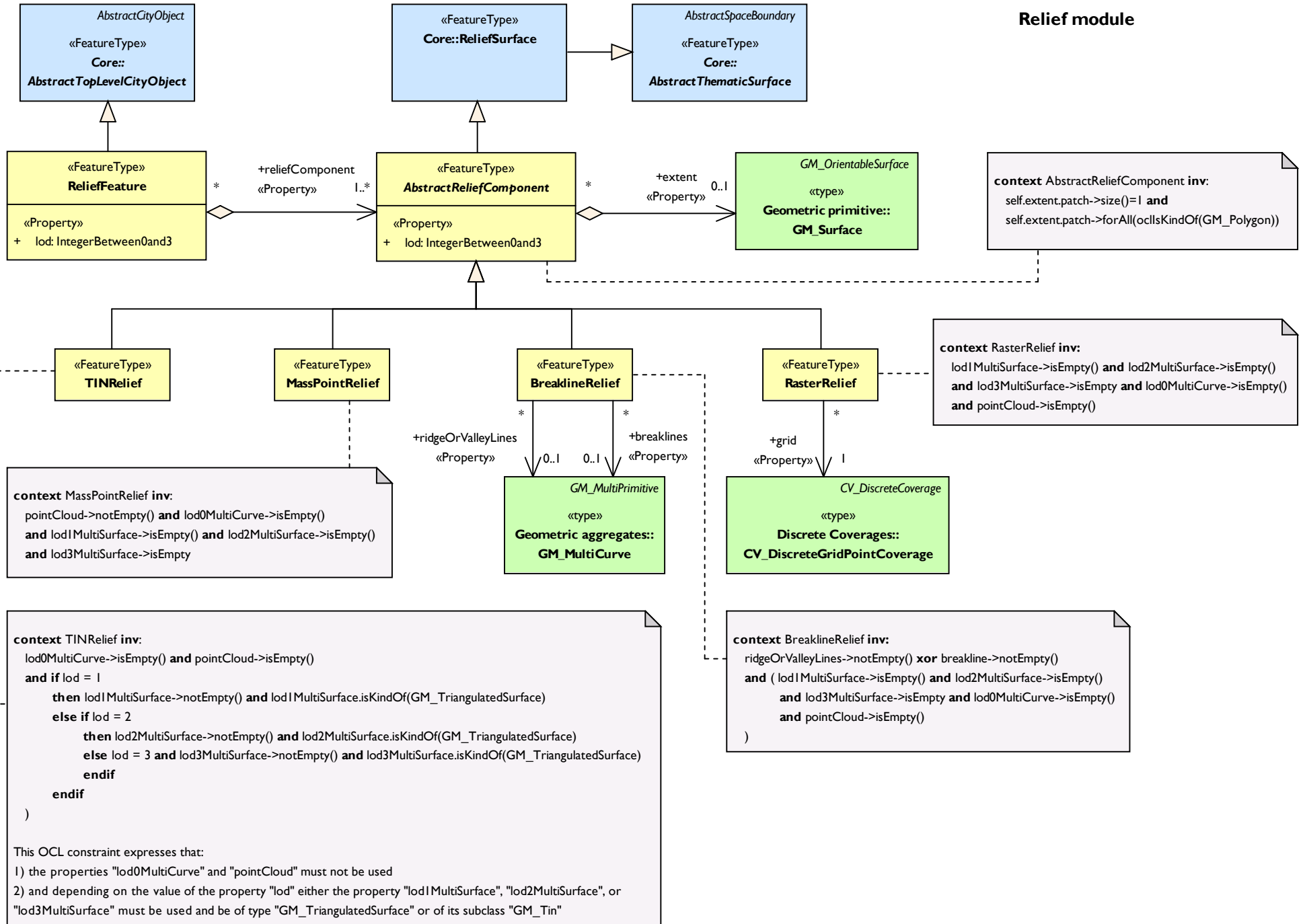
Generics module

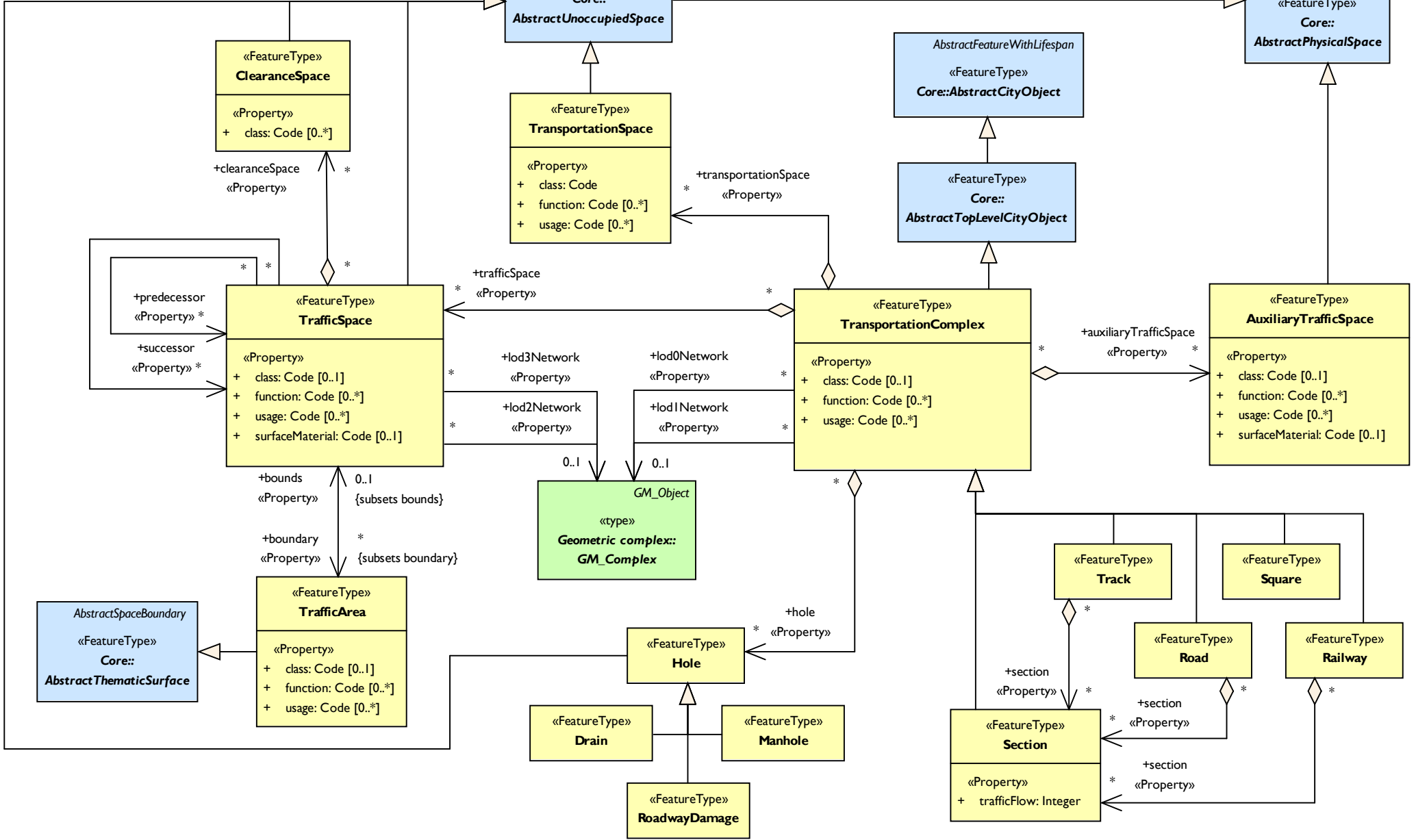
The **GenericSpaceRelation** and **GenericBoundaryRelation** can be used to represent an arbitrary relationship between two spaces or two space boundaries respectively. The relation type is expressed by an URI. Such relations could directly be mapped to RDF triples, where the source is pointing to the source space object, the target to the target space object, and the relation type is pointing to a definition of a relationship type in the Web.



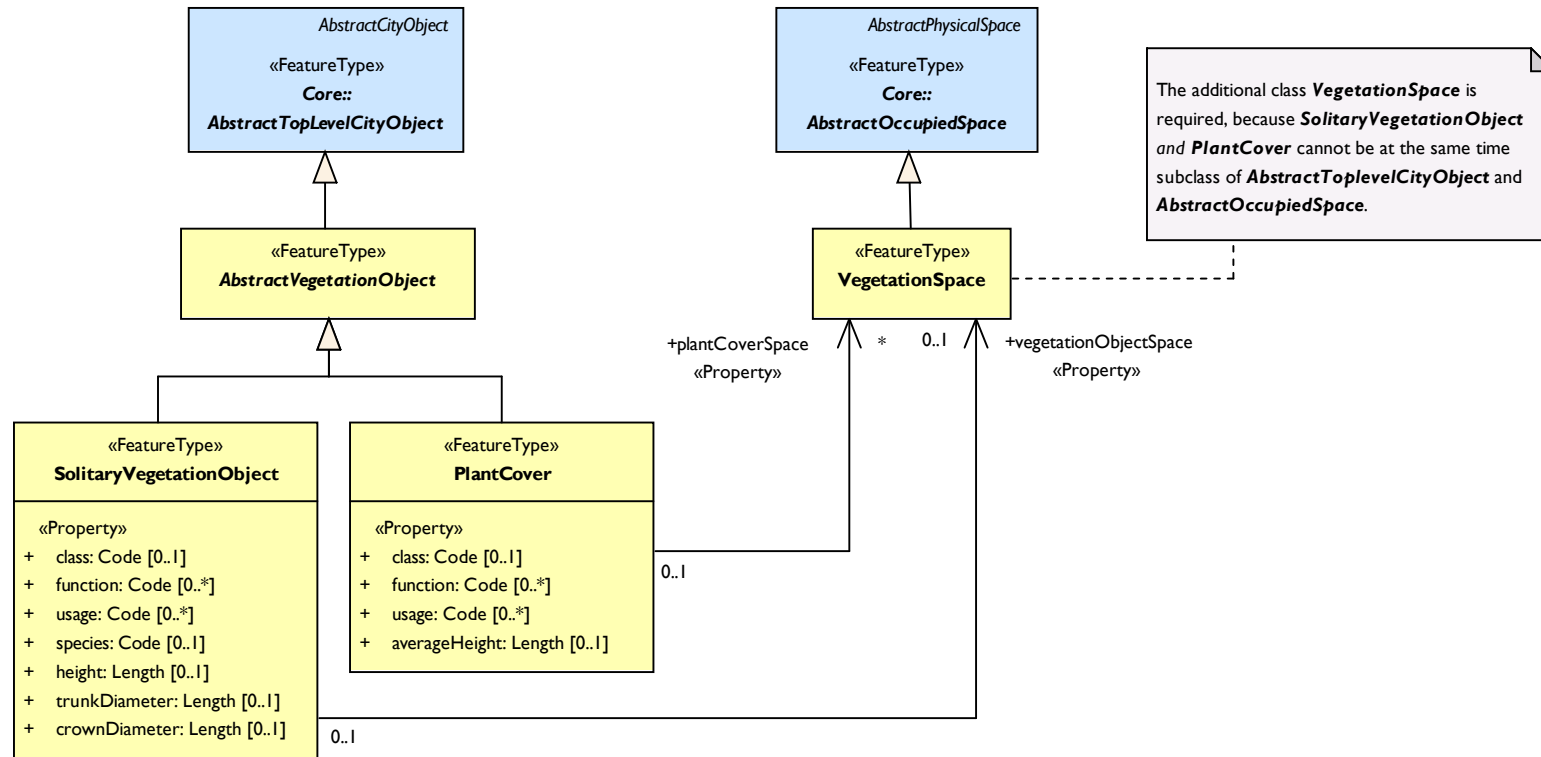
LandUse module





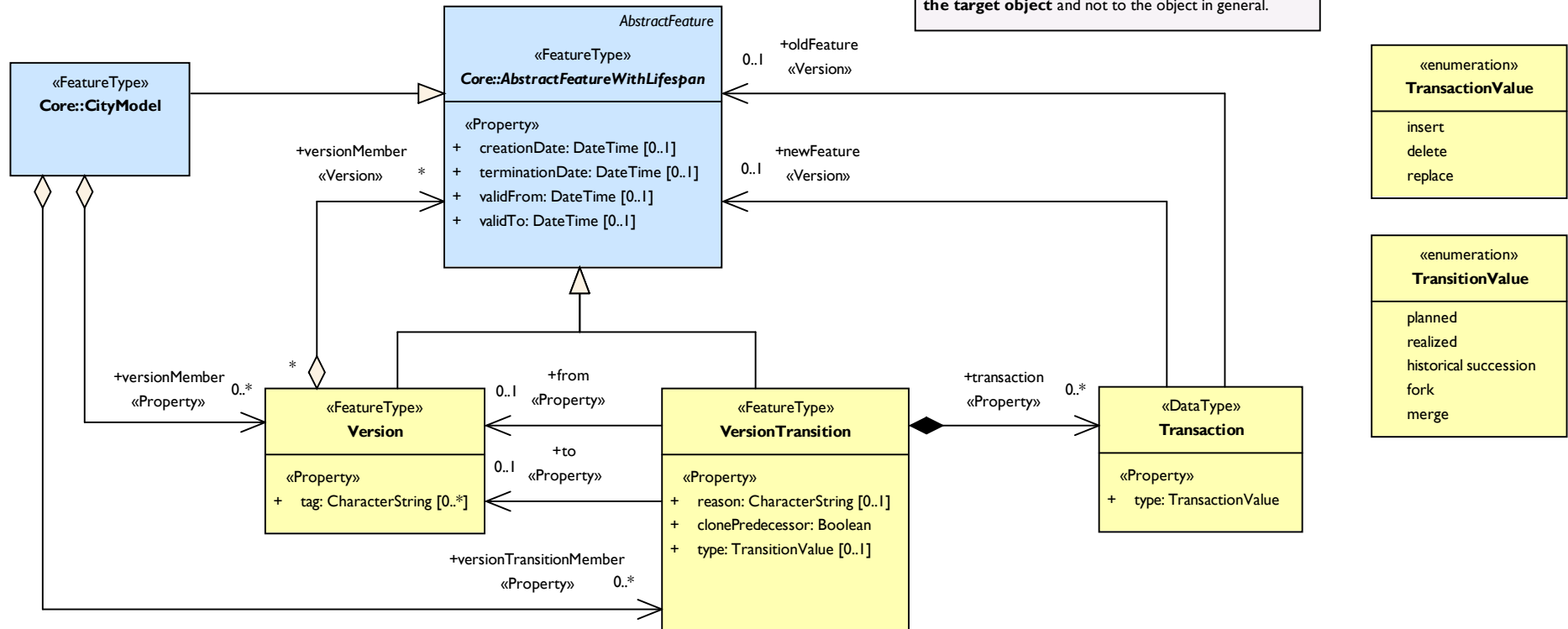
[illegible]

Vegetation module



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

