



D2.11

Virtual Collection Registry v2

Document information

Title	Virtual Collection Registry v2
ID	CLARINPLUS-D2.11 (CE-2017-1067)
Author(s)	Willem Elbers (CLARIN ERIC)
Responsible WP leader	Dieter Van Uytvanck (CLARIN ERIC)
Contractual Delivery Date	2017-07-31
Actual Delivery Date	2017-07-30
Distribution	Public
Document status in workplan	Deliverable

Project information

Project name	CLARIN-PLUS
Project number	676529
Call	H2020-INFRADEV-1-2015-1
Duration	2015-09-01 – 2017-08-31
Website	www.clarin.eu
Contact address	contact-clarinplus@clarin.eu

Table of contents

Executive Summary	2
1 Introduction	3
2 VCR implementation work	3
2.1 VCR 1.1	3
2.2 VCR 1.2	4
2.3 Deployment	7
3 Integrations and related work	7
3.1 CLARIN Virtual Language Observatory	7
3.2 CLARIN Language Resource Switchboard	7
3.3 Related work outside CLARIN-PLUS	8
3.3.1 EUDAT	8
3.3.2 Research Data Collections WG	8
3.4 Related future work	9
3.4.1 ORCID integration	9
3.4.2 DataCite Metadata Store integration	9
4 Conclusions	9
References	11

All relevant source code and the specification documents related to the Federated Content Search can be accessed via the supplemental material page at:

<https://www.clarin.eu/content/clarin-plus-supplemental-material>

Executive Summary

In this deliverable we report on the development work for the Virtual Collection Registry (VCR) during the CLARIN-PLUS project. We discuss all of the implemented improvements by summarizing the work for the v1.1 and v1.2 releases. We then show the VCR in a broader context by presenting the various integrations available and how the VCR fits in the broader context of ongoing efforts in initiatives such as the Research Data Alliance (RDA). Finally we discuss possibilities of future work and conclude with a short overview of the goals set in the CLARIN-PLUS project and how these have been achieved.

1 Introduction

The virtual collection¹ registry (VCR) is a central, web-based, CLARIN service, where users can manage (create, edit, delete and publish) virtual collections. The current production instance² is hosted by Das Institut für Deutsche Sprache³ (IDS).

A virtual collection is a coherent set of links to digital objects (e.g. annotated text, video) that can be easily created, accessed and cited. The links can originate from different archives, hence the term virtual. This is shown in figure 1. A virtual collection is suitable for manual access (using a web-browser) as well as automated processing (e.g. by a web-service).

The VCR is closely integrated with the CLARIN infrastructure, it provides persistent identifiers for the published collections, as well as federated login, and the collection metadata is openly available and accessible via the Virtual Language Observatory⁴.

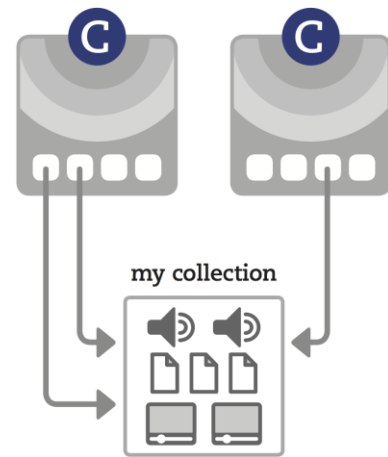


Figure 1: Virtual Collection

The service became available in 2010, but usage remained low. This was mainly due to the complexity of the user interface. Currently there are 5 publicly available collections in the Virtual Collection Registry. This task within the CLARIN-PLUS project aims to improve the VCR application in several ways, but making it easier to create and publish virtual collections is one of the main goals. It is expected that by increasing the overall user friendliness and robustness of the application the uptake will increase.

This deliverable describes the enhancements which have been implemented during the CLARIN-PLUS project. All source code is available on GitHub⁵.

2 VCR implementation work

The VCR improvements have been implemented in two releases. Release 1.1 to achieve milestone 2.9, Virtual Collection Registry v1 (software) [CLARINPLUS-M2.9] and release 1.2 to achieve the final deliverable D2.11 as defined in task 2.4.3 in the CLARIN-PLUS description of work. Both of these releases will be discussed in detail in this section. In addition to these implementation efforts, the deployment workflow has been brought in line with CLARIN best practices. This is discussed in this section as well.

2.1 VCR 1.1

For milestone 2.5 a number of improvements have been implemented: (1) getting the code base up--to--date with the recent version of Apache Wicket⁶, (2) improving the overall usability and (3) improving the administration functions.

¹ <https://www.clarin.eu/content/virtual-collections>

² <http://clarin.ids-mannheim.de/vcr/app/public>

³ <http://www1.ids-mannheim.de/>

⁴ <https://www.clarin.eu/vlo>

⁵ <https://github.com/clarin-eric/VirtualCollectionRegistry>

⁶ <https://wicket.apache.org/>

When development on the VCR was started in CLARIN-PLUS, the version of Wicket used was v1.4. The first undertaking was getting this dependency up-to-date by following a migration from v1.4 -> v1.5 -> v6⁷ -> v7. This was a big task since several backwards incompatibilities have been introduced between major Wicket versions. After these incompatibilities had been resolved, the codebase was using Wicket version 7. Because of the backward incompatibility issues in the Wicket codebase, several other dependencies which were no longer actively maintained, no longer worked and have been removed.

The next goal was to improve general usability of the user interface. Several smaller improvements have been made, such as setting default values and adding the logged in user as a creator making the creation of virtual collection a little easier. A citation button has been added to the actions associated with each virtual collection. Any published collection will show this citation button in the table row for that collection. The citation button is also shown in the details window of a virtual collection. This is shown in figures 2.1 and 2.2.



Figure 2.1: Citation in collection details window, as of release 1.2

Figure 2.2: Citation in collection table, as of release 1.2

Another addition in this release was an administration page, where users designated as administrator can perform update and delete actions on any virtual collection. This was a much needed feature allowing to clean up incorrect collections and/or maintain collections where the original creator(s) are no longer available. The administrator can select a collection space, which is either the space containing all public collections or the private space for a specific user. The private space of a user contains all collections that user has created but not published. After selecting a collection space, the administrator can use the default view to edit any collection in that specific collection space.

2.2 VCR 1.2

The three main goals for this release were: (1) apply the CLARIN base style, (2) further simplify the workflow to create a virtual collection and (3) integrate with the Language Resource Switchboard⁸ (LRS) [[CLARINPLUS-D2.5](#)].

The first task for this VCR release was to update Wicket to version 7.6. This was a minor update and no big issues were encountered. This paved the way for the next step.

In order to get the VCR in line with the uniform styling for CLARIN core applications, wicket-bootstrap⁹ has been included as a dependency and core user interface features have been rewritten to be based of the components provided by the wicket-bootstrap library. With the basic wicket-bootstrap elements in place, the next step was to apply

⁷ From Wicket v6 the versioning scheme was changed, hence the big jump from v1.5 to v6.

⁸ <http://weblicht.sfs.uni-tuebingen.de/clrs/#/>

⁹ <https://github.com/lorndn1kk0n/wicket-bootstrap>

the CLARIN base style. The CLARIN base style¹⁰ is essentially a customisation of the default Bootstrap theme and builds on the CLARIN human interface guidelines and style definitions developed in the related task. More information can be found in [CLARINPLUS-D3.1].

As part of this effort, the top menu bar has also been redesigned to fit the goals of responsive designs as shown in figure 3. A short explanation of the examples follows below:

- The top image is an example of the menu when there is no logged in user. Users can easily browse the public collections or create a new collection, which will trigger a login first for non authenticated users.
- The middle image is an example of the menu for authenticated users. Creating a collection will not trigger a login and users can click their username to access the list of their own public and private collections.
- The last image is an example of the menu bar on narrow screen, such as on a mobile device. The actions are grouped in a menu button, which when clicked will expand and show all options otherwise visible in the menu bar.

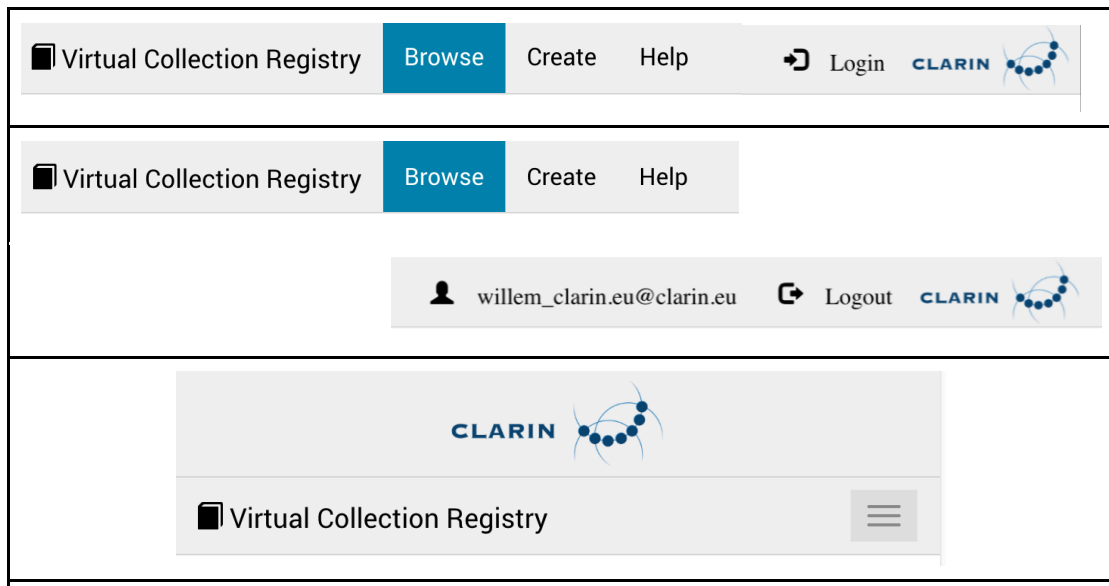


Figure 3: Redesigned responsive top menu bar as of release 1.2

Easier creation of virtual collections was one of the main goals for the VCR improvements within CLARIN-PLUS. Together with the default values set in the 1.1 release this has been achieved by the changes included in the 1.2 release. The original workflow was based on a wizard with 3 steps to provide the general information, information on creators and add the links to external sources. The new, simplified, workflow has been designed around a single page to supply all information, as shown in figure 4. For text input this is straightforward, the lists with controlled vocabularies have been replaced by button groups and the keywords, authors and resources input fields have been replaced by a more complex component that allows editing in place without the need for popups or other pages.

¹⁰ Available at https://github.com/clarin-eric/base_style

Figure 4: Simplified creation of a virtual collection as of release 1.2

The component to edit values for keywords, authors and resources in place allows users to fill in values and add them as a new row to the list of values. For each row of values, there is a trashcan button to remove this item. An example is shown in figure 5.

Author(s)	Person	Email	Organisation	
	Robbert van Sluijs	r.vansluijs@let.ru.nl	Radboud University	
	Margot van den Berg	M.C.vandenBerg1@uu.nl	Utrecht University	
	Pieter Muysken	p.muysken@let.ru.nl	Radboud University	
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Add"/>

Figure 5: Inline editing of authors as of release 1.2

For this release the Language Resource Switchboard has also been integrated as shown in figure 6. The LRS is described in more detail in section 8.1.






Resources		
Type	Reference	Action
Resource	Corpus Online Corpus supplement	
Resource	Link to NEHOL Database: Böhner1 (3.2.1.)	
Resource	Link to NEHOL Database: Böhner 2 (3.2.2.)	
Resource	Link to NEHOL Database: Geskiedenis (3.2.3.1.)	
Resource	Link to NEHOL Database: Geskiedenis, unknown (3.2.3.2.A)	

Figure 6: Language Resource Switchboard integration as of release 1.2

2.3 Deployment

To bring the VCR deployment workflow in line with CLARIN ERIC operational practices a dockerized¹¹ version of the application has been created. This docker image is utilizing the CLARIN tomcat 8¹² base image¹³. A docker-compose file is provided in the VCR GitHub repository which will both start a MariaDB database and a VCR application running in tomcat 8. This docker-compose file can be used for development and as an example for a production deployment.

To bring the service further in line with best practices for central services the VCR will be made available on the following domains:

- vcr.clarin.eu, running the production version of the application.
- beta-vcr.clarin.eu, running the latest development version of the application, which after testing will be deployed in production.

See the supplemental material page for more details.

3 Integrations and related work

3.1 CLARIN Virtual Language Observatory

The VCR implements an OAI-PMH provider and is configured as one of the providers for the Virtual Collection Registry (VLO) [CLARINPLUS-D2.10]. Any published virtual collection is automatically made available and discoverable in the VLO using this mechanism.

3.2 CLARIN Language Resource Switchboard

As written in D2.10 Virtual Language Observatory v2 [CLARINPLUS-D2.10]: “The development of the Language Resource Switchboard (LRS) was included as a task in the CLARIN-PLUS proposal to “bridge the gap between resources and processing workflows”. The LRS functions as a stand-alone application, but its full potential becomes apparent in the context of a ‘pipeline’ scenario in which users can create or discover resources and immediately get an *actionable* overview of processing options.”

The integration as implemented in this version of the VCR, allows users to open a virtual collection in the Language Resource Switchboard with a single click. This was defined both as a goal in the CLARIN-PLUS proposal and as a goal for the LRS [CLARINPLUS-D4.2].

¹¹ <https://www.docker.com/>

¹² <http://tomcat.apache.org/>

¹³ <https://gitlab.com/CLARIN-ERIC/docker-tomcat8>

3.3 Related work outside CLARIN-PLUS

3.3.1 EUDAT

There are multiple possibilities to integrate the VCR and EUDAT services. In this section we present two. An example of the VCR using an EUDAT service and an example of an EUDAT service potentially using the VCR.

Creators of virtual collections are making use of resources deposited in the EUDAT B2SHARE¹⁴ service. The collection defined in the VCR points to resources deposited in B2SHARE. An example of such a collection is the “Exploring genealogical blends: the Surinamese Creole Cluster and the Virgin Island Dutch Creole Cluster¹⁵” collection.

Another use case for integrating the VCR with EUDAT services would be for B2FIND¹⁶ to harvest the metadata on virtual collections directly from the VCR OAI provider. The VCR collections are not yet included in the production B2FIND service, however we are in contact with the B2FIND administrators and have already included the VCR records in the B2FIND staging instance.

3.3.2 Research Data Collections WG

The Research Data Alliance (RDA)¹⁷ Research Data Collections Working Group¹⁸ is working on a “unified cross-community approach to building and managing collections and no common model for understanding them”. On a high level the VCR is compatible with the definition of a collection as defined by working group. From the Research Collections Working Group - Specification - draft document:

“Imagine you have a number of objects that belong together. The type of object is a bit flexible, as long as it is in some digital form; this can include digital documents or scientific articles, individual data files, a zip of several files, digital images, audio or video recordings. The reason why these objects belong together is also not extremely rigid. There may for example be a number of files that came out of a scientific model calculation, or a number of recordings from a study session or a very distinctively selected choice of files grouped together for a particular analysis.

In conclusion, a collection is a very flexible mechanism to bind objects together. While the contents of a collection may therefore undergo changes, its structure is more rigid. A collection has a distinct identity even throughout such changes and that it offers a set of precisely defined actions used to modify it.”.

The working group is currently working on a specification of virtual collections and we have to make sure that we follow the specification wherever possible in the VCR. In addition to following the specification the working group is also working on an API to manage virtual collections¹⁹. We will investigate if and how we can make use of this API.

¹⁴ <https://b2share.eudat.eu/>

¹⁵ <http://hdl.handle.net/11372/VC-1003>

¹⁶ <http://b2find.eudat.eu/>

¹⁷ <https://www.rd-alliance.org>

¹⁸ <https://www.rd-alliance.org/groups/research-data-collections-wg.html>

¹⁹ <http://rdacollectionswg.github.io/apidocs/#>

3.4 Related future work

3.4.1 ORCID integration

Integration with ORCID has been investigated. Our aim for integrating with ORCID is to be able to associate ORCID identifiers to the creators of virtual collections. We have to be able to perform a lookup into the ORCID database and search for the ORCID id based on the email address and/or name we have for the creator. ORCID offers a number of APIs. For this use case we can use the search functionality of the public API.

The VCR has to be registered as an OAuth client as described in the “register a public API client application” article²⁰. After registering the client, the public search api can be used to issue any valid SOLR search are described in the “Basic Tutorial: Searching Data Using the ORCID API” article²¹.

An example search would look like the following:

```
Method: GET
Content-type: application/vnd.orcid+json
Authorization type: Bearer
Access token: [Stored access token]
URL: https://pub.sandbox.orcid.org/v2.0/search/?q="willem+elbers"
```

To integrate this functionality in the UI, the create/edit virtual collection screen will be adjusted. After adding or updating a creator a callout will be issue to the ORCID API in the background. After receiving the list of possible researchers, the UI is updated in-line and the user has to confirm the correct research of multiple results are available.

3.4.2 DataCite Metadata Store integration

Currently the VCR is minting PIDs under the CLARIN prefix when a virtual collection is published. This process is controlled by running the *perform* method of a *VirtualCollectionRegistryMaintenance* instance periodically. After a virtual collection has been published, it will not be publicly available until it has been processed by the periodic maintenance. Processing such a collection includes minting the PID. Minting of DOIs should be added to this step.

Before we can use the DataCite API we have to register as a client with a DataCite member. After registering with a DOI member, the Metadata Store API²² can be used to mint new PIDs. This comprises three steps: (1) generate a unique DOI name, (2) create the metadata record and (3) register the DOI. This is all implemented as a REST API, similar to the current EPIC API implementation. The minted DOIs should be shown next to the existing PIDs in the detail view of a virtual collection and in the citation popup, giving the user the option to use either the PID, the DOI or both as a reference.

4 Conclusions

Within the CLARIN-PLUS project two releases for the VCR have been made, resulting in a codebase with up-to-date dependencies, an application in line with CLARIN best

²⁰ <https://support.orcid.org/knowledgebase/articles/343182>

²¹ <https://members.orcid.org/api/tutorial/search-orcid-registry>

²² <https://support.datacite.org/docs/mds-2>

practices and a user interface improved on various levels all aimed at improving the uptake of the VCR application. As mentioned in the task definition an administration interface has been implemented to allow the correction of errors in existing virtual collections. The deploy workflow has been brought in line with CLARIN best practices.

Integration of ORCID ids and integration into the DataCite metadata store by minting DOIs has been investigated and a short description on how this could be implemented and added to the VCR has been included in section 8.4, related future work.

References

[CLARINPLUS-M2.9] Elbers, W. 2016. *CLARIN Virtual Collection Registry (VCR) Milestone 1*. https://office.clarin.eu/v/CE-2016-0819-VCR_Milestone_1.pdf

[CLARINPLUS-D2.5] Zinn, C. 2016. *LR Switchboard (software)*. https://office.clarin.eu/v/CE-2016-0881-CLARINPLUS-D2_5.pdf

[CLARINPLUS-D3.1] Kamran, A. and Straňák, P. 2016. *CLARIN Human Interface Guidelines*. https://office.clarin.eu/v/CE-2016-0794-CLARINPLUS-D3_1.pdf

[CLARINPLUS-D2.10] Goosen, T. Eckart, T and Windhouwer, M. 2017. *D2.10 Virtual Language Observatory v2*, https://office.clarin.eu/v/CE-2017-1057-CLARINPLUS-D2_10.pdf

[CLARINPLUS-D4.2] Zinn, C., Goosen, T., Hinrichs, M., Dima, E., Elbers, W., Van Uytvanck, D., Goldhahn, D., Trippel, T. and Mišutka, J. 2017. *Joint Infrastructure Services*. https://office.clarin.eu/v/CE-2017-0985-CLARINPLUS-D4_2.pdf

All relevant source code and the specification documents related to the Federated Content Search can be accessed via the supplemental material page at:

<https://www.clarin.eu/content/clarin-plus-supplemental-material>