

Title	CLARIN-PLUS CCR analysis
Version	1
Author(s)	Menzo Windhouwer (CLARIN ERIC/Meertens Institute)
Date	2015-11-26
Status	Final
Distribution	Public
ID	CE-2015-0688



1 Introduction

Until 2015 CLARIN has been using the ISOcat data category registry and the CLAVAS vocabulary registry. In 2015 CLARIN started using the new CLARIN Concept Registry (CCR) based on the same software platform as CLAVAS: OpenSKOS. The OpenSKOS platform had been changed a bit to make it a suitable replacement for ISOcat. And when the CCR was functional the data categories used by CLARIN were imported into it.

This task deals with these and additional changes, which are needed to better fit OpenSKOS into CLARIN's metadata workflow (CMDI), *e.g.*, relations among concepts, easier import of new concepts, concept lifecycle management and browse and search facilities for unauthenticated users. These changes need to be discussed and aligned with the OpenSKOS community to leverage the benefits of shared development. It is also important to anticipate future migration paths to either new versions of OpenSKOS or to a follow-up platform. This first T 2.2.3 milestone is an analysis of the current status and needs, which will then be followed by the actual development.

2 OpenSKOS community

The Meertens Institute, the host of the CCR and CLAVAS and the executor of this task, has started to have regular meetings with other (potential) users of OpenSKOS (mainly from the Dutch cultural heritage domain), *e.g.*, Sound and Vision, the Cultural Heritage Agency, Europeana and the National Library of the Netherlands. In these meetings the current usage and role of OpenSKOS in the cultural heritage domain and CLARIN are discussed and new requirements are defined, aligned and prioritized. This has resulted in a list of requirements (see Appendix A) for future OpenSKOS development. In the second half of 2015 Picturae (the original developer of OpenSKOS) started working on OpenSKOS 2, which implements some of these requirements. For this CLARIN-PLUS task the Meertens Institute already cooperates with Picturae and Sound and Vision to monitor and test the ongoing development and to prepare the merge of the CLARIN features developed by the Meertens Institute. This work aligns the various OpenSKOS forks that currently exist into one main OpenSKOS 2 codebase¹.

2.1 OpenSKOS user stories

The OpenSKOS user group has collected and prioritized a list of user stories shown in Appendix A. This list includes:

- The development on an OpenSKOS fork by the Meertens Institute to enable the transition from ISOcat to the CCR (marked in the table as green rows owned by the tenant CLARIN, *i.e.*, D-11, D-12, W-4 and W-5). As one of the first development actions these changes will be propagated from the CLARIN OpenSKOS fork to the OpenSKOS 2 main branch. This will be a manual merge as

¹ To be hosted at <https://github.com/OpenSKOS>

the code bases have drifted apart considerably. Also the browser (W-4) needs to be reimplemented due to the switch to a triple store in OpenSKOS 2 and the requirement to base it on the API (W-2).

- D-11: support for handles
- D-12: support for SKOS collections
- W-4: read-only browser
- W-5: Shibboleth-based login

- The development on OpenSKOS 2 by Picturae and Sound and Vision (marked in the table as green rows owned by the tenant BENG,² *i.e.*, D-10, D-14, D-17, D-18 and A-3). Currently the Meertens Institute already monitors and tests the results of these developments.

- D-10: SPARQL endpoint
- D-14: support for SKOS-XL
- D-17: entailments for relations
- D-18: complete exports
- A-3: backwards compatibility

- Not yet implemented requirements by CLARIN, which mainly focus on concept life cycle and relationship management (marked in the table as orange rows, *i.e.*, D-2, D-3, D-4, D-5, D-6, D-7, D-9, D-13, W-2, W-6 and W-7).

- Concept life cycle:

- D-2: concept genealogy
- D-3: explicit links to superseding concepts

- Relation management:

- D-4: group relationships
- D-5: attribution of relationships
- D-6: non-SKOS relationships
- D-7: relationships with external entities

- Generic issues:

- D-9: positive locking
- D-13: download a SKOS document
- W-2: use the API instead of UI specific backdoors
- W-6: make URLs in definitions, examples, etc. clickable
- W-7: allow seeing and editing non-SKOS properties

- Finally there are some CLARIN requirements with lower priorities (D-8, W-3) and requirements from other OpenSKOS users, which can be addressed if time allows.

- D-8: keep concepts under construction private
- W-3: discuss concepts

² BENG = Sound and Vision

3 Planning

Year	2015		2016			
Month	Q3	Q4	Q1	Q2	Q3	Q4
P1 testing						
M2.3 Analysis						
P2 Merge						
P3 Life cycle						
P4 Relationships						
D2.6 Reinforced concept registry						

This planning shows 4 major development phases:

- P1: The current phase mainly consists of testing and monitoring ongoing developments aimed at providing T 2.2.3 with a stable start on continuing the development of OpenSKOS 2.
- P2: A next phase where the already developed features are integrated into OpenSKOS 2, which for some features actually means a reimplementation, *e.g.*, the browser.
- P3: In this phase development focuses on supporting life cycle management for concepts and resolving some generic issues.
- P4: In this longer phase development focuses on internal and external relation management between concepts and the attribution of these relationships.

Notes:

- After P2 a more refined planning for the remaining development phases will be possible.
- Although the focus is mainly on the CCR the CLAVAS vocabulary service should also benefit from the release of OpenSKOS 2 in the P2 phase, especially as in 2016 Q1 CMDI 1.2 will be rolled out, which will draw new attention to CLAVAS.
- Although the planning spans 16 months the actual development effort has been limited to 6PM.

Appendix A

Nr.	Tenant	Stories	MoSCoW ³
		Data store, data model (and extended data model)	
D-1	RCE ⁴	As information manager, I would like a connection between OpenSKOS and my own thesaurus management application. So I can update the OpenSKOS version of my thesauri in real-time.	M

³ **M**ust have, **S**hould have, **C**ould have, **W**on't have (this time)

⁴ RCE = Cultural Heritage Agency

Nr.	Tenant	Stories	MoSCoW ³
D-2	CLARIN BENG ⁵	As a content coordinator I would like to have a status model that supports to store which APPROVED concept supersedes the OBSOLETE concept.	S
D-3	CLARIN BENG	As a developer, I would like the target URI for a REDIRECTED concept to be stored explicitly in an object property (like dcterms:isReplacedBy), rather than hidden in a changeNote.	S
D-4	CLARIN	As a linguistic researcher I need to be able to know which relations/alignments belong together in one of the several possible taxonomies. So you can access/export/visualize a coherent taxonomy.	M
D-5	CLARIN BENG	As a linguistic researcher or content coordinator I need to be able to access/store provenance information bound to a set of relations/alignments, <i>e.g.</i> , created by version X of algorithm Y or person Z.	M
D-6	CLARIN	As a content coordinator I need to be able to specify new relation/alignment types (outside of SKOS), <i>e.g.</i> , antonym from a WordNet vocabulary.	M
D-7	CLARIN	As a content coordinator I need to be able to enter relationships with external URIs, <i>e.g.</i> , of concepts from other concept registries.	S
D-8	CLARIN	As a content coordinator I need to be able to publish concepts that are first only visible for logged in users and later for everybody.	C
D-9	CLARIN BENG	As a content coordinator I need (positive) locking of concepts while editing, so that no updates will be lost in case of concurrent access.	M
D-10	CLARIN BENG	As an external partner in the Semantic Web community I would like to interact with the system via a SPARQL endpoint rather than with the API.	M
D-11	CLARIN	As a content coordinator I need handles to be assigned to new concepts, concept schemes and SKOS collections.	M
D-12	CLARIN	As a content coordinator I need to be able to create and maintain SKOS collections.	M
D-13	CLARIN BENG	As a (semantic web) user or external developer concept schemes or SKOS collections (and their concepts) should be downloadable as RDF documents (without using OAI-PMH).	S
D-14	BENG	As developer of an external system I want to be able to identify labels with unique identifier, so that I know exactly what has changed if downloading an update over OAI-PMH and can update	M

⁵ BENG = Sound and Vision

Nr.	Tenant	Stories	MoSCoW ³
		the labels within my own system.	
D-15	BENG	As (external) developer of a third party system I want to be able to know what has happened to (and what is the current status of) a candidate concept I proposed using the OpenSKOS Create API.	M
D-16	BENG	As developer I want to be able to access the configuration of specific search profiles, so that I can get the same result list from the API as a user gets in the editor.	
D-17	BENG	As administrator I need symmetric, inverse and transitive relations (entailments) for at least all SKOS relations.	M
D-18	BENG	As administrator, when exporting the data, I need every concept to contain all relations (also the symmetrical ones).	M
D-19	BENG	As organization I will not allow privacy sensitive information, to be exposed to the outside world. Contrary from outside information, <i>e.g.</i> , publicly available information like the concepts in the repository, that can be retrieved via the API and the OAI-PMH. It must be configurable what properties (or named graphs) are exposed.	S
		API	
A-1	BENG	As developer I want to be able to upload a file with SKOS matches/reasons/alignments between my thesaurus and another one.	S
A-2	BENG	As developer I want to be able to upload and use an alignment that I exported from Cultuurlink in OpenSKOS.	S
A-3	BENG	As a developer I need the OpenSKOS-XL API to be backwards compatible (e.g. so that the label searches will returns concepts). External systems depend on the API functionality, so changes to the underlying platform preferably don't change the API.	M
A-4	BENG	As front end users (in general) I need the auto completion API to be very fast.	M
A-5	BENG	As thesaurus manager I need the results to be ranked according to the contents of certain fields, e.g.	
		Web application	
W-1	RCE	As a thesaurus manager of the RCE, I would like to offer my end-users a user-friendly way to browse the thesauri and lookup the contents of the thesauri without logging in.	M
W-2	CLARIN	As a developer I would like increased stability by having the UI use the API	S

Nr.	Tenant	Stories	MoSCoW ³
W-3	CLARIN	As a content coordinator or linguistic researcher I would like to be able to post comments on concepts.	C
W-4	CLARIN	As a linguistic researcher I need read-only access via a faceted browser UI without the need to login.	M
W-5	CLARIN	A content coordinator I want to login to the editor using Shibboleth.	M
W-6	CLARIN	As an user I would like URIs in a textual field to be clickable	S
W-7	CLARIN	As an editor I would like to be able to see and edit additional (non-SKOS) properties (incl. object properties for relationships/alignments) in the editor	M
W-8	BENG	As an (thesaurus) administrator I want to be able to approve or delete concepts.	M
W-9	BENG	As an (thesaurus) administrator I will be able to assign roles to users. The rights are fixed per role (user, editor, administrator).	M
W-10	BENG	As an administrator I will be able to manage search profiles	M
W-11	BENG	As an administrator I will be able to attach fixed search profiles to (individual) users.	M
		Software provider	
S-1	BENG	As a product owner (for the editor) I want to be able to manage changes and minimize malfunctions.	