Discovering software resources in CLARIN

Jan Odijk
UiL-OTS
Utrecht University, the Netherlands
j.odijk@uu.nl

Abstract

We present a CMDI profile for the description of software that enables discovery of the software and formal documentation of aspects of the software, and a proposal for faceted search in metadata for software. We have tested the profile by making metadata for over 70 pieces of software. The profile forms an excellent basis for formally describing properties of the software, and for a faceted search dedicated to software which enables better discoverability of software in the CLARIN infrastructure.

1 Introduction

Enabling the easy discovery of resources is an important goal of CLARIN. The Virtual Language Observatory (VLO) serves this purpose, but it is currently mostly suited for the discovery of *data*. Discovering *software* is not so easy in the current VLO. In order to address this issue we present (1) a CMDI profile for the description of software that enables discovery of the software and formal documentation of aspects of the software, and (2) a proposal for faceted search in metadata for software. We have tested the profile by making metadata for over 70 pieces of software. We describe how we ensured the quality of these metadata descriptions. We describe how we are testing the proposed faceted search. We propose to add this faceted search to the VLO, and show how metadata curation software, combined with provided metadata curation files, can curate existing metadata descriptions for software using other profiles to make them suited for such faceted search.

2 Metadata Profile CLARINSoftwareDescription

The ClarinSoftwareDescription (CSD) profile[1] enables one to describe information about software in accordance with the CMDI metadata framework used in CLARIN (Broeder et al., 2012). The profile has been set up in such a way that it enables (1) the description of properties that support discovery of the resource, and (2) the description of properties for documenting the resource, in as formal a manner as possible.

We briefly describe the major components and elements of the profile. More details will be included in the actual presentation and the full paper. The elements crucial for finding the resource are dealt with in more detail in section 5.

The profile consists of the CMDI components *Generalinfo*, *SoftwareFunction*, *SoftwareImplementation*, *Access*, *ResourceDocumentation*, *SoftwareDevelopment*, *TechnicalInfo*, *Service* and *LRS*.

The component *Generalinfo* is an extension of the component *cmdi-generalinfo*[2] with elements for specifying the *CLARIN Centre* hosting the resource and the *national project(s)* in which it has been made part of the CLARIN infrastructure.

The *SoftwareFunction* component enables one to describe the function of the software in terms of the closed vocabulary elements *tool category, tool tasks*, *research phase(s)* (for which it is most relevant),

[1] clarin.eu:cr1:p_1342181139640.

[2] clarin.eu:cr1:c_1342181139620.

research domains and, for the linguistics domain, relevant *linguistic subdisciplines* for which it was originally developed.[3]

The *SoftwareImplementation* component enables one to describe information for users on the implementation and installation of the software. The most important components are for the description of the *interface*, the *input* and the *output* of the software.

The *Access* component enables one to describe information about the availability and accessibility of the resource.

The *ResourceDocumentation* component enables one to describe the documentation of the resource. The *SoftwareDevelopment* component is intended for information on the history and development of the software. The *TechnicalInfo* component enables one to describe technical information on a resource and is mainly aimed at developers.

The *Service* component (CLARIN-NL Web Service description) is intended for describing properties of web services. It is compatible with the CLARIN CMDI core model for Web Service description version 1.0.2.[4]

The *LRS* component is intended for the description of the properties of a particular task for the CLARIN Language Resource SwitchBoard (CLRS, (Zinn, 2016)). Multiple LRS components can be present. It is our viewpoint that specifications for an application for inclusion in the CLRS registry[5] should be derivable from the metadata for this application. This was not the case for the CSD profile when the CLRS came into existence, so we added a component to offer facilities for supplying the missing information. We devised a script to turn a CSD-compatible metadata record that contains an LRS component into the format required for the CLRS and tested it successfully with the Frog web service and application (van den Bosch et al., 2007).[6]

2.1 Semantics

Many of the profile's components, elements and their possible values have a semantic definition by a link to an entry in the CLARIN Concept Registry (CCR, (Schuurman et al., 2016)).[7] For the ones that were lacking we created definitions and provided other relevant information required for inclusion into the CCR. We submitted this file (2017-09-08), in the format required, to the maintainers of the CCR. However, the CCR coordinators'[8] mill runs slowly, and so far none of them have been incorporated in the CCR. After our submission to the CCR, we made some new modifications to the profile, so there are new elements and values for which the semantics does not exist yet.

2.2 Comparison with other profiles for software

There are about 20 profiles for the description of software in the CLARIN Component Registry (as determined on 2017-09-29), but most are not in use or in use for a single description only. The only profiles that are used for multiple software resources are *ToolProfile*[9] (49 resources), *WeblichtWebService*[10] (287 resources), *resourceInfo*[11] with the value *toolService* for the element *resourceType* (68 resources), and *OLAC-DcmiTerms*[12] (189 software resources)..

In the presentation and in the full paper we will make a detailed comparison between these profiles. Here we mention the most important differences: (1) the CSD profile is fully dedicated to the description of software (v. the *OLAC-DcmiTerms* profile); (2) the CSD profile can be used to describe any type of software (v. *WebLichtWebservice*); (3) it offers more elements, and more formalised elements than the other profiles, not only elements useful for discovery but also for (formalised) documentation.

[3] which, of course, does not preclude its use in other research domains that were not foreseen during development.

[4] This component was created by Menzo Windhouwer, and adapted to the requirements of CMDI version 1.2.

[5] `https://github.com/clarin-eric/LRSwitchboard/blob/master/app/back-end/Registry.js`

[6] https://languagemachines.github.io/frog/

[7] `https://concepts.clarin.eu/ccr/browser/.`

[8] https://www.clarin.eu/content/concept-registry-coordinators

[9] `clarin.eu:cr1:p_1290431694581.`

[10] `clarin.eu:cr1:p_1320657629644228.`

[11] `clarin.eu:cr1:p_1360931019836.`

[12] `clarin.eu:cr1:p_1288172614026.`

3 Metadata Descriptions using the CSD profile

We have described more than 70 software resources with the CSD profile, and describing these software resources resulted in various improvements of earlier versions of the profile.These software resources mainly concern resources from the Netherlands. Most descriptions started from the information contained in the CLARIN-NL Portal, Services part.[13] The information there was semi-automatically converted to CMDI metadata in accordance with the CSD profile. The resulting descriptions were further extended and then submitted to the original developers and CLARIN Centres that host the resources for corrections and/or additions.

4 Metadata Quality

All metadata descriptions have been validated against the profile definition. We created several schematron[14] files for issuing errors or warnings for phenomena that are syntactically correct but incorrect or potentially incorrect in other ways. These schematron files check for the presence or absence of important (but optional) elements, for dependencies between elements or their values, e.g. an element to specify the language that the software can apply to must be present unless the value for the element *languageIndependent* is *yes*. We also made a schematron file to check for the presence of elements that are crucial for the faceted search described in section 5. A script has been provided for validation and for applying the schematron files. Additionally, a script was made to identify all URLs in the metadata descriptions and check for their resolution.[15]

The quality checks offered by the CLARIN Curation Module[16] (Ostojic et al., 2017) have also been used but are less useful because they can be applied only to a single metadata description at a time, check for the presence of metadata relevant for faceted search for data in the VLO, many of which are not so relevant for software, and because the profiles are cached so that modifications of the profiles are not immediately taken into account.

5 Faceted Search

A major purpose of metadata is to facilitate the discovery of resources. An important instrument for this purpose in CLARIN is the Virtual Language Observatory (VLO, (Van Uytvanck, 2014)). The VLO offers faceted search for resources through their metadata, but its faceted search is fully tuned to the discovery of *data*. For this reason, we defined a new faceted search, specifically tuned to discovery of *software*. This faceted search offers *search* facets and *display* facets:

Search Facets LifeCycleStatus, ResearchPhase, toolTask, researchDomain, linguisticsSubject, inputLanguage, applicationType, NationalProject, CLARINCentre, input modality, licence

Display Facets name, title, version, inputMimetype, outputMimetype, outputLanguage, Country, Description, ResourceProxy, AccessContact, ProjectContact, CreatorContact, Documentation, Publication, sourcecodeURI, Project, MDSelfLink, OriginalLocation

Of course, for a faceted search application to work on the metadata offered by the VLO, first of all a distinction must be made between the metadata that describe data and the metadata that describe software. Currently, no such distinction is made, but it can be largely added automatically on the basis of the CMDI profiles used and some existing facets (in particular *resource type*).

Furthermore, all metadata profiles for the description of software must be able to provide the values for the facets. That is the case to a large extent, though a little bit of metadata curation is needed in some cases and existing values must be mapped to a restricted vocabulary for use in the faceted search. This is the topic of the next section.

[13] http://portal.clarin.nl/clarin-resource-list-fs.

[14] http://schematron.com/

[15] On 2018-03-26, 750 URLs were correctly found, 22 were not found, and 73 exceptions were raised. Most exceptions raised are due to on-going changes on the INT website.

[16] https://clarin.oeaw.ac.at/curate/

6 Curation of existing metadata for software

The basic idea is as follows: we create a new standardised metadata record for all software descriptions, in principle each time a record is harvested. This metadata record contains the components and elements that are required for the faceted search as defined above. The record is constructed from the original CMDI record for the resource, combined with the data for this resource contained in a curation file, by a script. The curation file contains a sequence of conditions on each relevant element, and a specification of which values for which elements should be included in the new record if all the conditions are met. In general, the conditions simply test for identity with a value. The curation file can be used to add information that was lacking or only present in an unformalised way, and it can be used to map existing values to other values, e.g. to restrict them to a specific closed vocabulary. We report on our experiments with such a curation file for the *WebLichtWebService* profile, since it was most needed and most complex for this profile.

The *WebLichtWebService* profile lacks many elements that are necessary for faceted search, e.g. *toolTask*, *researchDomain*, *linguisticsSubject*, *inputLanguage*, *outputLanguage*, *Country*, *CLARINCentre*, *Documentation*, *Publication*, *modality* and *license*. We made a curation file for many of these properties, which can be used to add the relevant information in a new metadata record for a WeblichtWebService description: this is the case for the facets *toolTask*, *researchDomain*, *linguisticsSubject*, *inputLanguage*, *outputLanguage*, *Country*, and *modality*.

We still have to make curation files for the *ToolProfile*, *resourceInfo* and the *OLACDcmiTerms* profiles. We already inventoried the problems for the first two profiles, and curation files for these will be much simpler than the one for the *WebLichtWebService* profile.

7 Concluding Remarks

In the full paper we will summarise our conclusions. We will also describe some problems we encountered, which we only briefly mention here: (1) definition of closed vocabularies (2) no possibility to reuse metadata elements ; (3) a lot of variety in the contents of the CMDI envelope element *MdSelfLink*, resulting in several unresolved or syntactically incorrect results; (4) lack of good CMDI metadata editors. Finally, we will identify some future work, in particular on deriving CLRS registry entries for CLAM-based applications and web services.[17]

Acknowledgements

The work on metadata for tools described here started already in 2012 but has been interrupted several times. Many people have worked with me on the profile and the metadata descriptions, in particular Eline Westerhout and Rogier Kraf. Eric Renckens wrote many of the descriptions on the CLARIN-NL Portal pages that formed the basis for these metadata descriptions. Daan Broeder created the faceted search in the CLARIN in the Netherlands Portal. I am indebted to Menzo Windhouwer and Twan Goosen for their excellent support. The developers of the software and the CLARIN Centre managers hosting the software and their metadata provided and/or corrected the information contained in the metadata descriptions.

References

[Broeder et al.2012] Daan Broeder, Menzo Windhouwer, Dieter Van Uytvanck, Twan Goosen, and Thorsten Trippel. 2012. CMDI: A component metadata infrastructure. In *Proceedings of the LREC workshop 'Describing LRs with Metadata: Towards Flexibility and Interoperability in the Documentation of LR'.*, pages 1–4, Istanbul, Tyrkey. European Language Resources Association (ELRA).

[Ostojic et al.2017] Davor Ostojic, Go Sugimoto, and Matej uro. 2017. The curation module and statistical analysis on VLO metadata quality. In *Selected papers from the CLARIN Annual Conference 2016, Aix-en-Provence, 2628 October 2016*, number 136 in Linkping Electronic Conference Proceedings, pages 90–101. Linkping University Electronic Press, Linkpings Universitet.

[17] https://proycon.github.io/clam/.

[Schuurman et al.2016] Ineke Schuurman, Menzo Windhouwer, Oddrun Ohren, and Daniel Zeman. 2016. CLARIN Concept Registry: The New Semantic Registry. In Koenraad De Smedt, editor, *Selected Papers from the CLARIN Annual Conference 2015, October 1416, 2015, Wroclaw, Poland*, number 123 in Linköping Electronic Conference Proceedings, pages 62–70, Linköping, Sweden. CLARIN, Linköping University Electronic Press. `http://www.ep.liu.se/ecp/article.asp?issue=123&article=004`.

[van den Bosch et al.2007] A. van den Bosch, G.J. Busser, W. Daelemans, and S. Canisius. 2007. An efficient memory-based morphosyntactic tagger and parser for Dutch. In F. Van Eynde, P. Dirix, I. Schuurman, and V. Vandeghinste, editors, *Selected Papers of the 17th Computational Linguistics in the Netherlands Meeting*, pages 99–114. Leuven, Belgium.

[Van Uytvanck2014] Dieter Van Uytvanck. 2014. How can I find resources using CLARIN? Presentation held at the *Using CLARIN for Digital Research* tutorial workshop at the *2014 Digital Humanities Conference*, Lausanne, Switzerland. `https://www.clarin.eu/sites/default/files/CLARIN-dvu-dh2014_VLO.pdf`, July.

[Zinn2016] Claus Zinn. 2016. The CLARIN language resource switchboard. `https://www.clarin.eu/sites/default/files/08%20-%20ZINN-Lg-Sw-Board.pdf`. Presentation at the CLARIN 2016 Annual Conference.