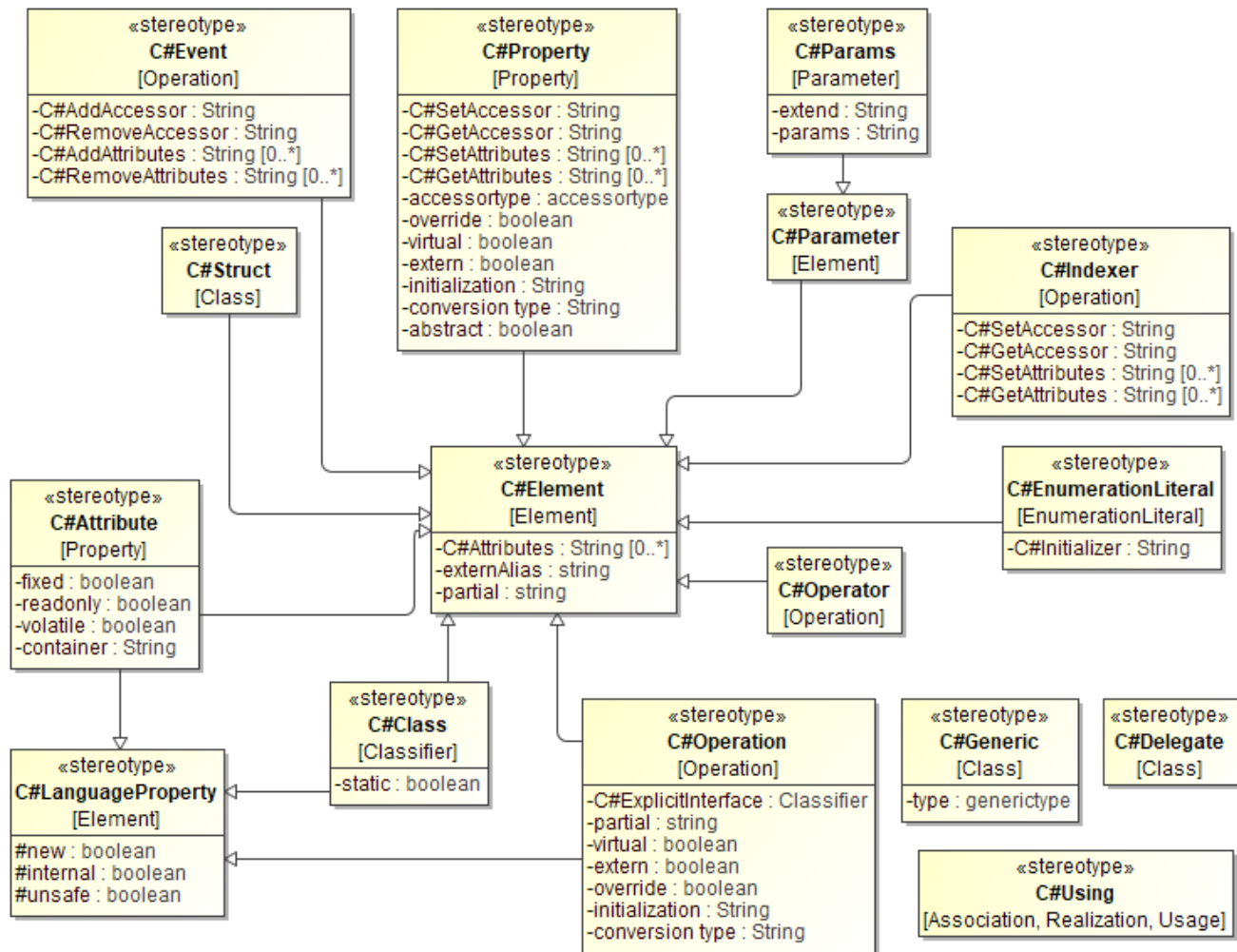


C# stereotypes

The diagram below shows stereotypes of the C# profile.



C# Stereotypes with Tag values

Now our modeling tool's C# Profile language properties is moved to tagged value in a specific stereotypes. Language properties were moved:

- Class is extended with Stereotype «C#Class»
- Attribute is extended with Stereotype «C#Attribute»
- Operation is extended with Stereotype «C#Operation»
- The common information of the old language properties (Class, Attribute and Operation) is extended with Stereotype C#LanguageProperty

Here you will find description about:

- [C#Attribute](#)
- [C#Class](#)
- [C#Delegate](#)
- [C#Element](#)
- [C#EnumerationLiteral](#)
- [C#Event](#)
- [C#Generic](#)
- [C#Indexer](#)
- [C#LanguageProperty](#)
- [C#Operation](#)
- [C#Operator](#)
- [C#Parameter](#)
- [C#Params](#)
- [C#Property](#)
- [C#Strut](#)

- [C#Using](#)

C#Attribute

«C#Attribute» is an invisible stereotype that is used to define C# attribute properties. This is used to store the language properties of the attribute in C#.

Name	Meta class
C#Attribute	Property

Tag	Type	Description
container	String	
fixed	Boolean	Represents the usage of Fixed Size Buffer <pre>unsafe struct A { public fixed int x[5]; }</pre>
readonly	Boolean	Represents the usage of read-only attribute <pre>Class A { public readonly int x; }</pre>
volatile	Boolean	Represents the usage of volatile attribute <pre>Class A { public volatile int x; }</pre>

C#Class

«C#Class» is a stereotype that is used to define property of C# class. This is used to store the language properties of the class in C#. Moreover this class is used for store the information about the static class too.

Name	Meta class
C#Class	Class

Tag	Type	Description
Static	boolean	Represents the usage of static class <pre>static Class A { public int x; }</pre>

C#Delegate

«C#Delegate» is a stereotype that indicates that model class represents C# delegate.

Name	Meta class
C#Delegate	Class

C#Element

«C#Element» is a stereotype that is used to define properties of all element in model.

Name	Meta class	Constraints
C#Element	Element	This stereotype can be applied to all elements such as class, attribute, operation and parameter.

Tag	Type	Description
externAlias	String	Represents the usage of extern alias extern alias X; <pre>class Test { X::N.A a; }</pre>
partial	String	
C#Attributes	String	Represents C# attributes for element.

C#EnumerationLiteral

«C#EnumerationLiteral» is a stereotype that is used to define enumeration.

Name	Meta class
C#EnumerationLiteral	EnumerationLiteral

Tag	Type	Description
C#Initializer	String	Represents enumeration member's constant value.

C#Event

«C#Event» is a stereotype that is used to indicate that operation represents C# event.

Name	Meta class
C#Event	Operation

Tag	Type	Description
C#AddAccessor	String	Adds add accessor for event
C#AddAttributes	String	Defines C# attributes for add accessor.
C#RemoveAccessor	String	Adds remove accessor for event
C#RemoveAttributes	String	Defines C# attributes for remove accessor.

C#Generic

«C#Generic» is a stereotype that is used to define generic properties.

Name	Meta class
C#Generic	Class

Tag	Type	Description
type	generictype	Defines type for generic constraint parameter, class, struct, and new()

C#Indexer

«C#Indexer» is a stereotype that is used to indicate that operation represents C# indexer.

Name	Meta class
C#Indexer	Operation

Tag	Type	Description
C#GetAccessor	String	Adds get accessor for indexer.
C#GetAttributes	String	Defines C# attributes for get accessor.
C#SetAccessor	String	Adds set accessor for indexer.
C#SetAttributes	String	Defines C# attributes for set accessor.

C#LanguageProperty

«C#LanguageProperty» is the parent of «C#Class», «C#Attribute», and «C#Operation», So the «C#Class», «C#Attribute» and «C#Operation» also have its definition.

Name	Meta class
C#LanguageProperty	Element

Tag	Type	Description
internal	boolean	
new	boolean	
unsafe	boolean	Represents the usage of unsafe element unsafe struct A { public fixed int x[5]; }

C#Operation

«C#Operation» is a stereotype which is used to define properties of the operation. This is used to store the language properties of the operation in C#.

Name	Meta class
------	------------

C#Operation	Operation
-------------	-----------

Tag	Type	Description
conversion type	String	
extern	boolean	Represent the usage of extern operation Class A { public extern int x(); }
initialization	String	
override	boolean	
partial	String	
virtual	boolean	
C#ExplicitInterface	Classifier	Defines C# explicit interface for explicit interface member implementation.

C#Operator

«C#Operator» is a stereotype that is used to indicate that the operation represents C# operator.

Name	Meta class
C#Operator	Operation

C#Parameter

«C#Parameter» is a stereotype that is used to indicate that the element represents C#Parameter.

Name	Meta class
C#Parameter	Element

C#Params

«C#Params» is a stereotype that is used to indicate parameter arrays.

Name	Meta class
C#Params	Parameter

Tag	Type
extend	String
params	String

C#Property

«C#Property» is a stereotype that is used to indicate that the operation represents C# property.

Name	Meta class
C#Property	Operation

Tag	Type	Description
C#GetAccessor	String	Adds get accessor for indexer.
C#GetAttributes	String	Defines C# attributes for get accessor.
C#SetAccessor	String	Adds set accessor for indexer.
C#SetAttributes	String	Defines C# attributes for set accessor.

C#Struct

«C#Struct» is a stereotype that is used to indicate that the model class represents C# structure.

Name	Meta class
C#Struct	Class

C#Using

«C#Using» is a stereotype that is used to indicate that the usage dependency is C# Using directive.

Name	Meta class	Constraints
C#Using	Association, Realization, Usage	The clients of dependency are Component, and Namespace.

Related pages:



Unknown macro: 'list-children'