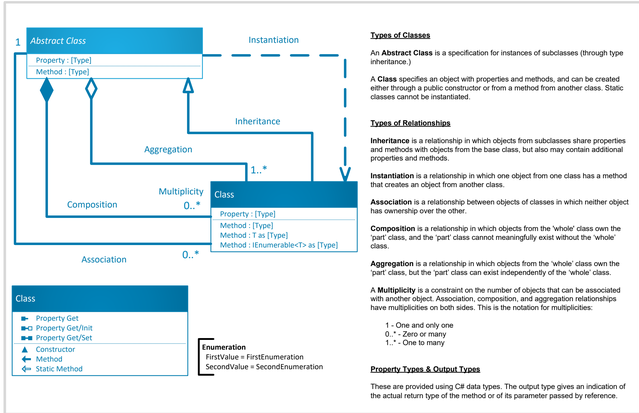
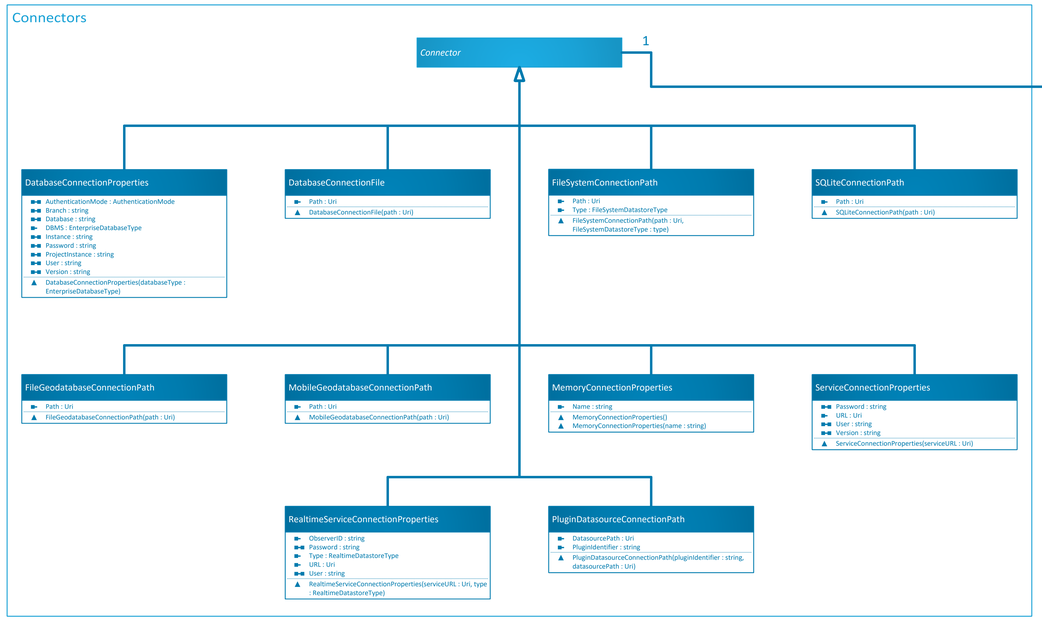
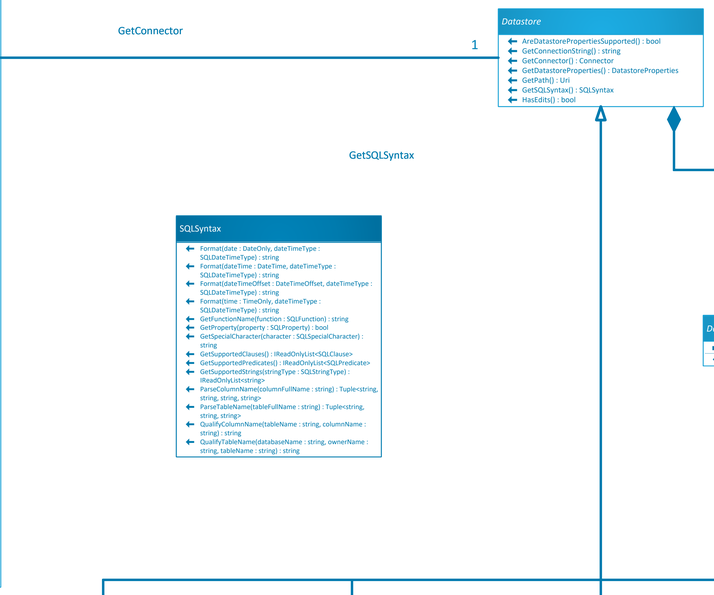
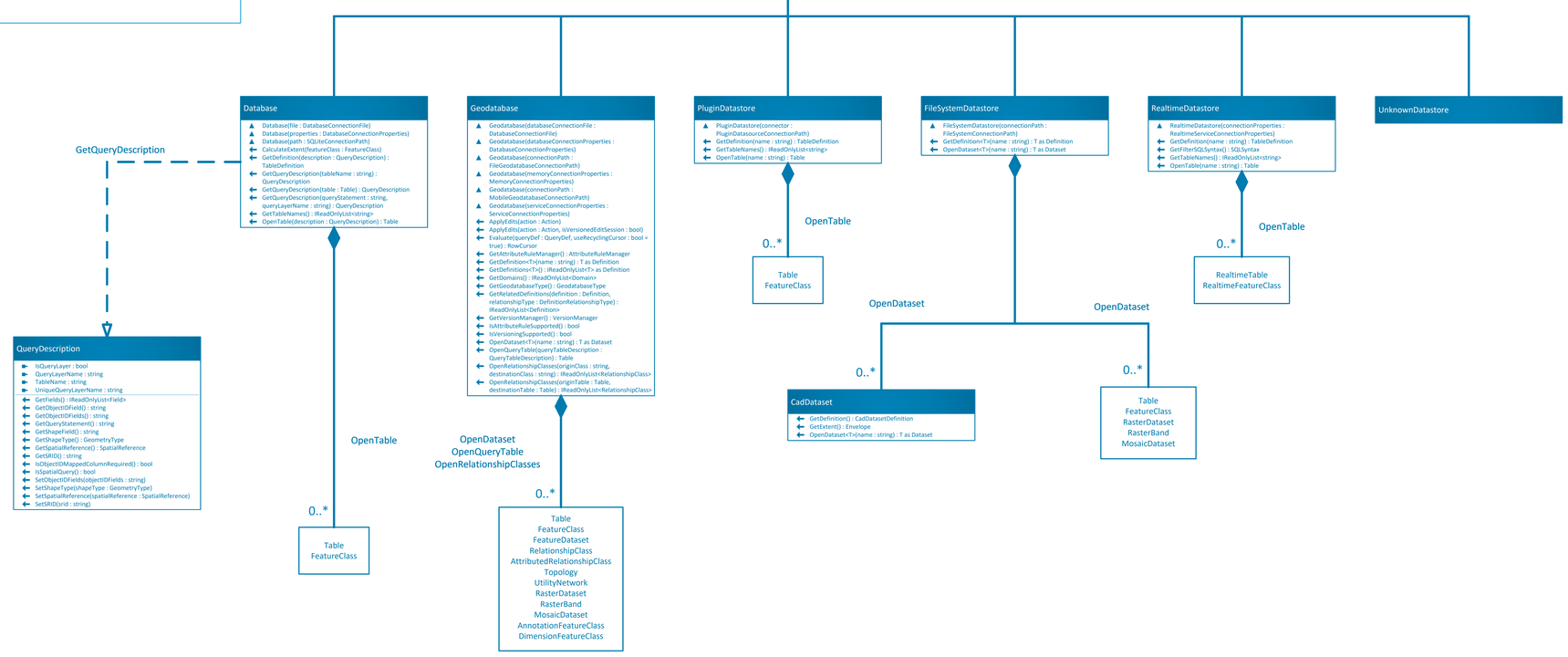
Geodatabase Object Model
Datastore
Esri® ArcGIS® Pro SDK 3.6

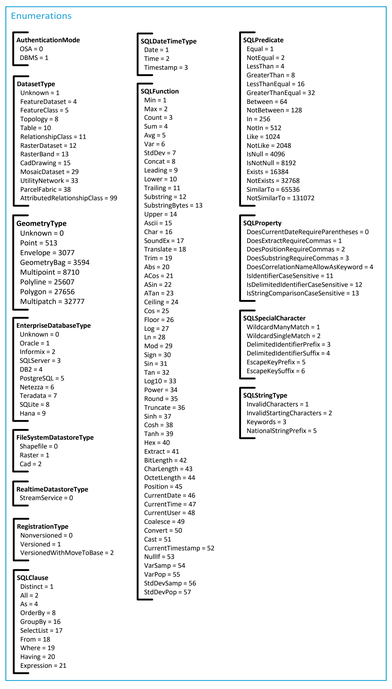
© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.







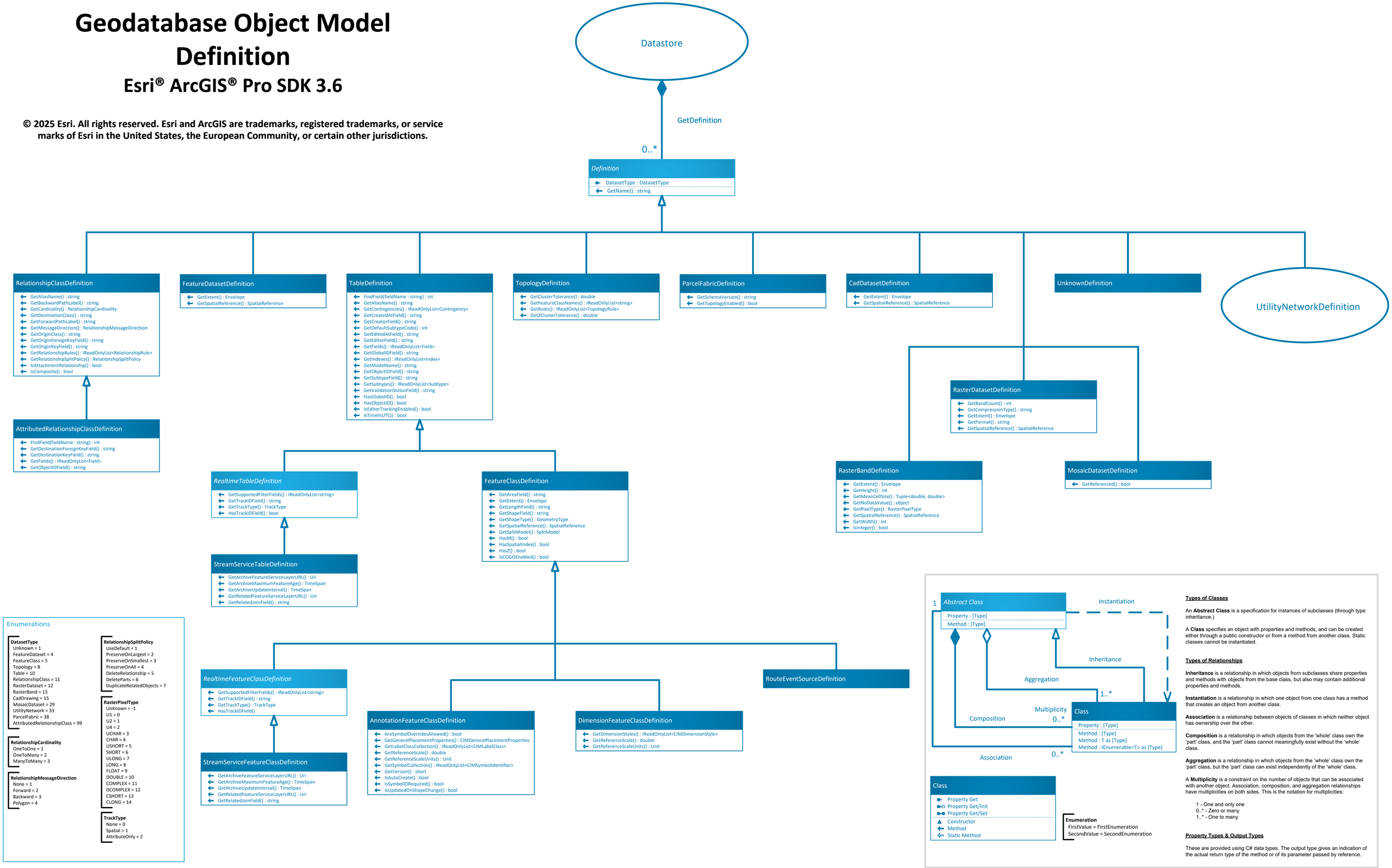




Geodatabase Object Model Definition

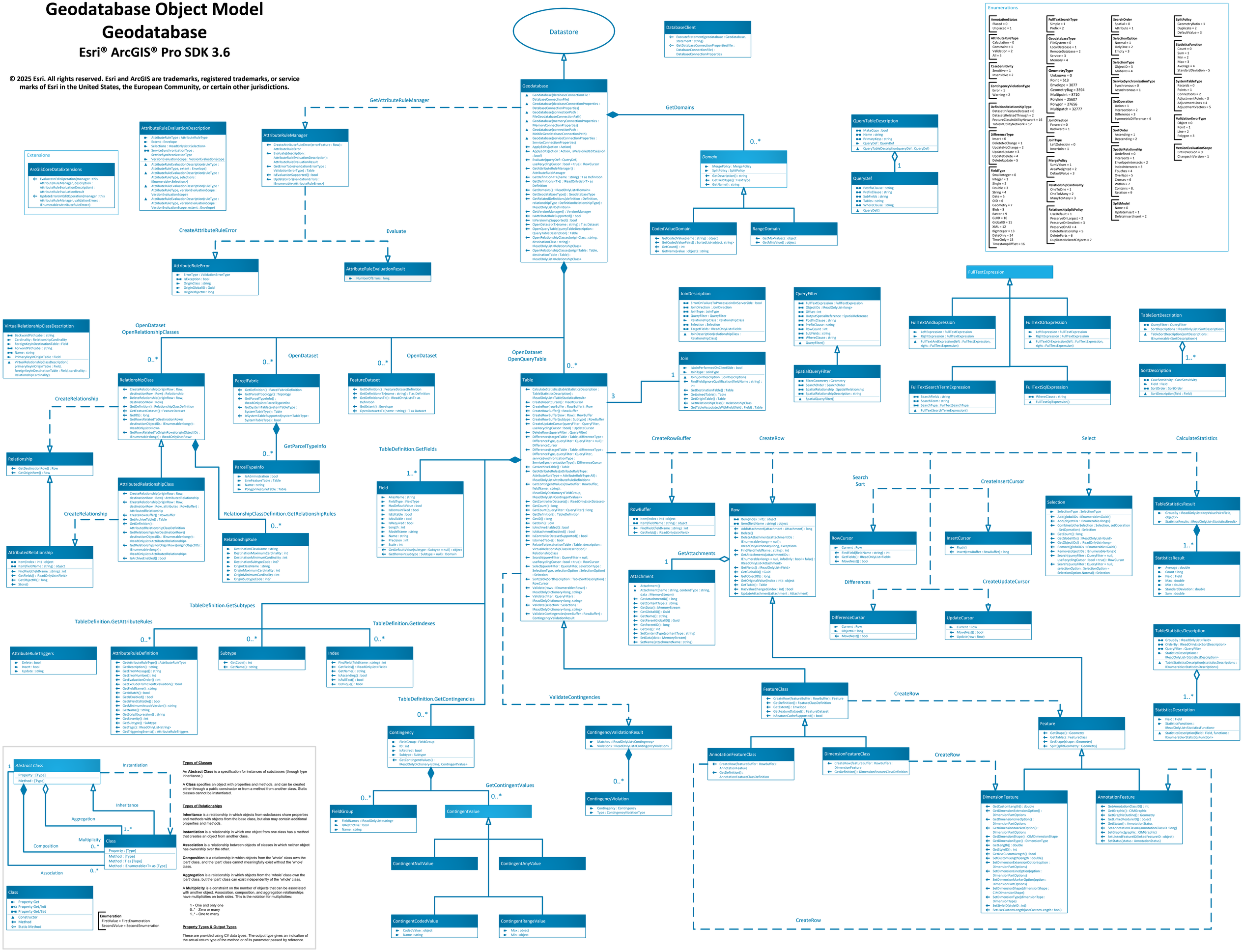
Esri® ArcGIS® Pro SDK 3.6

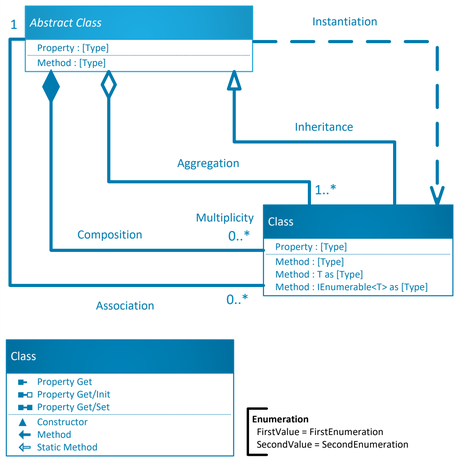
© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.



Geodatabase Object Model
Geodatabase
Esri® ArcGIS® Pro SDK 3.6

© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.





Types of Classes

An **Abstract Class** is a specification for instances of subclasses (through type inheritance.)

A **Class** specifies an object with properties and methods, and can be created either through a public constructor or from a method from another class. Static classes cannot be instantiated.

Types of Relationships

Inheritance is a relationship in which objects from subclasses share properties and methods with objects from the base class, but also may contain additional properties and methods.

Instantiation is a relationship in which one object from one class has a method that creates an object from another class.

Association is a relationship between objects of classes in which neither object has ownership over the other.

Composition is a relationship in which objects from the 'whole' class own the 'part' class, and the 'part' class cannot meaningfully exist without the 'whole' class.

Aggregation is a relationship in which objects from the 'whole' class own the 'part' class, but the 'part' class can exist independently of the 'whole' class.

A **Multiplicity** is a constraint on the number of objects that can be associated with another object. Association, composition, and aggregation relationships have multiplicities on both sides. This is the notation for multiplicities:

1 - One and only one
0..* - Zero or many
1..* - One to many

Property Types & Output Types

These are provided using C# data types. The output type gives an indication of the actual return type of the method or of its parameter passed by reference.

Geodatabase Object Model
Versioning
Esri® ArcGIS® Pro SDK 3.6

© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.

Enumerations

ConflictDetectionType
ByRow = 0
ByColumn = 1

ConflictResolutionMethod
Abort = 0
Continue = 1

ConflictResolutionType
FavorTargetVersion = 0
FavorEditVersion = 1

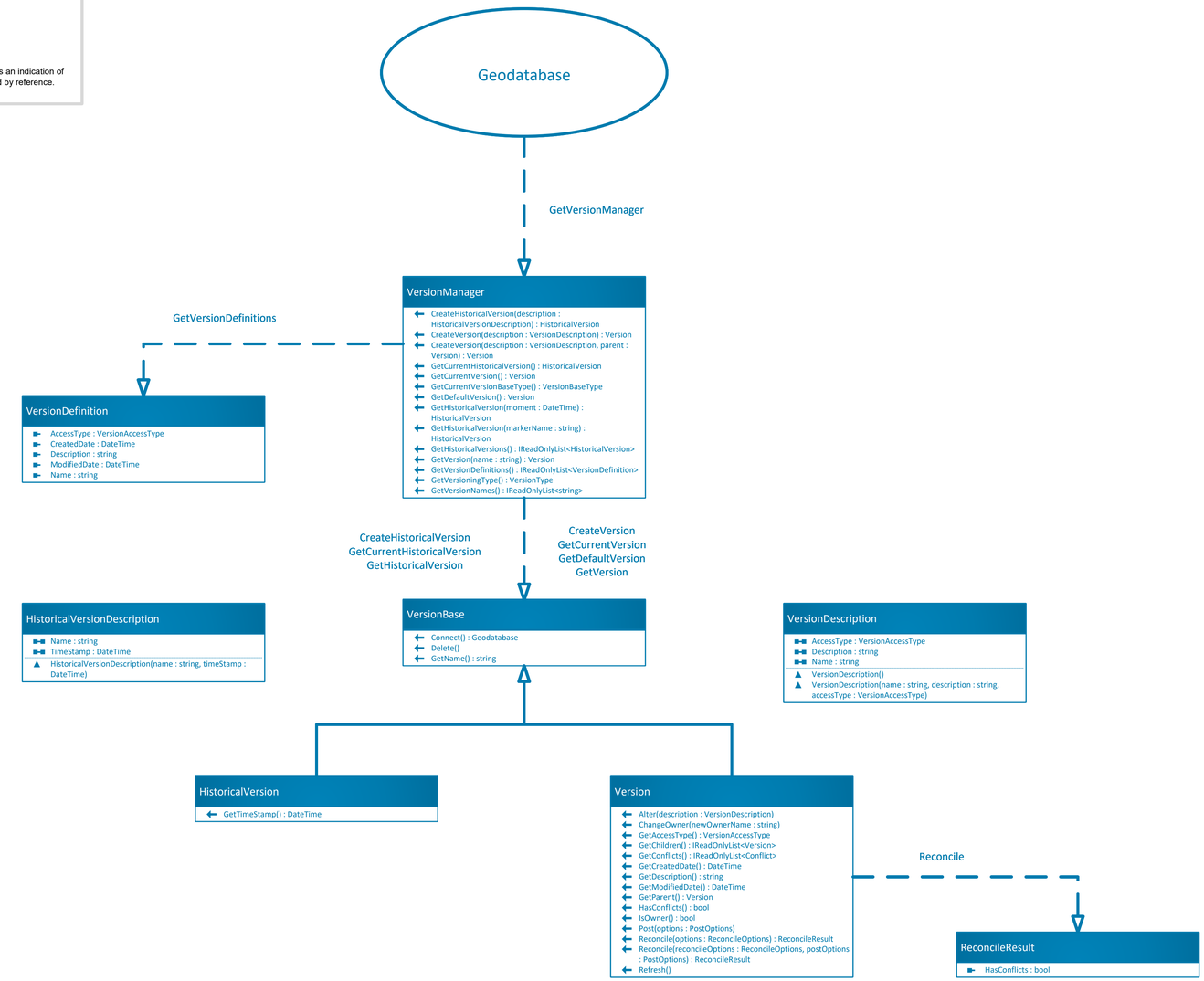
ConflictType
UpdateUpdate = 3
UpdateDelete = 4
DeleteUpdate = 5

ServiceSynchronizationType
Synchronous = 0
Asynchronous = 1

VersionAccessType
Private = 0
Public = 1
Protected = 2

VersionBaseType
Version = 0
HistoricalVersion = 1

VersionType
Traditional = 0
Branch = 1



ReconcileOptions

- ConflictDetectionType : ConflictDetectionType
- ConflictResolutionMethod : ConflictResolutionMethod
- ConflictResolutionType : ConflictResolutionType
- ServiceSynchronizationType : ServiceSynchronizationType
- TargetVersion : Version

- ReconcileOptions()
- ReconcileOptions(targetVersion : Version)

PostOptions

- PartialPostSelections : List<Selection>
- ServiceSynchronizationType : ServiceSynchronizationType
- TargetVersion : Version

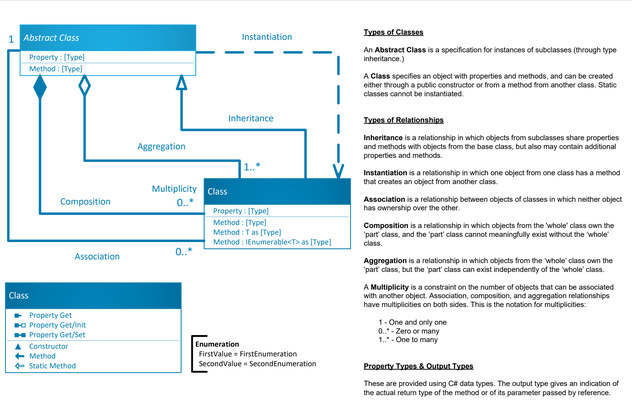
- PostOptions()
- PostOptions(targetVersion : Version)

Conflict

- AncestorVersionValues : IReadOnlyList<FieldValue>
- ChildVersionValues : IReadOnlyList<FieldValue>
- ConflictType : ConflictType
- DatasetName : string
- ObjectID : long
- ParentVersionValues : IReadOnlyList<FieldValue>

Geodatabase Object Model
DDL
Esri® ArcGIS® Pro SDK 3.6

© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.



Types of Classes

An **Abstract Class** is a specification for instances of subclasses (through type inheritance.)

A **Class** specifies an object with properties and methods, and can be created either through a public constructor or from a method from another class. Static classes cannot be instantiated.

Types of Relationships

Inheritance is a relationship in which objects from subclasses share properties and methods with objects from the base class, but also may contain additional properties and methods.

Instantiation is a relationship in which one object from one class has a method that creates an object from another class.

Association is a relationship between objects of classes in which neither object has ownership over the other.

Composition is a relationship in which objects from the 'whole' class own the 'part' class, and the 'part' class cannot meaningfully exist without the 'whole' class.

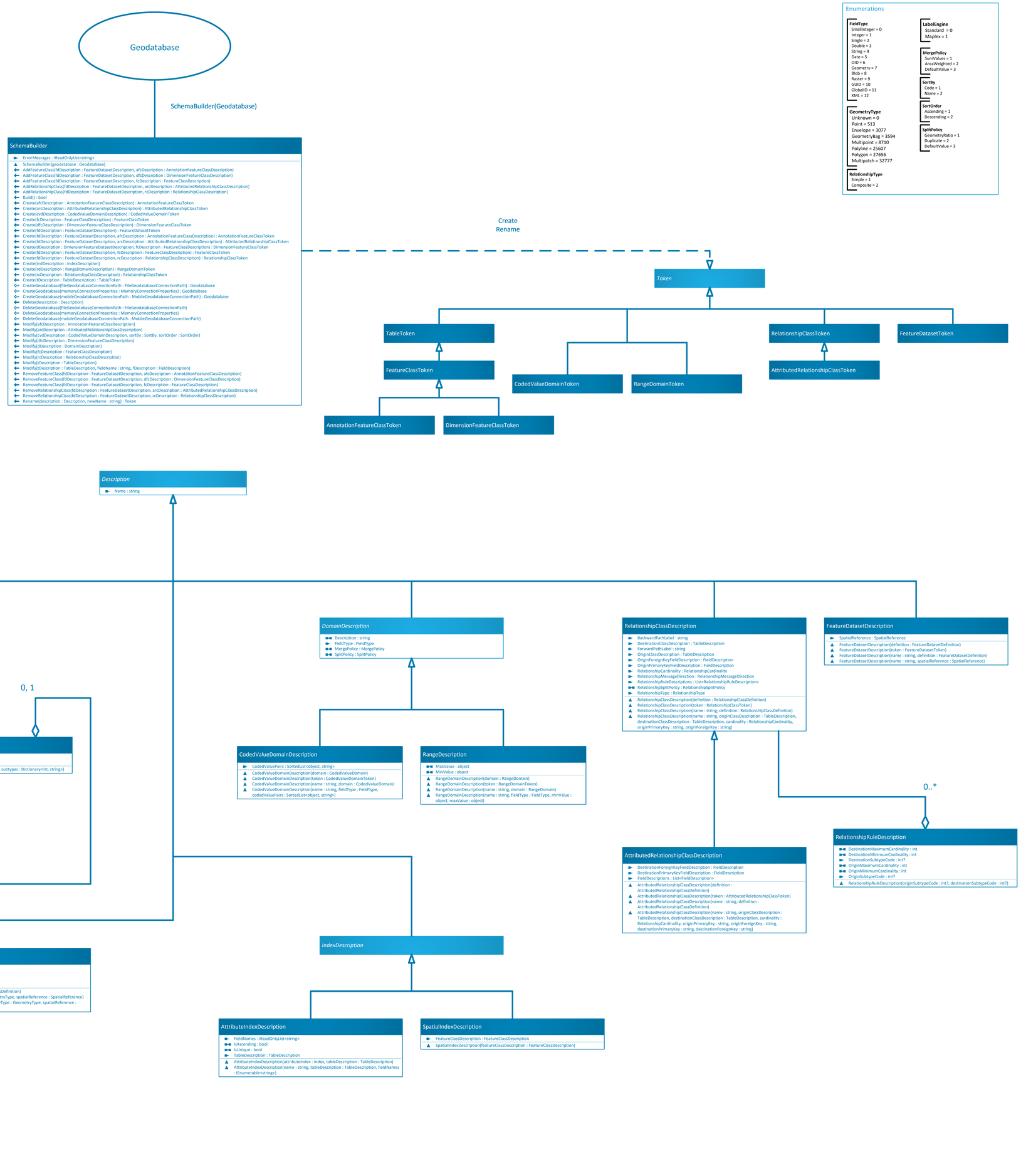
Aggregation is a relationship in which objects from the 'whole' class own the 'part' class, but the 'part' class can exist independently of the 'whole' class.

A **Multiplicity** is a constraint on the number of objects that can be associated with another object. Association, composition, and aggregation relationships have multiplicities on both sides. This is the notation for multiplicities:

1 - One and only one
0..* - Zero or many
1..* - One to many

Property Types & Output Types

These are provided using C# data types. The output type gives an indication of the actual return type of the method or of its parameter passed by reference.



Geodatabase Object Model
Topology
Esri® ArcGIS® Pro SDK 3.6

© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.

Enumerations

ErrorType
- ErrorOnly = 0
- ExceptionOnly = 1
- ErrorAndException = 2

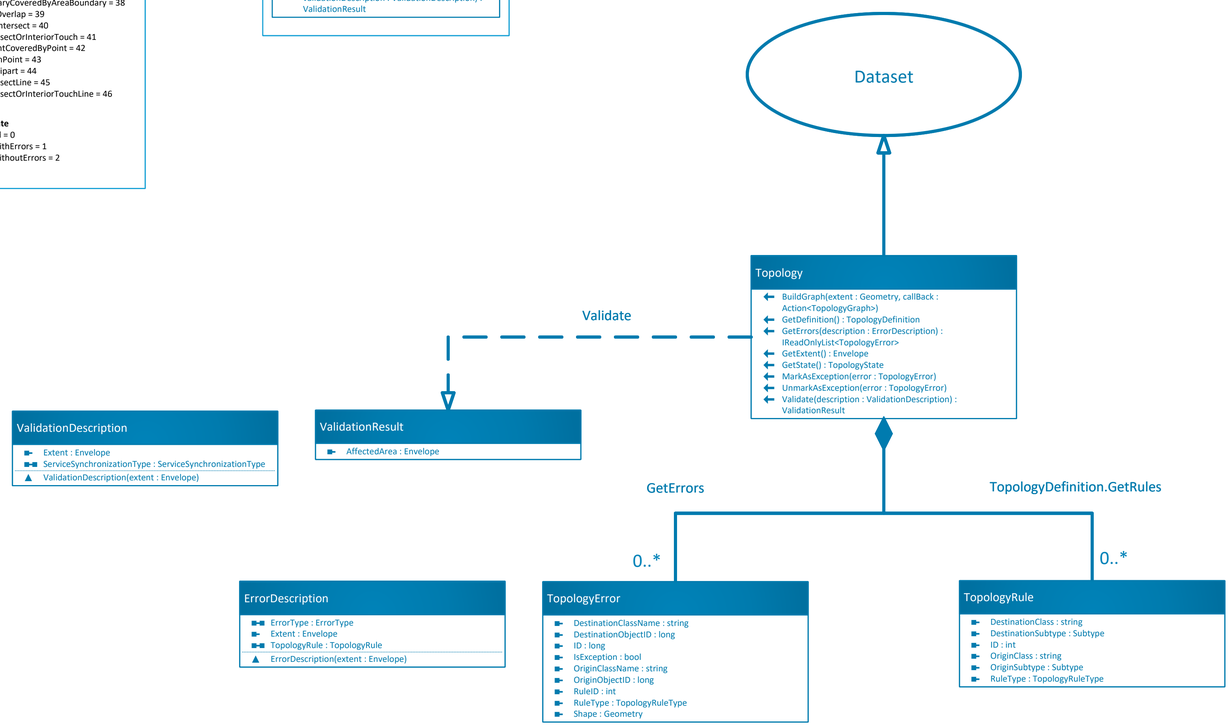
ServiceSynchronizationType
- Synchronous = 0
- Asynchronous = 1

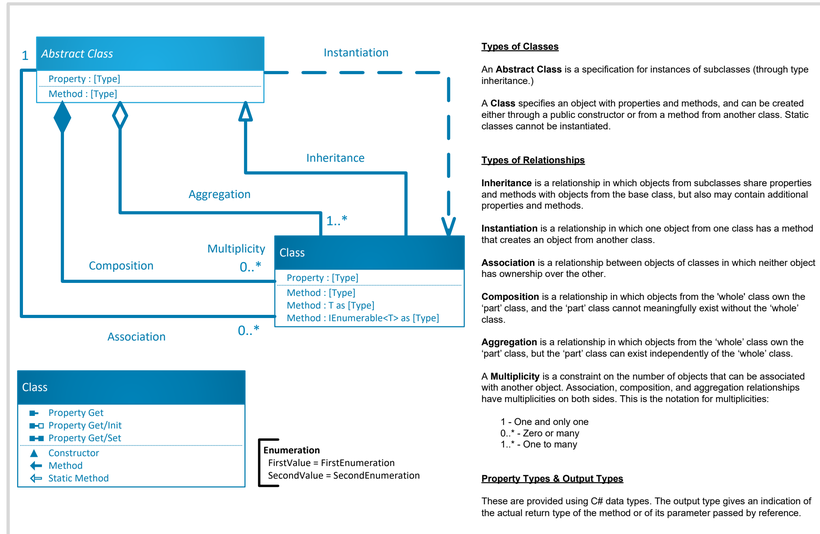
TopologyRuleType
- FeatureLargerThanClusterTolerance = 0
- AreaNoGaps = 1
- AreaNoOverlap = 3
- AreaCoveredByAreaClass = 4
- AreaAreaCoverEachOther = 5
- AreaCoveredByArea = 7
- AreaNoOverlapArea = 8
- LineCoveredByAreaBoundary = 10
- LineInsideArea = 11
- PointCoveredByAreaBoundary = 13
- PointProperlyInsideArea = 15
- AreaContainOnePoint = 16
- LineNoOverlap = 19
- LineNoIntersection = 20
- LineNoDangles = 21
- LineNoPseudos = 22
- LineCoveredByLineClass = 26
- LineNoOverlapLine = 28
- PointCoveredByLine = 29
- PointCoveredByLineEndpoint = 31
- PointDisjoint = 34
- PointCoincidePoint = 35
- AreaBoundaryCoveredByLine = 37
- AreaBoundaryCoveredByAreaBoundary = 38
- LineNoSelfOverlap = 39
- LineNoSelfIntersect = 40
- LineNoIntersectOrInteriorTouch = 41
- LineEndpointCoveredByPoint = 42
- AreaContainPoint = 43
- LineNoMultipart = 44
- LineNoIntersectLine = 45
- LineNoIntersectOrInteriorTouchLine = 46

TopologyState
- Unanalyzed = 0
- AnalyzedWithErrors = 1
- AnalyzedWithoutErrors = 2
- Empty = 3

Extensions

ArcGISCoreDataExtensions
- MarkAsExceptionInEditOperation(topology : this Topology, topologyError : TopologyError)
- UnmarkAsExceptionInEditOperation(topology : this Topology, topologyError : TopologyError)
- ValidateInEditOperation(topology : this Topology, validationDescription : ValidationDescription) : ValidationResult





Types of Classes

An **Abstract Class** is a specification for instances of subclasses (through type inheritance.)

A **Class** specifies an object with properties and methods, and can be created either through a public constructor or from a method from another class. Static classes cannot be instantiated.

Types of Relationships

Inheritance is a relationship in which objects from subclasses share properties and methods with objects from the base class, but also may contain additional properties and methods.

Instantiation is a relationship in which one object from one class has a method that creates an object from another class.

Association is a relationship between objects of classes in which neither object has ownership over the other.

Composition is a relationship in which objects from the 'whole' class own the 'part' class, and the 'part' class cannot meaningfully exist without the 'whole' class.

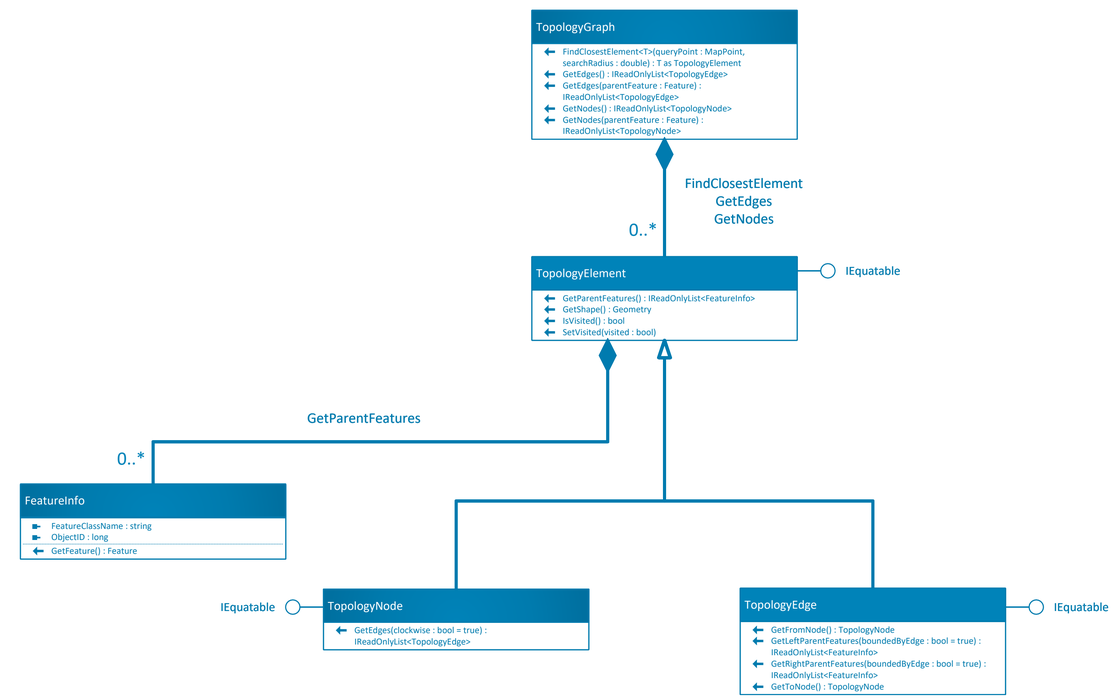
Aggregation is a relationship in which objects from the 'whole' class own the 'part' class, but the 'part' class can exist independently of the 'whole' class.

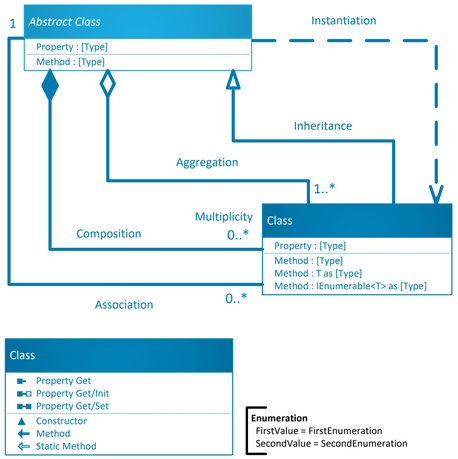
A **Multiplicity** is a constraint on the number of objects that can be associated with another object. Association, composition, and aggregation relationships have multiplicities on both sides. This is the notation for multiplicities:

1 - One and only one
0..* - Zero or many
1..* - One to many

Property Types & Output Types

These are provided using C# data types. The output type gives an indication of the actual return type of the method or of its parameter passed by reference.





Types of Classes

An **Abstract Class** is a specification for instances of subclasses (through type inheritance.)

A **Class** specifies an object with properties and methods, and can be created either through a public constructor or from a method from another class. Static classes cannot be instantiated.

Types of Relationships

Inheritance is a relationship in which objects from subclasses share properties and methods with objects from the base class, but also may contain additional properties and methods.

Instantiation is a relationship in which one object from one class has a method that creates an object from another class.

Association is a relationship between objects of classes in which neither object has ownership over the other.

Composition is a relationship in which objects from the 'whole' class own the 'part' class, and the 'part' class cannot meaningfully exist without the 'whole' class.

Aggregation is a relationship in which objects from the 'whole' class own the 'part' class, but the 'part' class can exist independently of the 'whole' class.

A **Multiplicity** is a constraint on the number of objects that can be associated with another object. Association, composition, and aggregation relationships have multiplicities on both sides. This is the notation for multiplicities:

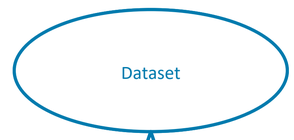
1 - One and only one
0..* - Zero or many
1..* - One to many

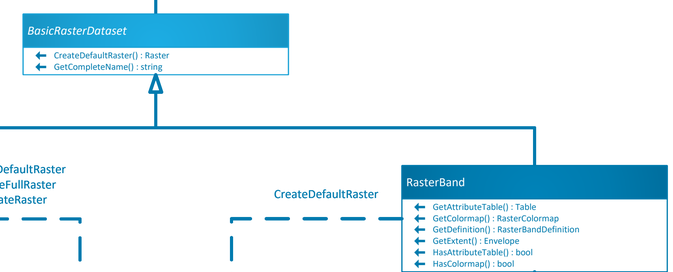
Property Types & Output Types

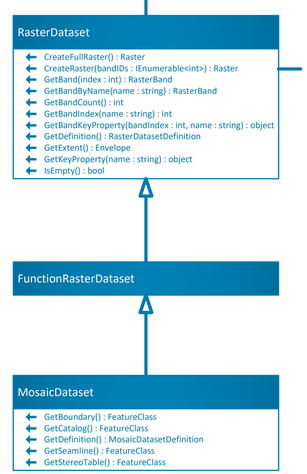
These are provided using C# data types. The output type gives an indication of the actual return type of the method or of its parameter passed by reference.

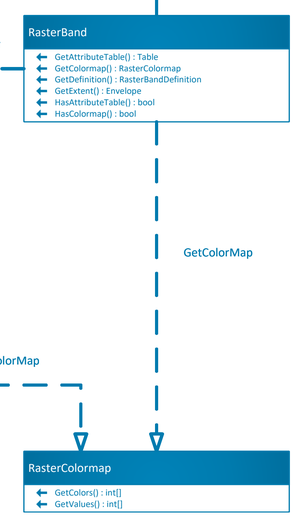
Geodatabase Object Model
Raster
Esri® ArcGIS® Pro SDK 3.6

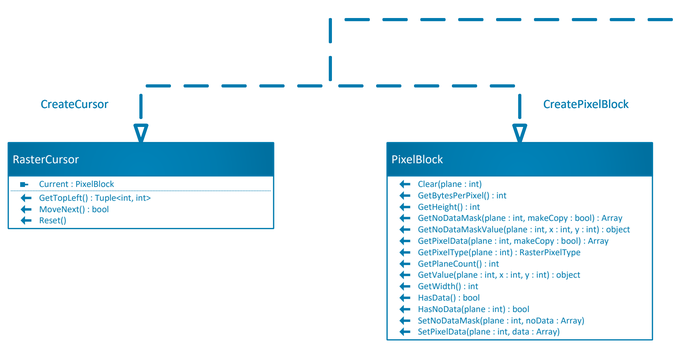
© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.

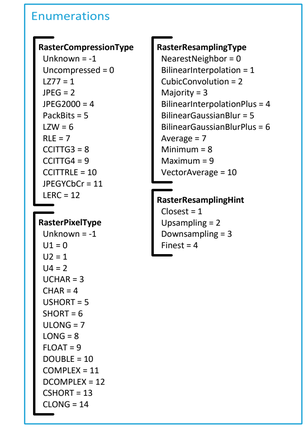


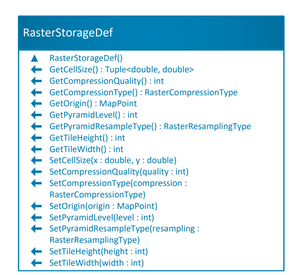








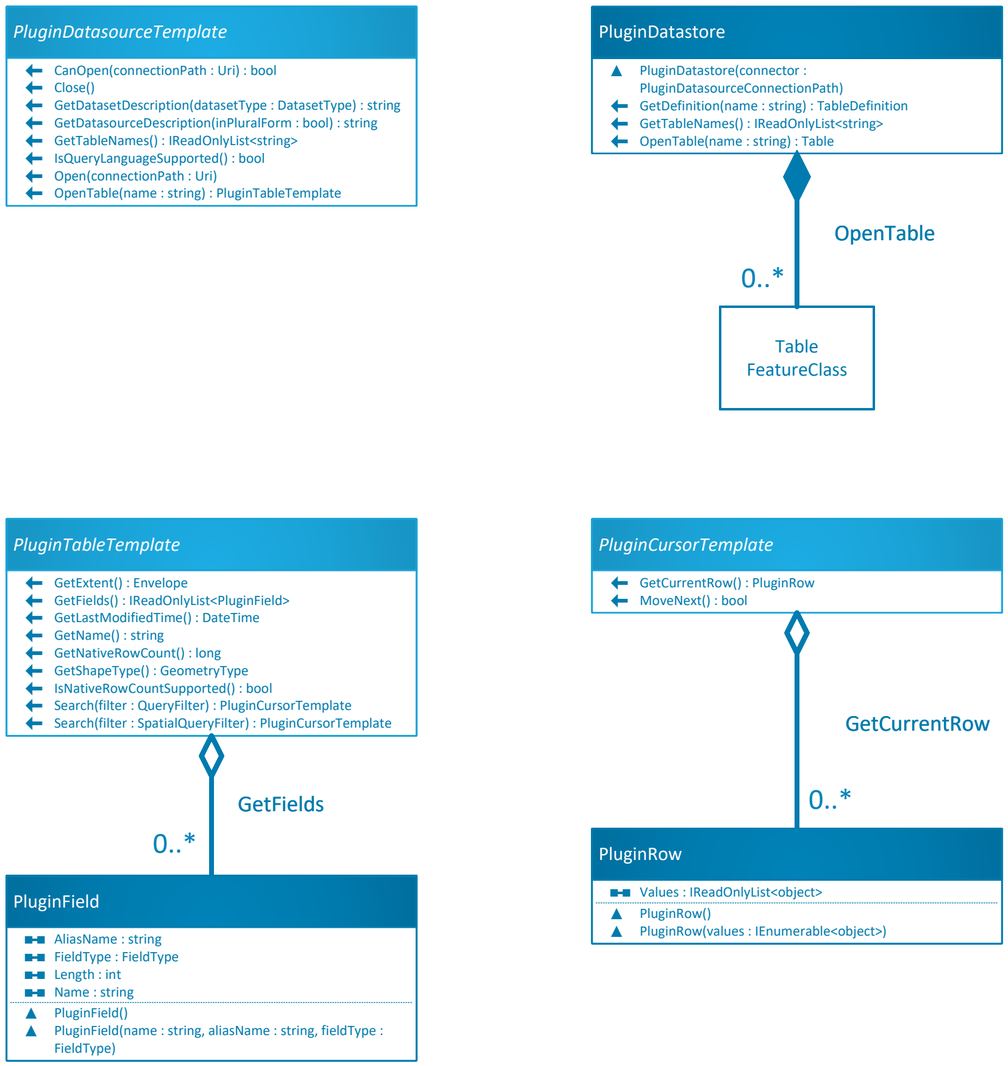


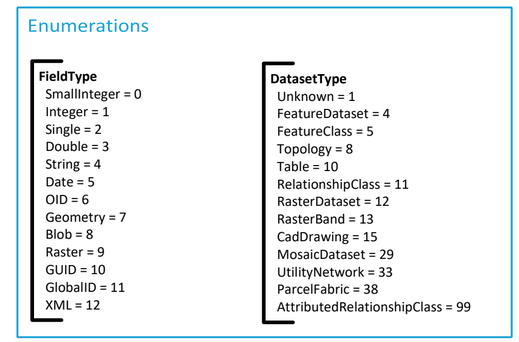


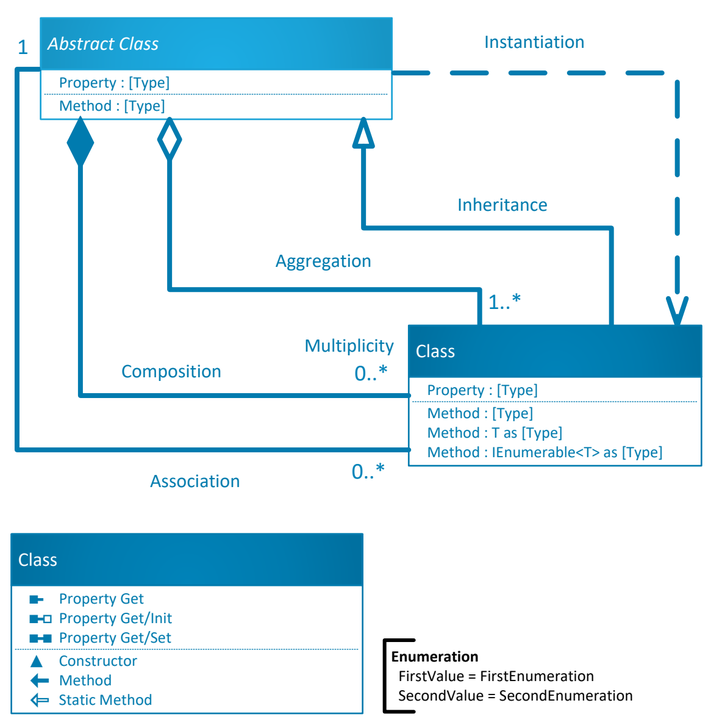
Geodatabase Object Model Plugin

Esri® ArcGIS® Pro SDK 3.6

© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.







Types of Classes

An **Abstract Class** is a specification for instances of subclasses (through type inheritance.)

A **Class** specifies an object with properties and methods, and can be created either through a public constructor or from a method from another class. Static classes cannot be instantiated.

Types of Relationships

Inheritance is a relationship in which objects from subclasses share properties and methods with objects from the base class, but also may contain additional properties and methods.

Instantiation is a relationship in which one object from one class has a method that creates an object from another class.

Association is a relationship between objects of classes in which neither object has ownership over the other.

Composition is a relationship in which objects from the 'whole' class own the 'part' class, and the 'part' class cannot meaningfully exist without the 'whole' class.

Aggregation is a relationship in which objects from the 'whole' class own the 'part' class, but the 'part' class can exist independently of the 'whole' class.

A **Multiplicity** is a constraint on the number of objects that can be associated with another object. Association, composition, and aggregation relationships have multiplicities on both sides. This is the notation for multiplicities:

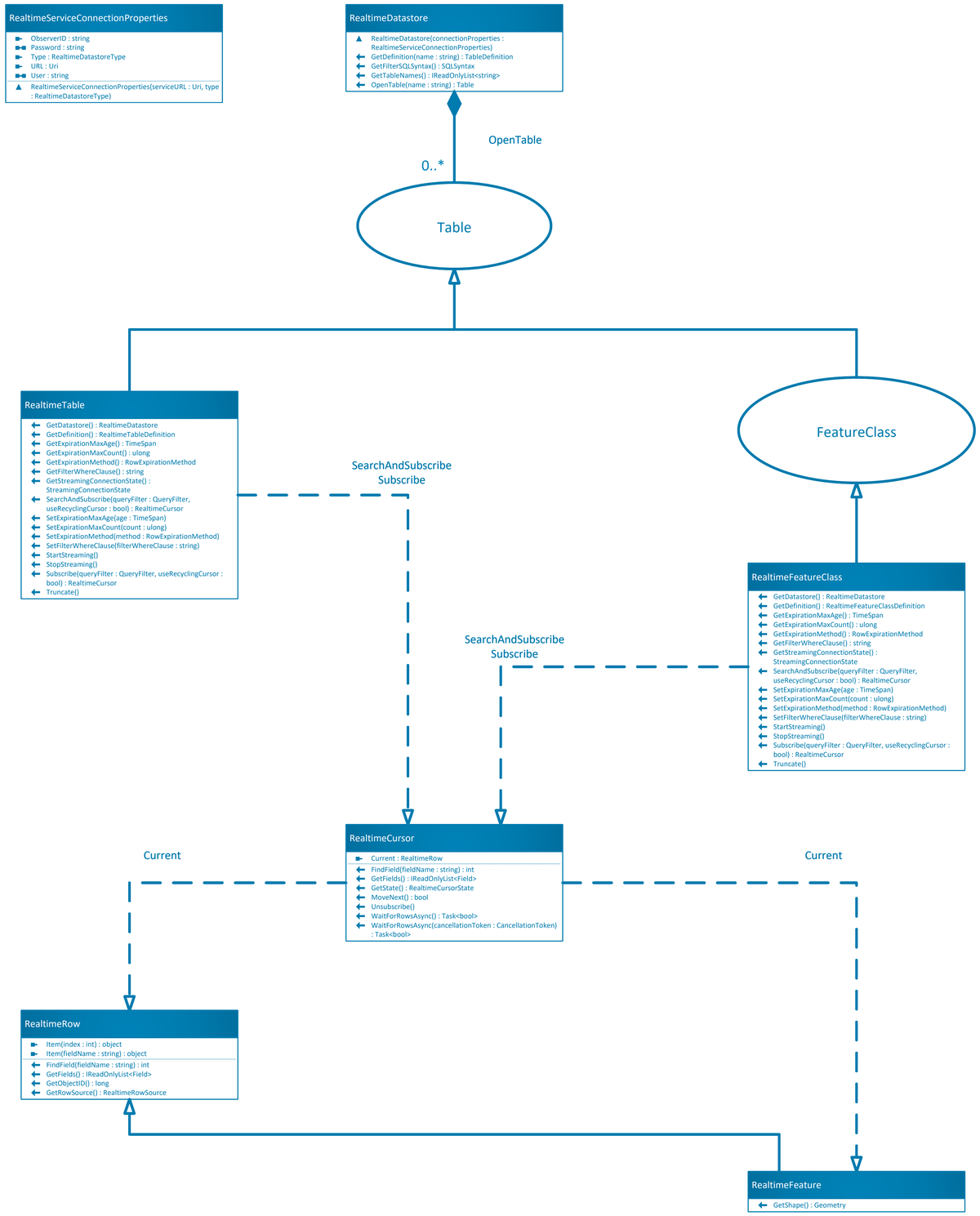
1 - One and only one
0..* - Zero or many
1..* - One to many

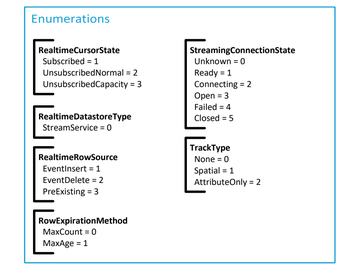
Property Types & Output Types

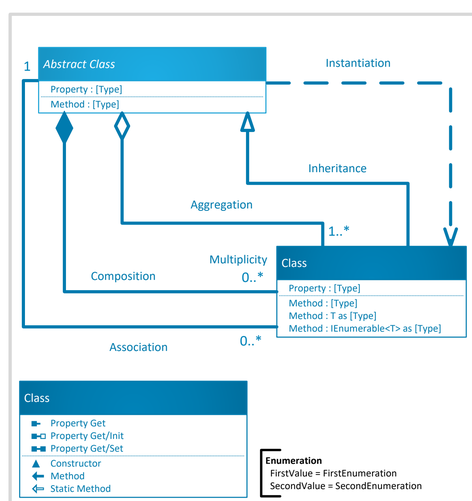
These are provided using C# data types. The output type gives an indication of the actual return type of the method or of its parameter passed by reference.

Geodatabase Object Model
Realtime
Esri® ArcGIS® Pro SDK 3.6

© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.







Types of Classes

An **Abstract Class** is a specification for instances of subclasses (through type inheritance.)

A **Class** specifies an object with properties and methods, and can be created either through a public constructor or from a method from another class. Static classes cannot be instantiated.

Types of Relationships

Inheritance is a relationship in which objects from subclasses share properties and methods with objects from the base class, but also may contain additional properties and methods.

Instantiation is a relationship in which one object from one class has a method that creates an object from another class.

Association is a relationship between objects of classes in which neither object has ownership over the other.

Composition is a relationship in which objects from the 'whole' class own the 'part' class, and the 'part' class cannot meaningfully exist without the 'whole' class.

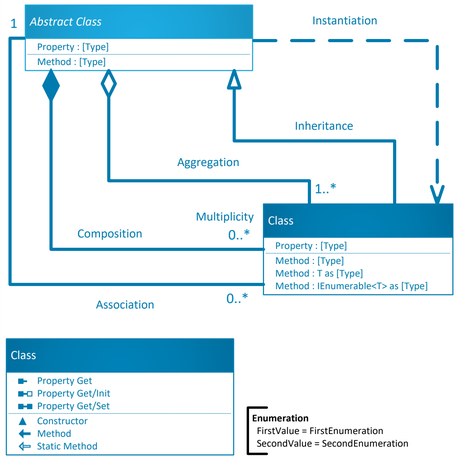
Aggregation is a relationship in which objects from the 'whole' class own the 'part' class, but the 'part' class can exist independently of the 'whole' class.

A **Multiplicity** is a constraint on the number of objects that can be associated with another object. Association, composition, and aggregation relationships have multiplicities on both sides. This is the notation for multiplicities:

1 - One and only one
0..* - Zero or many
1..* - One to many

Property Types & Output Types

These are provided using C# data types. The output type gives an indication of the actual return type of the method or of its parameter passed by reference.



Types of Classes

An **Abstract Class** is a specification for instances of subclasses (through type inheritance.)

A **Class** specifies an object with properties and methods, and can be created either through a public constructor or from a method from another class. Static classes cannot be instantiated.

Types of Relationships

Inheritance is a relationship in which objects from subclasses share properties and methods with objects from the base class, but also may contain additional properties and methods.

Instantiation is a relationship in which one object from one class has a method that creates an object from another class.

Association is a relationship between objects of classes in which neither object has ownership over the other.

Composition is a relationship in which objects from the 'whole' class own the 'part' class, and the 'part' class cannot meaningfully exist without the 'whole' class.

Aggregation is a relationship in which objects from the 'whole' class own the 'part' class, but the 'part' class can exist independently of the 'whole' class.

A **Multiplicity** is a constraint on the number of objects that can be associated with another object. Association, composition, and aggregation relationships have multiplicities on both sides. This is the notation for multiplicities:

1 - One and only one
0..* - Zero or many
1..* - One to many

Property Types & Output Types

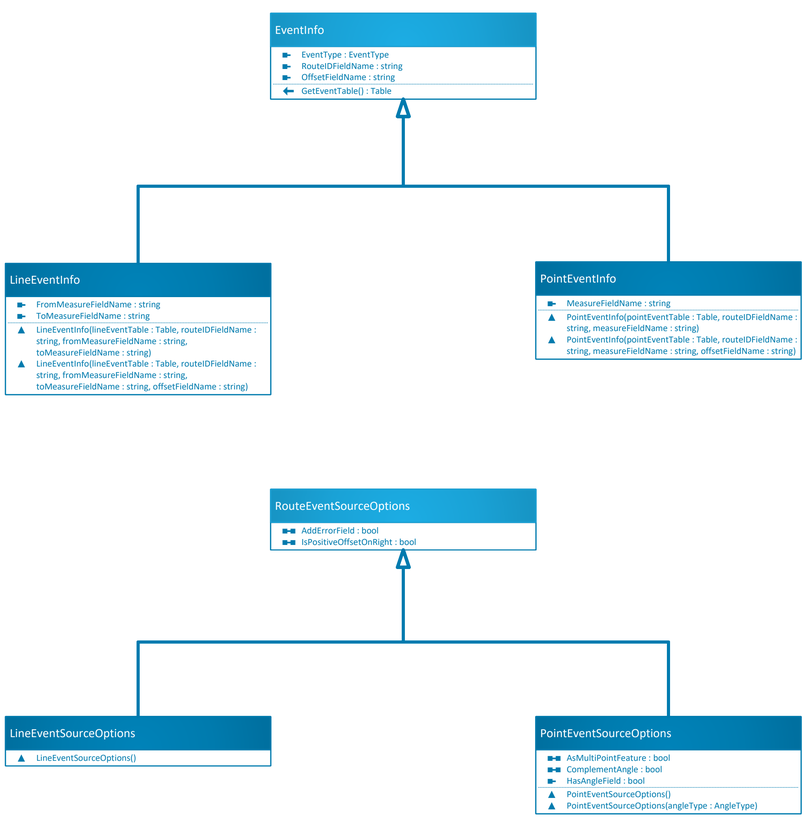
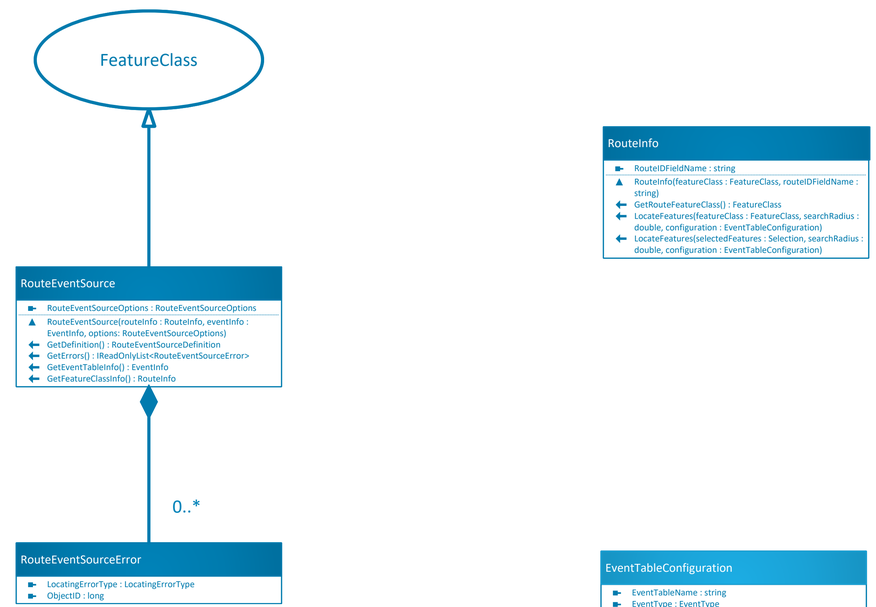
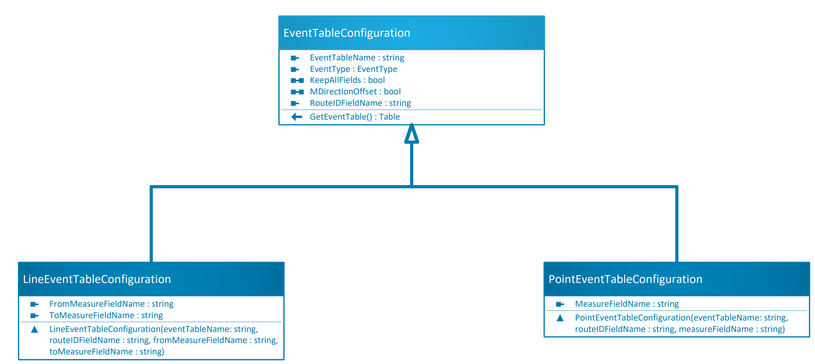
These are provided using C# data types. The output type gives an indication of the actual return type of the method or of its parameter passed by reference.

Geodatabase Object Model
Linear Referencing
Esri® ArcGIS® Pro SDK 3.6

© 2025 Esri. All rights reserved. Esri and ArcGIS are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.

Enumerations

AngleType
Tangent = 0
Normal = 1

EventType
Line = 0
Point = 1

LocatingErrorType
InvalidRouteID = 1
InvalidMeasure = 2
CannotFindRoute = 3
RouteShapeEmpty = 4
CannotFindLocation = 5
CannotFindExtent = 6
FromPartialMatch = 7
ToPartialMatch = 8
RouteMValuesNull = 9
RouteNotMAware = 10
FromToPartialMatch = 11
MultipleLocations = 12
NullExtent = 13