

15.3 SACM View

SACMView allows for the display of Assurance Cases in some readable forms for people. These forms usually take the form of Graphical or Tabular form. SACMView defines the structure of those readable forms so that they can be serialized with the model (e.g. in XMI).

15.3.1 OMG Diagram Definition and Interchange

DI (Data Interchange) is a specification out of the DD (Diagram Definition) standard [<https://www.omg.org/spec/DD>] which specifies how to interchange Diagrams in general for languages especially ones that are built from MOF. The SACMDiagram exchange is based on this standard whose diagram is reproduced below [from fig. 9.2 of the DD standard], Diagram Definition Specification Version 1.1.

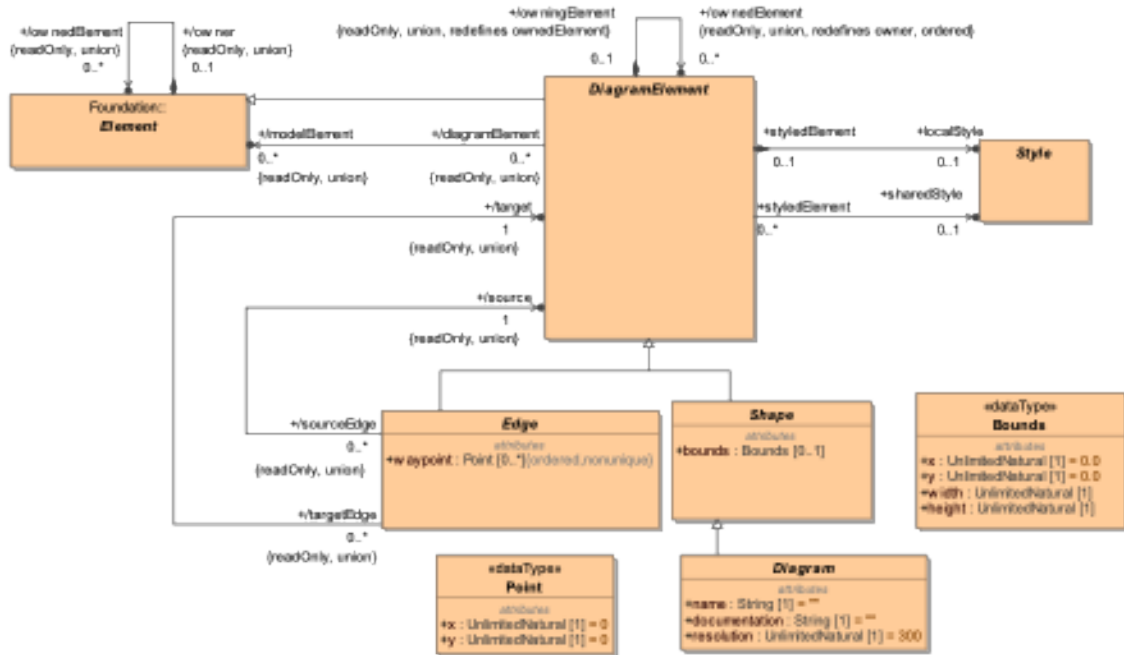


Figure 15.3.1 – OMG’s DI Standard Diagram

15.3.2 SACMView Model

SACMView allows for the specification of multiple types of Views that can be used to represent an AssuranceCase. SACMView itself is an abstract metaclass which can be extended to provide different modeling artifacts (e.g. Diagrams, Tables). SACMDiagram (also abstract) allows for the representation of a Diagram with SACMDiagramElements. There are specific concrete specializations of SACMDiagram which allow for various parts of Structured Assurance Case Diagrams to be separated on different diagrams.

15.3.5 SACMDiagramElement

SACMDiagramElements are the visible DiagramElements within a SACMDiagram.

Supertype

DiagramElement

Attributes

representation : MultiLangString [0..1] - representation (e.g. SVG) of the DiagramElement within the SACMDiagram. Language of representation should be specified in the MultiLangString.

Associations

element : SACMElement [0..*] {redefines modelElement} - SACMElements that this SACMDiagramElement represents in the SACMDiagram.

15.3.6 SACMDiagram Types

Four separate specific diagrams are defined that allows a general SACM diagram (StructuredAssuranceCaseDiagram) as well as three specific domain diagrams (StructuredAssuranceTerminologyDiagram, StructuredAssuranceArtifactDiagram, StructuredAssuranceArgumentDiagram). Each of these diagrams can have BaseElements on them, as well as specific concepts from each of the three domains in their respective diagrams.

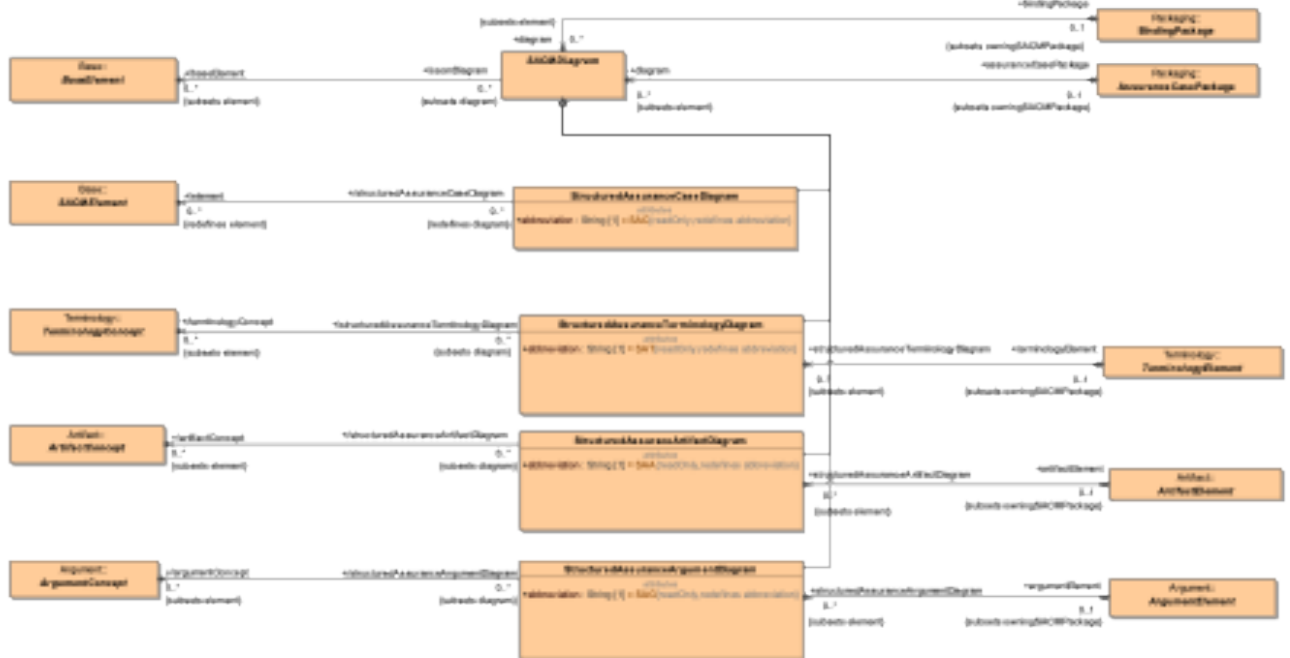


Figure 15.3.4 – SACMDiagramTypes

15.3.7 StructuredAssuranceCaseDiagram

StructuredAssuranceCaseDiagram allows any SACMElement to be represented on them.

Supertype

SACMDiagram

Attributes

abbreviation : String [1] = SAC {readOnly, redefines abbreviation}

AssociationEnds

/element : SACMElement [0..*] {redefines element} - Any SACMElement can be represented by SACMDiagramElements on a StructuredAssuranceCaseDiagram

15.3.8 StructuredAssuranceTerminologyDiagram

StructuredAssuranceTerminologyDiagram allows any BaseElement or TerminologyConcept to be represented on them.

Supertype

SACMDiagram

Attributes

abbreviation : String [1] = SAT {readOnly, redefines abbreviation}

AssociationEnds

/terminologyConcept : TerminologyConcept [0..*] {subsets element} - only TerminologyConcepts (along with BaseElements) should be represented by SACMDiagramElements on a StructuredAssuranceTerminologyDiagram

15.3.9 StructuredAssuranceArtifactDiagram

StructuredAssuranceArtifactDiagram allows any BaseElement or ArtifactConcept to be represented on them.

Supertype

SACMDiagram

Attributes

abbreviation : String [1] = SAA {readOnly, redefines abbreviation}

AssociationEnds

/artifactConcept : ArtifactConcept [0..*] {subsets element} - only ArtifactConcepts (along with BaseElements) should be represented by SACMDiagramElements on a StructuredAssuranceArtifactDiagram

15.3.10 StructuredAssuranceArgumentDiagram

StructuredAssuranceArgumentDiagram allows any BaseElement or ArgumentConcept to be represented on them.

Supertype

SACMDiagram

Attributes

abbreviation : String [1] = SAG {readOnly, redefines abbreviation}

AssociationEnds

/argumentConcept : ArgumentConcept [0..*] {subsets element} - only ArgumentConcepts (along with BaseElements) should be represented by SACMDiagramElements on a StructuredAssuranceArgumentDiagram

15.3.11 AssuranceCasePackage [Additions to 11.2]

AssociationEnds

diagram: SACMDiagram [0..*] {subsets element} - any SACMDiagram can be contained within an AssuranceCasePackage

15.3.12 BindingPackage [Additions to 10.3]

AssociationEnds

diagram: SACMDiagram [0..*] {subsets element} - any SACMDiagram can be contained within a BindingPackage

15.3.13 TerminologyElement [Additions to 12.2]

AssociationEnds

structuredAssuranceTerminologyDiagram : StructuredAssuranceTerminologyDiagram [0..*] {subsets element} - Only StructuredAssuranceTerminologyDiagrams can be contained in a TerminologyElement.

ArtifactElement [Additions to 9.7]**AssociationEnds**

structuredAssuranceArtifactDiagram : StructuredAssuranceArtifactDiagram [0..*] {subsets element} - Only StructuredAssuranceArtifactDiagrams can be contained in a ArtifactElement.

15.3.14 ArgumentElement [Additions to 13.2]**AssociationEnds**

structuredAssuranceArgumentDiagram : StructuredAssuranceArgumentDiagram [0..*] {subsets element} - Only StructuredAssuranceArgumentDiagrams can be contained in a ArgumentElement.