**Mapping SKOS Platform Classification Scheme to Vocab DB**

This mapping is associated with ‘SkosSamplePlatformClassificationScheme.rdf’. The file essentially declares a Classification Scheme, included Categories and Platform Type Terms associated with each category.

|  |  |  |
| --- | --- | --- |
| **Skos Elements** | **Vocabulary DB Tables and [Column Names]** | **Notes/DB Schema Amendments** |
| Skos:ConceptScheme | Classification\_Scheme[Classification\_Scheme\_Name] |  |
| dc:title | Classification\_Scheme [Title] |  |
| dc:description | Classification\_Scheme [Classification\_Scheme\_Description] |  |
| dc:creator<foaf:organization<foaf:name/>/> | Classification\_Scheme [Organisation\_Name\_Id] then Organization\_Name [Organisation\_Name] |  |
| dc:publisher<foaf:name/> | Classification\_Scheme [Person\_Id] then Person [Person\_Name] |  |
| dc:rights | No mapping in DB | Should have a standard rights statement that applies to all AODN vocabs that can be inserted directly. |
| dcterms:issued | Classification\_Scheme [Date\_Added] |  |
| Skos:TopConcept | Classification\_Scheme\_Category [Classification\_Scheme\_Category\_Name] | The Classification\_Scheme\_Category\_Names selected should be those that are associated with the chosen Classification\_Scheme\_Name {these associations can be found in the Classification\_Scheme\_Category Table and/or in the Classification\_Scheme\_Association Table) |
| Skos:prefLabel | Classification\_Scheme\_Category [Label] |  |
| Skos:definition | Classification\_Scheme\_Category [Classification\_Scheme\_Category\_Definition] |  |
| dcterms:issued | Classification\_Scheme\_Category [Date\_Added] |  |
| Skos:inScheme | When skos:inScheme appears as a property of skos:TopConcept it refers to the Classification\_Scheme\_Name for the skos:ConceptScheme being described. |  |
| Skos:Concept | Vocabulary\_Term [Vocabulary\_Term\_UID] | Vocabulary\_Term\_UIDs are selected and found by using the Classification\_Scheme\_Association Table |
| Skos:inScheme | When skos:inScheme appears initially as a property of skos:Concept the resource referred to is the Classification\_Scheme\_Name for the skos:ConceptScheme being described. |  |
| Skos:broader | The resource being referred to is the Classification\_Scheme\_Category [Classification\_Scheme\_Category\_Name] | This resource is found by looking through the Classification\_Scheme\_Association Table and matching the Vocabulary\_Term\_Code (associated with the Vocabulary\_Term\_UID) with its Classification\_Scheme\_Category\_Name (represented by a Classification\_Scheme\_Category\_Id) |
| Skos:inscheme | When skos:inScheme appears for a second time as a property of skos:Concept the resource referred to is the Register\_UID for the Register in which the term being described has been taken. The Register Name is found in the Vocabulary\_Term Table and this links to a Vocabulary\_Register Table (in which the Register\_UID is located). |  |
| Skos:prefLabel | Vocabulary\_Term [Vocabulary\_Term\_Name] |  |
| Skos:definition | Vocabulary\_Term [Vocabulary\_Term\_Definition] |  |
| dc:source | Vocabulary\_Term [Reference\_Source\_Id] then Reference\_Source [Citation\_String]  OR  Register\_Owner [Register\_Name] | If Reference\_Source\_Id in the Vocabulary Term table is not null, then get the Reference\_Source\_Id and check the Reference\_Source table for the Citation\_String (e.g. L06 SeaVox Platform Devices).  Else, look-up Register\_Name in the Register\_Owner table. |
| dc:publisher | Reference\_Source [Organisation\_Name\_Id] then Organisation\_Name [Organisation\_Name]  OR  Register\_Owner [Organisation\_Name\_Id] then Organisation\_Name [Organisation\_Name] | If Reference\_Source\_Id in the Vocabulary Term table is not null, then get the Reference\_Source\_Id and check the Reference\_Source table for the Organisation\_Name\_Id and look-up the Organisation\_Name in the Organisation Name  Else, look-up Organisation\_Name\_Id in the Register\_Owner table and find Organisation\_name in the Organisation Name table. |

**Mapping SKOS Platform Vocabulary Scheme to Vocab DB**

This mapping is associated with ‘SkosAODNPlatformVocabulary.rdf’. The file essentially declares a Platform Vocabulary Scheme, which includes Platform Type Terms and Platform Instances.

|  |  |  |
| --- | --- | --- |
| **Skos Elements** | **Vocabulary DB Tables and [Column Names]** | **Notes/DB Schema Amendments** |
| Skos:ConceptScheme | Vocabulary\_Register [Register\_UID] |  |
| dc:title | Vocabulary\_Register [Register\_Name] |  |
| dc:description | Vocabulary\_Register [Register\_Content\_Summary] |  |
| dc:creator<foaf:organization<foaf:name/>/> | Register\_Owner[Organisation\_Name\_Id] then Organization\_Name [Organisation\_Name] |  |
| dc:publisher<foaf:name/> | Register\_Owner[Person\_Id] then Person [Person\_Name] |  |
| dc:rights | No mapping in DB | Should have a standard rights statement that applies to all AODN vocabs that can be inserted directly. |
| dcterms:hasVersion | Vocabulary\_Register [Register\_Version] | Change Register Version to be a URI (or URL) which is actually the path to the versioned vocabulary file (e.g. http//vocab/aodn.org.au/Version 1.0/Platform/). This versioned URL IS NOT used by default to reference the current Platform scheme. The URL given at Register\_UID is used instead for routinely accessing content from this scheme. In contrast this URI accessing the described content, for this version, would live in an archive somewhere so that we can provide access to previous versions of the scheme when required. |
| dcterms:issued | Vocabulary\_Register [Register\_Date] |  |
| Skos:Concept | Vocabulary\_Term [Vocabulary\_Term\_UID] | This is the Platform Type. Skos:concept declarations for Platform Type (in green) are repeated until there are no more Platforms in the Vocabulary. |
| Skos:inScheme | Vocabulary\_Register [Register\_UID] |  |
| Skos:prefLabel | Vocabulary\_Term [Vocabulary\_Term\_Name] |  |
| Skos:definition | Vocabulary\_Term [Vocabulary\_Term\_Definition] |  |
| dc:source | Vocabulary\_Term [Reference\_Source\_Id] then Reference\_Source [Citation\_String]  OR  Register\_Owner [Register\_Name] | If Reference\_Source\_Id in the Vocabulary Term table is not null, then get the Reference\_Source\_Id and check the Reference\_Source table for the Citation\_String (e.g. L06 SeaVox Platform Devices), else look-up Register\_Name in the Register\_Owner table. |
| dc:publisher | Reference\_Source [Organisation\_Name\_Id] then Organisation\_Name [Organisation\_Name]  OR  Register\_Owner [Organisation\_Name\_Id] then Organisation\_Name [Organisation\_Name] | If Reference\_Source\_Id in the Vocabulary Term table is not null, then get the Reference\_Source\_Id and check the Reference\_Source table for the Organisation\_Name\_Id and look-up the Organisation\_Name in the Organisation Name  Else, look-up Organisation\_Name\_Id in the Register\_Owner table and find Organisation\_name in the Organisation Name table. |
| Skos:narrower | The resource being referred to is the Internal\_Association\_Terms [Object\_Term\_Id, Vocabulary\_Term\_Name]  where Subject\_Term\_Id = Vocabulary Term\_Id for the skos\_concept above,  and Association\_Type\_Name = “isInstanceOf” | Refers to a Platform Instance. The number of times this ‘skos:narrower’ property is written depends on how many Platform Instances have an “instanceOf’ association with the Platform Type (i.e. the skos:concept above which is the subject of the statement). |
| Skos:concept | Using the Internal\_Association\_Terms [Object\_Term\_Id] get the Vocabulary\_Term [Vocabulary\_Term\_Name] where the Subject\_Term\_Id = Vocabulary\_Term\_Id for a skos:concept (Platform Type) and Association\_Type\_Name = “isInstanceOf”. | This is the Platform Instance. Skos:concept declarations for Platform Instance (i.e. all fields below) are repeated until there are no more Instances in the Vocabulary that are associated with the Platform Types declared above. |
| Skos:inScheme | Vocabulary\_Register [Register\_UID] |  |
| Skos:prefLabel | Vocabulary\_Term [Vocabulary\_Term\_Name] |  |
| Skos:definition | Vocabulary\_Term [Vocabulary\_Term\_Definition] |  |
| dc:source | Vocabulary\_Term [Reference\_Source\_Id] then Reference\_Source [Citation\_String]  OR  Register\_Owner [Register\_Name] | If Reference\_Source\_Id in the Vocabulary Term table is not null, then get the Reference\_Source\_Id and check the Reference\_Source table for the Citation\_String (e.g. L06 SeaVox Platform Devices), else look-up Register\_Name in the Register\_Owner table. |
| dc:publisher | Reference\_Source [Organisation\_Name\_Id] then Organisation\_Name [Organisation\_Name]  OR  Register\_Owner [Organisation\_Name\_Id] then Organisation\_Name [Organisation\_Name] | If Reference\_Source\_Id in the Vocabulary Term table is not null, then get the Reference\_Source\_Id and check the Reference\_Source table for the Organisation\_Name\_Id and look-up the Organisation\_Name in the Organisation Name  Else, look-up Organisation\_Name\_Id in the Register\_Owner table and find Organisation\_Name in the Organisation Name table. |

**Additional Notes**

* Need to add dcterms:hasVersion to Platform Scheme pattern (in rdf file). I’m hoping this will partially solve our versioning issue in that we will have a mechanism to refer to a URI (preferably URL) which tells us the version of the vocab (for our own publishing and access purposes) but it doesn’t affect the Vocabulary URL which is used to refer to the current version of things (which is a version-less URL). Unlike the UK example I provided earlier (and again below), I believe the first dct:hasVersion refers to the current version of the vocabulary and subsequent statements using this property refer to previous versions. For example:

<<http://education.data.gov.uk/doc/inspection-report/12345>>

rdfs:label "Inspection Report for Such-and-Such School"@en ;

dct:hasVersion

<<http://education.data.gov.uk/doc/inspection-report/12345/2009>> ,

<<http://education.data.gov.uk/doc/inspection-report/12345/2006>> ,

<<http://education.data.gov.uk/doc/inspection-report/12345/2003>> ,

... .

<<http://education.data.gov.uk/doc/inspection-report/12345/2009>>

rdfs:label "2009 Inspection Report for Such-and-Such School"@en ;

dct:created "2009-10-18"^^xsd:date ;

dct:replaces <<http://education.data.gov.uk/doc/inspection-report/12345/2006>> ;

dct:isVersionOf <<http://education.data.gov.uk/doc/inspection-report/12345>> .

We should, however, then follow the second part of the example above to specifically document which immediate previous version this version replaces (and when that superceded version was created). This has not been encoded in the example patterns because as yet we only have unpublished beta version of our vocabs. This info is for future governance of vocabs.

* Need to reference some BODC skos files to see how they refer to themselves (I think they use ‘British Oceanographic Data Centre’ for the owner/creator of concept schemes and ‘Natural Environment Research Council’ is the ‘publisher’ of schemes). This is important when we update content in the Vocab database.
* There were some mistakes in syntax of RDF files that I provided to Craig through cut and paste errors, so I have updated these files.
* Subject\_Term\_Id and Object\_Term\_Id are re-used internal codes (from Vocabulary\_Term\_Code).
* I consulted with Craig on the issue of syntax namespace pattern for URIs (URLs). He is consulting the software development team with a view to giving us a definitive answer next week, but it is likely that namespaces will start with:

http://vocab.aodn.org.au/…./…./…/

Where ‘vocab’ indicates the type of server or service (vocabulary service in our case) and ‘aodn.org.au’ is the domain space that IMOS has control over.

Recap on previous suggestion for the remainder of the path for the different things we are managing (Vocabulary Schemes for different Vocabulary Types, Vocabulary Terms, Classification Schemes and Classification Categories, Classification Schemes with embedded Vocabularies).

Suggested syntax for:

**Vocabulary Terms:** [http://vocab.aodn.org.au/def/VocabularyType/TermType[optional]/TermID/](http://vocab.aodn.org.au/def/VocabularyType/TermType%5boptional%5d/TermID/)

**Vocabulary Schemes:** [http://vocab.aodn.org.au/def/VocabularyType/TermType[optional]/](http://vocab.aodn.org.au/def/VocabularyType/TermType%5boptional%5d/)

**Classification Scheme Name:** <http://vocab.aodn.org.au/def/ClassScheme/Name/>

**Classification Scheme Category:** <http://vocab.aodn.org.au/def/ClassScheme/Name/Category/TermID/>

* We need to remember that there needs to be a new register for each type of Vocabulary (or Classification Scheme with an embedded vocabulary)