

9 ~~8.2~~ SACMElement (abstract)

SACMElement is the base class for SACM.

Superclass

MOF:Element

Attributes

gid:String[0..1] – a unique identifier that is unique within the scope of the model instance

isCitation[0..1]=false – a flag to indicate whether the SACMElement cites another SACMElement.

isAbstract[0..1]=false – a flag to indicate whether the SACMElement is considered to be abstract. For example, this can be used to indicate whether an element is part of a pattern or template.

Associations:

citedElement:SACMElement[0..1] – a reference to another SACMElement that the SACMElement cites

abstractForm:SACMElement[0..1] – an optional reference to another abstract SACMElement to which this concrete SACMElement conforms.

Semantics

All the elements of a structured assurance case effort created with SACM correspond to a SACMElement.

Constraints:

If citedElement is populated, isCitation must be true. OCL: self.citedElement <> null implies self.isCitation = true

When +abstractForm is used to refer to another SACMElement, +isAbstract of the SACMElement is false, and the +isAbstract of the referred SACMElement should be true. The referred SACMElement should be of the same type of the SACMElement. If ImplementationConstraints are expressed on the referred SACMElement, the SACMElement should satisfy these ImplementationConstraints.

~~9 8.3 LangString~~

~~LangString is the format SACM uses for description. It serves the same purpose as String but with the additional specification of the language used for the content.~~

~~Superclass~~

~~MOF:Element~~

~~Attributes~~

~~lang:String[0..1] – a field to indicate the language used in the string.~~

~~content:String[0..1] – the content of the string~~

~~Semantics~~

~~LangString serves the same purpose as String, SACM uses LangString for description, which containing the information of the language it uses in the content.~~

Move to Terminology Class

9 8.4 ExpressionLangString

ExpressionLangString is used to denote a structured expression, it ~~contains a description (LangString) and it also (optionally) points to an ExpressionElement in the Terminology Package.~~

inherits description (LangString) and it also (optionally) points to an ExpressionElement. This will allow terms in one language (perhaps only in US English) to have the term translated into multiple other languages for use as well.

Superclass

MultiLangString

Attributes

expression:ExpressionElement[0..1]

~~expression:Terminology::ExpressionElement[1] (composition) – a reference to an ExpressionElement in the TerminologyPackage~~

Semantics

~~ExpressionLangString provides a means for description, it can also be used to link to an ExpressionElement in the Terminology package.~~

~~**Constraints**~~

~~If expression is not empty, then content should be empty.~~

9 8.5 MultiLangString

MultiLangString, as its name suggests, provides a means to describe things using different languages.

Superclass

~~Element~~ **Foundation::Element**

Associations

~~value:LangString[1..*] (composition) – contains the descriptions which bear the same meaning but in different languages~~

content:String[0..*] {ordered,nonunique} – the content of the string.
lang:String[0..*] {ordered} – a field to indicate the language used in the string.

Semantics

MultiLangString provides a means to describing things using different languages. It contains a ~~list of LangString, which the user can specify their languages and the descriptions in the languages.~~

Constraints

set of contents each of which is in a different language represented by the lang attribute.

~~For each of the LangString in the value feature, their lang must be unique.~~

LangIfContentMoreThan1

If content has more than one value, then lang must have string representing the languages of the values.

inv: content->size() > 1 implies lang->size() = content->size()

9 8.6 3 ModelElement (abstract)

ModelElement is the base element for the majority of modeling elements.

Superclass

SACMElement

Associations

name:LangString[1] (composition) – the name of the ModelElement.

implementationConstraint: ImplementationConstraint [0..*] (composition) – a collection of implementation constraints.

description: Description[0..1] (composition) – the description of the ModelElement.

note:Note[0..*] (composition) – a collection of notes for the ModelElement.

taggedValue: TaggedValue [0..*] (composition) – a collection of TaggedValues, TaggedValues can be used to describe additional features of a ModelElement

Semantics

All the individual and identifiable elements of a SACM model correspond to a ModelElement.

Constraints

ImplementationConstraints should only be specified if +isAbstract is true

OCL:

self.implementationConstraint->size() > 0 implies self.isAbstract = true

9 ~~8.7~~ 4 UtilityElement (abstract)

UtilityElement is the base element for a number of auxiliary elements which can be added to ModelElements.

Superclass

SACMElement

Associations

content:MultiLangString[0..1] (composition) – a MultiLangString to describe the content of the UtilityElement in (possibly) multiple languages

Semantics

UtilityElement supports the specification of additional information for a ModelElement.

9 ~~8.8~~ 5 ImplementationConstraint

ImplementationConstraint specifies details of any implementation constraints that must be satisfied whenever a referencing ModelElement is to be converted from *isAbstract = true* to *isAbstract = false*. For example in the context of a SACM pattern fragment, an element will need to satisfy the implementation rules of the pattern.

Superclass

UtilityElement

Semantics

ImplementationConstraints indicate the conditions to fulfill in order to allow an abstract ModelElement (*isAbstract = true*) to become non-abstract (*isAbstract = false*).

9 ~~8.9~~ 6 Description

Description is used to specify a description that may be associated with a ModelElement. In many cases Description is used to provide the ‘content’ of a SACM element. For example, it would be used to provide the text of a Claim.

Superclass

UtilityElement

Semantics

A Description provides details about ModelElements in relation to aspects such as their content or purpose. Therefore, Descriptions can be used to both characterize ModelElements and facilitate their understanding.

9 ~~8.10~~ 7 ArtifactElement (abstract)

ArtifactElement acts as the base class for elements in other SACM packages. Essentially, all elements which extend ArtifactElement is considered to be an artifact, and therefore can be referenced using Argument:ArtifactReference.

Superclass

ModelElement

Semantics

ArtifactElement corresponds to the base class for specifying all the identifiable units of data modelled and managed in a structured assurance case effort.

9 ~~8.11~~ 8 Note

This class specifies a generic note that may be associated with a ModelElement. For example a note may include a number of explanatory comments.

Superclass

UtilityElement

Semantics

Notes are used to specify additional (typically optional) generic, unstructured, untyped information about a ModelElement. An example of this kind of information could be a comment about a ModelElement.

9 ~~8.12~~ 9 TaggedValue

This class represents a simple key/value pair that can be attached to any element in SACM. This is a simple extension mechanism to allow users to add attributes to each element beyond those already specified in SACM.

Superclass

UtilityElement

Associations

key:MultiLangString[1] (composition) – the key of the TaggedValue.

Semantics

TaggedValues can be used to specify attributes, and their corresponding values, for ModelElements.

10 Structured Assurance Case Terminology Classes

10.1 General

This chapter presents the normative specification for the SACM Terminology Metamodel. It begins with an overview of the metamodel structure followed by a description of each element

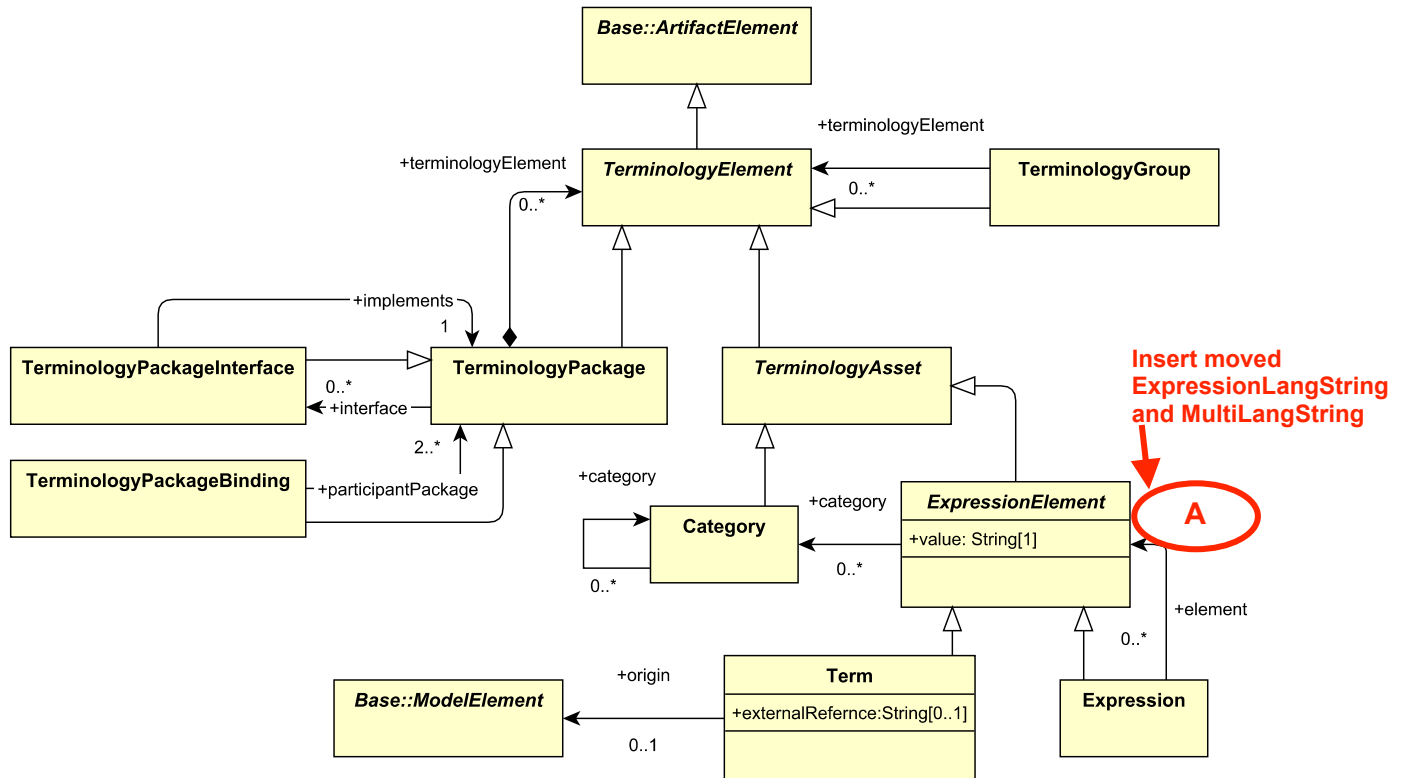


Figure 10.1 - Terminology Class Diagram

This portion of the SACM metamodel describes and defines the concepts of term, expression and an external interface to terminology information from others. This area of the Structured Assurance Case Metamodel also provides the starting foundation for formalism in the assembly of terms into expressions without mandating the formalism for those that do not need it.

10.2 TerminologyElement (abstract)

TerminologyElement is an abstract class that serves as a parent class for all SACM terminology assets (TerminologyAsset) and the grouping of TerminologyElements (TerminologyGroup). TerminologyElement extends Base::ArtifactElement, this implies that all elements in the Terminology package are artifacts.

Superclass

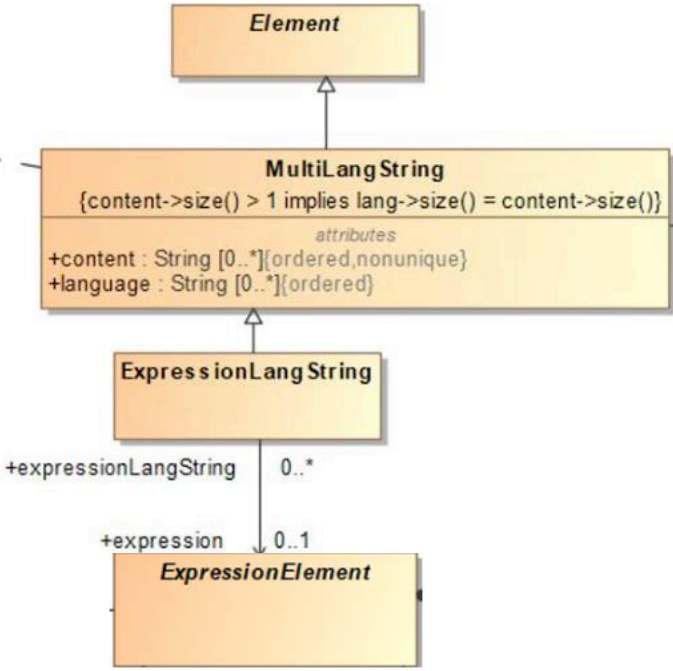
Base::ArtifactElement

Semantics

TerminologyElement is the base class for specifying the terminology aspects of an assurance case (AssuranceCasePackage).

A

«comment»
It is understood that some tools may not be able to support MultiLangString and will only support String



«comment»
Ordering is meant to keep the String "value" array and the String "language" aligned

10.11 Term

Term is used to model both abstract and concrete terms in SACM. Abstract Terms can be considered placeholders for concrete terms and are denoted by the inherited `isAbstract:Boolean` attribute being set true. A concrete term is denoted by `isAbstract:Boolean` being false.

Superclass

`ExpressionElement`

Attributes

`externalReference: String[0..1]` – an attribute recording an external reference (e.g., URI) to the object referred to by the Term

Associations

`origin:Base::ModelElement[0..1]` – a reference which points to the origin of the Term.

Semantics

Term class is used to model both abstract and concrete terms in SACM. Abstract Terms can be considered placeholders for concrete terms and are denoted by the inherited `isAbstract:Boolean` attribute being set true. A concrete term is denoted by `isAbstract:Boolean` being false.

The `externalReference` attribute enables the referencing of the object signified by the term (i.e., the signifier). It also provides a mechanism whereby terms can reference concepts and terms defined in other ontology and terminology models.

Moved from Base Class

10.12 ExpressionLangString

ExpressionLangString is used to denote a structured expression, it inherits description (LangString) and it also (optionally) points to an ExpressionElement.

Superclass

MultiLangString

Attributes

`expression:ExpressionElement[0..1]` – a reference to an ExpressionElement in the TerminologyPackage

Semantics

ExpressionLangString provides a means for description, it can also be used to link to an ExpressionElement.

10.13 MultiLangString

MultiLangString, as its name suggests, provides a means to describe things using different languages.

Superclass

Foundation::Element

Associations

`content:String[0..*] {ordered,nonunique}` – the content of the string.

`lang:String[0..*] {ordered}` – a field to indicate the language used in the string.

Semantics

MultiLangString provides a means to describing things using different languages. It contains a set of contents each of which is in a different language represented by the lang attribute.

Constraints

LangIfContentMoreThan1

If content has more than one value, then lang must have string representing the languages of the values.

inv: content->size() > 1 implies lang->size() = content->size()