

interface: AssuranceCasePackageInterface [0..\*] – a number of optional assurance case package interfaces that the current package may implement

artifactPackage: ArtifactPackage [0..\*] (composition) – a number of optional artifact sub-packages

terminologyPackage: TerminologyPackage [0..\*] (composition) – a number of optional terminology sub-packages

argumentPackage: Argument::ArgumentPackage[0..\*] (composition) – a number of optional argument packages.

### Semantics

AssuranceCasePackage is the root class for creating structured assurance cases.

## 9.3 AssuranceCasePackageInterface

AssuranceCase

AssuranceCasePackageInterface is a kind of AssuranceCasePackage that defines an interface that may be exchanged between users. An AssuranceCasePackage may declare one or more ~~ArtifactPackageInterfaces~~.

### Superclass

AssuranceCasePackage

### Associations

implements: AssuranceCasePackage[1] – the AssuranceCasePackage that the AssuranceCasePackageInterface declares.

### Semantics

AssuranceCasePackageInterface enables the declaration of the elements of an AssuranceCasePackage that might be referred to (cited) in another AssuranceCasePackage. These declarations are provided by containing AssuranceCasePackageInterface(s)/ArgumentPackageInterface(s)/ArtifactPackageInterface(s)/TerminologyPackageInterface(s) to the packages contained by the AssuranceCasePackage (for which the interface provided).

### Constraints

AssuranceCasePackageInterface are only allowed to contain the following: AssuranceCasePackageInterface, ArgumentPackageInterfaces, ArtifactPackageInterfaces, and TerminologyPackages.

### OCL:

```
self.assuranceCasePackage->forall(acp|acp.ocllsTypeOf(AssuranceCasePackageInterface)) and  
self.argumentPackage->forall(ap|ap.ocllsTypeOf(Argumentation::ArgumentPackageInterface)) and  
self.artifactPackage->forall(ap|ap.ocllsTypeOf(Artifact::ArtifactPackageInterface)) and  
self.terminologyPackage->forall(tp|tp.ocllsTypeOf(Terminology::TerminologyPackageInterface))
```

## 9.4 AssuranceCasePackageBinding

Sub-packages within the AssuranceCasePackage can be bound together by means of AssuranceCasePackageBindings. AssuranceCasePackageBindings bind the participant packages by means of ArgumentPackageBindings/TerminologyPackageBindings/ArtifactPackageBindings elements that bind the contained packages of the participant packages.

### Superclass

AssuranceCasePackage

### Associations

participantPackage: AssuranceCasePackage[2..\*] – references to AssuranceCasePackages which the AssuranceCasePackageBinding binds together.