The following changes should be made to the ModSpec Enterprise Architect UML model to bring it in-line with the data dictionary in Annex C

Specification

* Change the name of the Specification class to “Standard”
* Add an attribute for the “testSuite” association
* Change the name of the “class” association to “classes”
* Add an attribute for the “class” association.
* Change the name of the “reference” association to “references”
* Add an attribute for the “reference” association

Conformance Suite

* Change the name of the “class” association to “classes”

ConformanceClass

* Change the name of the “dependency” association to “dependencies”
* Change the name of the “requirements” association to “requirementsClass”
* Change the name of the “module” association to “modules”

RequirementsClass

* Change the name of the “dependency” association to “dependencies”
* Change the name of the “module” association to “modules”
* Change the cardinality of the target of the “module” association to “1..\*”.

RequirementsModule

* Change the name of the “requirement” association to “requirements”
* Change the data type for the “requirement” association to “requirement”

OR

* Change the data type in the data dictionary to “NormativeStatement”
* Add a Non-normative statement association.

NormativeStatement

* No change

Requirement

* Add an association for the “parts” attribute

InformativeStatement

* Create a new class with Recommendations and Permissions as subclasses.

Recommendation

* Fix misspelling of class name

ConformanceTest

* Add “abstract” attribute (Boolean)
* Change cardinality of “testMethod” attribute to “0..\*”
* Change data type of “testMethod” attribute to ConformanceTestMethod
* Change name of “reference” attribute to “references”
* Change cardinality of “reference” attribute to “0..\*”
* Change data type of “reference” attribute to RichText
* Change name of requirement association to “requirements”
* Change requirement association to a directed or by-directional association (adds arrowheads)

StandardizationTarget

* Change target of “instantiate” association to a “type” association (1..1)

StandardizationTargetType

* Populate this class in the data dictionary

ConformanceModule

* Add this class to the data dictionary

ConformanceTestMethod

* Add this class to the data dictionary

TestType

* Add this class to the data dictionary

Principal

* Add this class to the data dictionary

**Semantically ordered definitions (please see comments)**

[Clause 4](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#cls-4) formally defines the terms used in the conformance tests in alphabetical order. It may be easier to understand the more significant terms in the following less formal definitions arranged in a bottom-up order:

1. a *standardization target type* ([Clause 4.27](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target-type)) is a type of entity about which a standard is written. An instance of a *standardization target type* ([Clause 4.27](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target-type)) is a *standardization target* ([Clause 4.26](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target)). A standard may address multiple targets in separate [*conformance classes*](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#conformance-class).
2. a *requirement* ([Clause 4.21](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirement)) is a statement of a condition to be satisfied by a single *standardization target type* ([Clause 4.27](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target-type)), and it must be stated in “normative” language.
3. a *conformance test* ([Clause 4.3](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test)) checks if a set of *requirements* ([Clause 4.21](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirement)) are met (**pass**) or not met (**fail**) by a *standardization target* ([Clause 4.26](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target)). The relationship between *conformance tests* ([Clause 4.3](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test)) and *requirements* ([Clause 4.21](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirement)) is many-to-many.
4. all *conformance tests* ([Clause 4.3](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test)) are graded as **pass** or **fail** against each instance of the *standardization target* ([Clause 4.26](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target)).
5. a *requirement* ([Clause 4.21](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirement)) is associated to one *conformance test* ([Clause 4.3](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test)).
6. a *recommendation* ([Clause 4.20](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-recommendation)) is a suggestion and is not associated to any *conformance test* ([Clause 4.3](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test)).
7. a *requirements class* ([Clause 4.22](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirements-class)) is a set of one or more *requirements* ([Clause 4.21](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirement)) all with the same *standardization target type* ([Clause 4.27](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target-type)).
8. a *conformance (test) class* ([Clause 4.6](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test-class)) is a collection of *conformance tests* ([Clause 4.3](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test)) that are associated to and only to the requirements in a corresponding *requirements class* ([Clause 4.22](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirements-class)).
9. a *conformance (test) module* ([Clause 4.5](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test-module)) is also collection of **term conformance test classes not resolved via ID conformance-test-classes** that group *conformance tests* ([Clause 4.3](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test)) on a single *standardization target type* ([Clause 4.27](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target-type)).
10. a **conformant implementation** is a *standardization target type* ([Clause 4.27](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target-type)) that has successfully passed all tests in a specified *conformance (test) class* ([Clause 4.6](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test-class)) and received a *certificate of conformance* ([Clause 4.2](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-certificate-of-conformance))
11. the *core requirements class* ([Clause 4.8](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-core-requirements-class)) of a standard is the minimal set of *requirements* ([Clause 4.21](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirement)) which must be supported by all **conformant implementations**. If a standard addresses multiple *standardization target types* ([Clause 4.27](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target-type)), it may have a **core** for each **target type**.
12. an **extension** of a *requirements class* ([Clause 4.22](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirements-class)) is an additional *requirements class* ([Clause 4.22](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirements-class)) (the extension) that adds additional *requirements* ([Clause 4.21](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirement)) to the first *requirements class* ([Clause 4.22](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-requirements-class)) (the **base requirements class** being extended). The extension is said to be dependent on the **base**. Any *conformance test class* ([Clause 4.6](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-conformance-test-class)) must identify all its dependencies during the execution of conformance tests against a candidate *standardization target* ([Clause 4.26](file:///C:\WorkSpace\GitHub\ogc-modspec%20-%20Copy\sources\document.html#term-standardization-target)).

Update change list based on Carl’s updates to Annex C.

Create a new class for non-normative Statements. Recommendations and Permissions are non-normative statements. Non-normative statements are associated with a normative statement (this association defines the context in which the non-normative statement is applicable).

Hierarchy – requirement module, requirement class, requirement; conformance module, conformance class, conformance test.

Action: update Annex C, verify that definitions are correct across the entire document, verify requirements are correct, work on non-normative text.

Include 19106 profiling in Mod Spec as well?

Make sure you are subscribed to the modspec reflector.