Annex G: Transformation from SACM 2.3 to SACM 2.4

(informative)

In general, the mapping of models from SACM 2.3 to SACM 2.4 is fairly straightforward and uses the same constructs. There have been some changes of attributes and AssociationEnd names to enhance consistency across the model. In addition, some concepts have been expanded to include a wider application to other elements in the model. For example, ImplementationConstraints were made into Constraints and can now apply to anything in the model..

G.1 Foundation

Foundation was added not as new elements (all of them are abstract) but as an aid to defining semantics of elements already in SACM 2.3. These semantics are nothing new, they come from UML (and KerML) concepts. These are now tied to understood elements (e.g. like UML Packages) so that SACM element meanings are not ambiguous. Include in this section is Element, DirectedRelationship, NamedElement, Namespaced, PackageableElement, and Package.

G.2 Base

1. TagValue is renamed NamedValue since Tags are already used in modeling terminology
2. NamedValues are separate objects from their SACMElements so multiple SACMElements can have reference to it
3. SACMConstraints are now separate objects, and can reference any object (including the language that SACM is embedded in)
4. needsDevelopment can be placed on any SACMElement
5. Visibility is now available on any SACMElement
6. SACMComments are now separate objects, and can reference any object (including the language that SACM is embedded in)
7. BaseElements (SACMComment, SACMConstraint, SACMDependency, NamedValue, Group) can be used with any other elements
8. Citation and Abstraction can be done through reference (to something in the local model) or IRI (to non-local model elements)
9. SACMDependency has been added to allow dependency relationships in Assurance Cases
10. The Concept of a SACMModel, to indicate whether or not something is part a SACMModel or not
11. The Concept of isPattern can be set on specialization of SACMModel, including Groups, SACMDiagrams, and SACMPackages
12. SACMModels can have sub-models defined
13. Model ownership is defined throughout the model, ModelElements can own any BaseElements
14. There is now only one Group element (rather than a Group per section), Groups can contain elements from any section

G.3 Packaging

1. Packages now have a much stricter rules on which kinds of packages can own other packages
2. AssuranceCasePackage can own subpackages of
   type: AssuranceCasePackage, ArgumentPackage, ArtifactPackage, TerminologyPackage, and BindingPackage
3. Section Packages (e.g. ArgumentPackage) can own another section package, or a section interface package (e.g. ArgumentInterfacePackage)
4. Section Interface Package can own another Section Interface Package
5. Section Packages and Interface Packages can own their Section Assets
6. Any Package can own BaseElements
7. There is only one BindingPackage rather than one for each section, and a BindingPackage can own any SACMElement

G.4 Terminology

1. MultiLangString is now the combination of LangString and MultiLangString into one Object
2. MultiLangString has been moved to the Terminology section

3. MultiLangString and ExpressionLangString are now types of content in the model (so things can now point to elements in Terminology)4
4. Categories can own Categories, Expressions, and Terms
5. Expressions can own Expressions and Term

G.5 Artifact

1. Artifact date has been changed to versionReleaseDate
2. Artifact can have a reference to an external document using externalDocumentIRI
3. Artifact can reference an internal part of a document using internalReference
4. Any ArtifactAsset can be owned by any other ArtifactAsset

G.6 Argument

1. Any ArgumentAsset can own any other ArgumentAsset
2. Artifact is a specialization of ArtifactRefrence
3. ArtifactReference references any SACMElement (sometimes an Artifact)
4. ArtifactRefence can represent evidence provided by a whole SACMElement (e.g. Package)
5. A SACMModel (Group, Package, Diagram) can be an Assertion
6. AssertionDeclaration is one of axiomatic, assumed, asserted
7. AssertedContext can tie an ArtifactReference (possibly an Artifact) to an ArgumentAsset
8. AssertedEvidence can tie an ArtifactRefernce (possibly an Artifact) to a Targetable (ArtifactReference or Assertion)
9. AssertedInference can tie an Assertion to a Targetable
10. Claim can refer to a whole SACMModel (e.g. Package)
11. Claim can define its subject(s) SACMElement through its association end subject
12. Any ArgumentConcept can be defeated in an argument
13. AssertedArtifactSupport was removed, In SACM 2.4, One can have an AssertedEvidence between one ArtifactReference and another, and therefore a separate Relationship is not necessary
14. AssertedArtifactContext was removed, In SACM 2.4, One can have an AssertedContext between one ArtifactReference and another, and therefore a separate Relationship is not necessary