**High Level Functional Requirements For Vocabulary and Term Publishing**

This is a dot point list of basic functional requirements for a tool (or a tool component) designed to publish vocabularies and their terms so that they can be searched and retrieved by end users. Ideally these requirements should be read in conjunction with the use-cases already drafted specifically for ‘vocabulary management and authoring’ because there is an overlap in requirements.

1. Main function is to provide a resolvable endpoint for a vocabulary and its included terms (and details) using persistent identifiers.\*
2. Resolvable content should be structured (or at least be able to be queried) using an RDF/SKOS encoding model. Content may however be adorned by other languages/metadata models (e.g. RDFS, OWL, Dublin Core). \*
3. It should be possible to access vocabularies and their terms via an (administratively) customisable Web-client interface and service interfaces. \*
4. Most users require read only access to content. \*
5. There should ideally be a service interface that is REST-based\* and a SPARQL service end-point.
6. The publishing and retrieval service should offer and receive re-direction(s) so that vocabularies or terms hosted on different domains (under differing content authorities) can still be accessed via the service (if desired). \*
7. The Web-client should support some basic ‘canned’ querying (e.g. free text search against concept, collection and scheme ‘labels’; traversing a named vocabulary via hypertext links to explore included terms, their details and any matches or mappings to other published vocabularies). \*
8. The Web-Client should be able to display in a user-friendly way (e.g. using tables or forms) vocabulary and term details (e.g. scheme, collection and concept labels; alt labels, its type, description, membership, relations [including some Dublin core relations such as ‘publisher’ and ‘owner’ and revision info]). \*
9. The Web-Client should be able to display categorized (classified) lists of discoverable content (e.g. all vocabs by provided by owner X; all terms in vocab Y)
10. The Web-Client should offer different formats in which to download vocabularies or their terms (e.g. RDF\*, text\*, json, html\*)
11. The Web-Client should offer some statistics for users on the type and volume of content available (e.g., number of vocabularies that can be accessed and the number of terms in each vocabulary).
12. The publishing and retrieval service should be capable of being configured to dynamically read one or more repository sources to access content that needs to be published. \*
13. Response times for retrieving queried content should be ‘user-tolerable’. \*
14. System should provide an administrative console/configuration files to enable simple maintenance and administration (e.g., small changes to Web-client interface displays and supported queries; to detect and fix broken links in client-based hypertext; detecting missing details in retrieved content – indicating content needs moderating/validating; provide basic statistics on service usage).
15. There should be ‘meaningful’ error messaging provided in response to service calls that cannot be satisfied (or which have been framed incorrectly). \*