

OpenScout deliverable D2.2.1 Initial Competence Service Library

Citation for published version (APA):

Kalz, M., Börner, D., & Overby, E. (2011). *OpenScout deliverable D2.2.1 Initial Competence Service Library*.

Document status and date:

Published: 28/12/2011

Document Version:

Peer reviewed version

Document license:

CC BY-NC-SA

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

<https://www.ou.nl/taverne-agreement>

Take down policy

If you believe that this document breaches copyright please contact us at:

pure-support@ou.nl

providing details and we will investigate your claim.

Downloaded from <https://research.ou.nl/> on date: 12 Feb. 2025

Open Universiteit
www.ou.nl



ECP 2008 EDU 428016**OpenScout****D2.2.1 Initial Competence Service Library**

Deliverable number	<i>D2.2.1</i>
Dissemination level	<i>Public</i>
Delivery date	<i>28 February 2011</i>
Status	<i>Final</i>
Author(s)	<i>Marco Kalz (OUNL)</i>
Contributor(s)	<i>Dirk Börner (OUNL), Erlend Overby (HYP)</i>

eContentplus

This project is funded under the *eContentplus* programme¹,
a multiannual Community programme to make digital content in Europe more accessible, usable and exploitable.

¹ OJ L 79, 24.3.2005, p. 1.

Table of Contents

1	INTRODUCTION	3
2	COMPETENCE METADATA SERVICE	4
2.1	TOP DOWN COMPETENCE ENRICHMENT SERVICE	4
2.2	BOTTOM-UP TAGGING SERVICE	6
3	ACCESSIBILITY SERVICE	9
3.1	ACCESSIBILITY FUNCTIONALITY OF THE PORTAL	9
3.2	ACCESSIBILITY AWARENESS	10
3.3	ACCESSIBILITY VALIDATION	11
3.3.1	<i>OpenScout Accessibility validator service</i>	<i>11</i>
3.3.2	<i>WEB - validating</i>	<i>12</i>
3.3.3	<i>PDF - validation</i>	<i>12</i>
3.3.4	<i>Web-course</i>	<i>12</i>
3.3.5	<i>Validator results</i>	<i>13</i>
3.3.6	<i>OpenScout accessibility validator API</i>	<i>13</i>
3.3.7	<i>Search results</i>	<i>13</i>
4	DISCUSSION.....	14
	REFERENCES	15

1 Introduction

This document describes the implementation of the competence-service library version 1 of the OpenScout project. The current status is based on the model presented in the deliverable D2.1 “Analysis Report on Competence Services” and the integration plan discussed there. The OpenScout project supports competence-based search and browsing of open educational resources in the domain of business and management education. The competence service library evolves during the project from a resource-oriented service library (version 1) towards a service library that supports individual competence development of OpenScout users (version 2).

In the first part of the deliverable the competence metadata services are described. The service consists of a competence metadata service and a bottom-up tagging service that supports three tag types. The second chapter of the deliverable describes the eAccessibility-validator that assesses the suitability of resources from the OpenScout federation for users with special needs.

2 Competence Metadata Service

2.1 Top down competence enrichment service

The core of the Competence Services is the provided competence catalogue. The catalogue contains competence classifications (competence categories) and their related competencies as well as resources, experts, evidences and proficiency scales related to these competences.

The competence catalogue itself is an object-oriented application written in Java and is able to output the contained data in several formats such as XML and JSON. The application cannot be used stand-alone. It can only be used via the designated web services (competence web service and competence administration service) described later in this section. The web services are deployed on a Glassfish v2 application server and can be accessed using its SOAP API.

The web services enable front-end applications, especially the portal, competence widgets, and the competence administration application to query and manipulate data contained in the competence catalogue.

A.) Competence Web Service

WSDL:

http://openscout.ou.nl:8080/OpenScout_Competence_Webservices/CompetenceServicesService?wsdl

Tester:

http://openscout.ou.nl:8080/OpenScout_Competence_Webservices/CompetenceServicesService?tester

The competence web service is an abstraction layer to the competence catalogue to provide access to it. The service accesses the contained data via a web service, which provides the following methods:

B.) Competence Web Service Description

METHOD	PARAMETER	DESCRIPTION
getVersion	none	Get the current version of this webservice
getStatus	none	Get a list of all the tag types.
getDomains	String userID, String outputFormat	Get a list of all the available domains
getCompetenceList	String userID, String outputFormat	Get the full list of competences from the database
getCompetenceListFromDomain	String userID, Integer domainID, String outputFormat	Get a basic list of competences from the database for a specific domain
getBasicCompetence	String userID, Integer competenceID, String outputFormat	Get a basic competence from the system
getBasicCompetenceList	String userID, String outputFormat	Get a basic list of competences from the database
queryCompetences	String userID, String query, String outputFormat	Query competences in the database
getCompetence	String userID, Integer competenceID, String outputFormat	Get one competence from the database and all the related experts, evidences and resources
getResourcesFromCompetenceId	String userID, Integer competenceID, String outputFormat	Get all the resources for a certain competence
getProficiencyScalesForCompetence	String userID, Integer competenceID, String outputFormat	Get all the evidences for a certain competence
getExpertsFromCompetenceId	String userID, Integer competenceID, String outputFormat	Get all the experts for a certain competence
getProficiencyScale	String userID, Integer proficiencyScaleID, String outputFormat	Get a proficiency scale from the database

Method Parameter:

String userID	Can be any string (just for logging purposes), e.g. “portal”
String outputFormat	Specifies the output format (XML or JSON), e.g. “xml”
domainID	Unique Domain aka competence classification identifier, e.g. 110
competenceID	Unique competence identifier, e.g. 239
query	Can be any string, e.g. “SWOT”
proficiencyScaleID	Unique proficiency scale identifier

This service has been extended with bottom-up tagging functionality described next.

2.2 Bottom-up tagging service

To support the construction of bottom-up folksonomies besides the top-down competence taxonomies the competence metadata service has been extended with a bottom-up tagging functionality (Lohmann, Thalmann, Harrer, & Maier, 2008). Since there will be already a tagging service available that allows users to apply free tags to learning resources for domain related tagging the approach followed in WP2 is targeting to collect additional competence-related information from users via the tagging functionality. Therefore we do not apply a completely free tagging but a pre-structured tagging approach (Golder & Huberman, 2006) where users are asked to tag a resource according to the following three competence-related aspects:

- Purpose of a resource (see Strohmaier, 2008)
- Context of a resource (see Vuorikari, 2009)
- Problem solved by the resource

These three tag-types will be supported by the bottom-up tagging web-service. The tagging service can be used by all logged-in users of the platform. To differentiate these tags from the domain-related tags the user will be able to chose, which tag-clouds are displayed for the learning resources.

A.) Competence tagging service

WSDL:

http://openscout.ou.nl:8080/OpenScout_Compentence_Webservices/TaggingServicesService?WSDL

Tester:

http://openscout.ou.nl:8080/OpenScout_Compentence_Webservices/TaggingServicesService?tester

B.) Competence Tagging Webservice Description

METHOD	PARAMETER	DESCRIPTION
getVersion	none	Get the current version of this webservice
getTagTypes	id, name, description, outputFormat	Get a list of all the tag types. Parameters are used to filter. getTagTypes(0,"","","") gives all tag types.
getTags	id, name, userId, objectId, type, outputFormat	Parameters are used to filter. getTags(0,"","","0,") gives all tags for all userids, objectids and tag types.
updateTages	userId, objectId, type, tags	Update tags for given userid, objectId and tag type. These three parameters cannot be empty (or 0 for type).
getUserIds	id, name, userId, objectId, type, outputFormat	Get a filtered list of user ids, Parameters are used to filter. getUserids(0,"","","0,") gives all userids.
getObjectIds	id, name, userId, objectId,	Get a filtered list of object ids, Parameters are used to filter. getObjectids(0,"","","0,") gives all objectids.

userId	the Id of the user (string), if empty
--------	---------------------------------------

	tags for all user ids
name	the name of the tag type (string), if empty tag types with all names
id	the id of the tag type (number), if 0 tag types with all ids
objectId	the Id of the object (string), if empty tags for all object ids
type	the type of the tag (number), if 0 tags for all types
outputFormat	specifies which output format is used (xml or json)

Accessibility services are another important component of the competence service library that supports users with special requirements to use learning resources from the OpenScout federated infrastructure. This component is described in the next chapter of the deliverable.

3 Accessibility Service

3.1 Accessibility functionality of the portal

Within education and life long learning, accessibility is of great importance. As society is changing so are the requirements for the work force to learn new skills and competences. Since we cannot predict anything about the capabilities of the learners accessing, using and learning from and interacting with the learning resources made available in OpenScout we need to have a constant focus on accessibility.

In December 2009 we did a survey among the content providers of the OpenScout consortium, and we found that there was no focus on accessibility. There was little understanding of accessibility as an issue, and little or no knowledge about different accessibility requirements and accessibility features. As a consequence of this we have tried to raise the awareness of accessibility among the content providers. However it is hard to demonstrate a demand for accessible resources, since potential learners with accessibility requirements most likely have problems in accessing the portals that provide information about the learning resources. This leads to a hen-and-egg like problem: On the one hand the content providers are not aware of the low accessibility of the resources, on the other hand no potential users can use these resources and notify them.

As a consequence of this we are currently re-implementing the OpenScout portal to be as accessible as possible by providing a dedicated accessible alternative version to the search interface of the OpenScout portal. There is usually a dilemma in making one portal accessible and at the same time rich on interaction and use of highly advanced features, many tried to compromise on this and develop a portal that does not really work for anyone, not the one requiring rich interactions, and the ones requiring accessibility. Since the OpenScout portal is developed with a separation between the logics of the portal, and how the information is presented, it is rather straightforward to develop an accessible version of the search and retrieve functionality of the portal. The highly accessible HTML version of the OpenScout search functionality is currently in development, to ensure that everyone will have access to the portal, and that everyone will have the possibility to search for and identify relevant learning resources.

3.2 Accessibility awareness

The next challenge is how to make the content providers more aware of the market for accessible learning resources. We will do this by adding accessibility information to the learning resources in the Open Scout federated infrastructure.

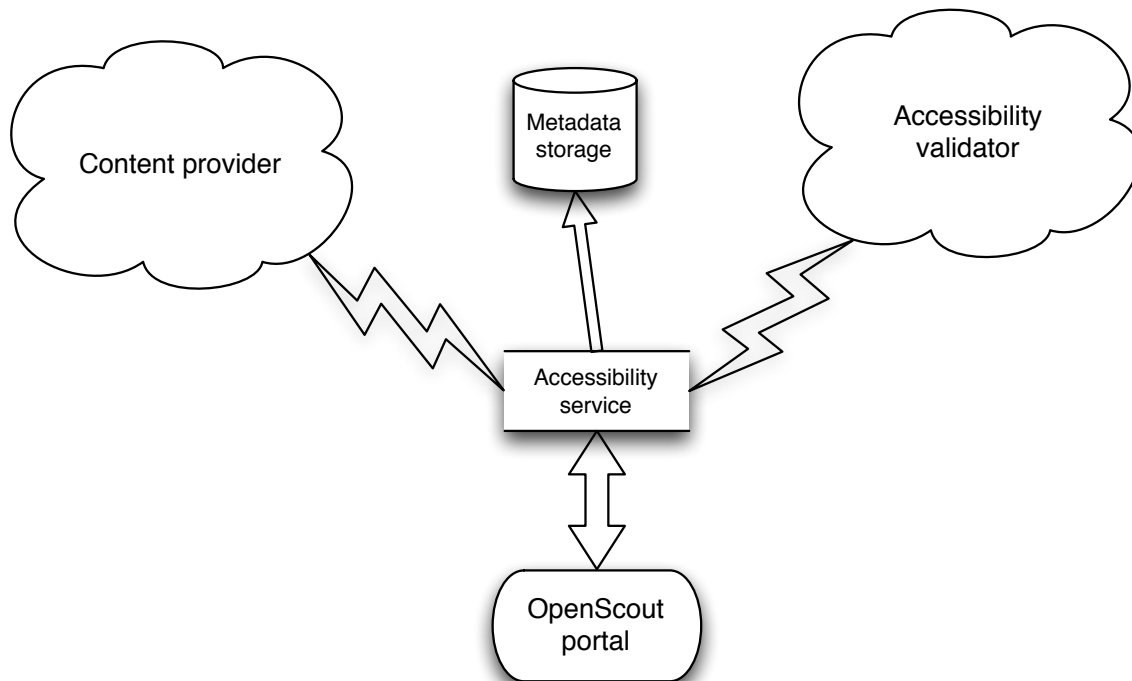


Fig. 1 Accessibility service

By providing such a service we hope to achieve several aims.

- When all resources have information about their accessibility users who need accessible resources will have a tool easily available for identifying the level of accessibility of a resource.
- When accessibility tests are performed, and a report is sent to the content provider, we will notify the content provider about accessibility issues that need to be fixed
- When content providers are receiving a report on accessibility issues, they are also notified that there is a demand for the resources to be accessible.

How this service would work is illustrated in Fig. 1 Accessibility service.

3.3 Accessibility validation

As an external resource for validating and checking the accessibility of a resource we use the services provided by the European egovmon project. In this project the accessibility of governmental web-sites in Europe is monitored and rated. This will ensure that the OpenScout learning resources are checked against the same requirements as other European web-sites.

At the first phase we are developing a servlet for communicating with the accessibility validator, and for parsing, and presenting the results from the accessibility validator. At the next phase we will update the metadata of the learning resource with the level of accessibility, and start sending notifications to the content providers. Finally the search functionality will be enhanced to filter search based on accessibility requirements.

In the next part the Accessibility validator service is described in detail.

3.3.1 OpenScout Accessibility validator service

The OpenScout accessibility validator service will raise the awareness of the users of how accessible a resource is, and to help the users of the OpenScout portal find resources that are accessible.

We did a survey of accessibility features implemented in the available learning resources by the content providers when the OpenScout project started in 2009. In this survey we found that none of the content providers had any policy on the accessibility of their resources. Many resources are in in-accessible formats such as Flash and Silverlight; others are in PDF and HTML. There was no focus on making accessible PDF, nor was there any focus on making the HTML accessible. The survey demonstrated little focus on accessibility and little knowledge on how to make resources accessible.

To address this problem we will develop an accessibility validation service for end users, and to inform the content providers of accessibility issues of the resources that are provided. In the OpenScout project we will not develop our own accessibility checker but facilitate already existing validation services.

The accessibility validator service we will use in OpenScout are the same used for monitoring the accessibility of all governmental web-sites in Europe that was developed in the European Internet Accessibility Observatory project, and that are maintained in the EGOVMON project. We chose this validator since it is developed in another European project and we envisage that the validator will be updated and follow requirements applicable in Europe.

Currently the EGOVMON accessibility validator checks for several accessibility features such as most of the WCAG1.0² recommendations, and there are plans for checking the WCAG2.0³ recommendations. The validator also checks CSS styles sheets, and accessibility of PDF documents. Based on the development and plans for the validator checker we find that

² WCAG1 - <http://www.w3.org/TR/WCAG10/>

³ WCAG2 - <http://www.w3.org/TR/WCAG20/>

this validator will work well with the OpenScout accessibility service. However if we need to use additional checkers this would easily be implemented in the OpenScout accessibility validator service by providing several target validators. The validator can check several resources types which are available in the OpenScout content federation.

3.3.2 WEB - validating

The Accessibility validator follows the Unified Web Evaluation Methodology 1.2⁴ and checks web sites against WCAG1.0⁵ recommendations, such as:

- That all images have an ALT attribute, explaining the semantics of the image.
- That all form elements have a LABEL attribute, explaining what the form element is for.
- That the document has a correct DOCTYPE declaration, ensuring that the page is interpreted and validated corrected.
- That all heading levels are in sequence i.e. H1, H2, H3 is the sequence of headings, and that no heading level is skipped e.g. H1, H3. Skipped heading levels will cause problems for assistive technologies presenting the content of the web page to a user with special needs.

In addition many other accessibility features are checked.

3.3.3 PDF - validation

The Accessibility validator also checks accessibility features of a PDF document, such as:

- That all images/figures have an alternative text, explaining the semantics of the image/figure
- That the PDF document have a document title
- That natural language is specified - this is used by screen readers when choosing how to read out the document.
- That the document is correctly structured. I.e. that all headings and titles in the document have correct structural tags.

In addition several other PDF accessibility features are checked.

3.3.4 Web-course

Many web-based learning resources are a collection of web sites, and in the OpenScout repository we have the address to the start page of the document, usually a accessibility checker then will only validate the start page. However the OpenScout accessibility validator service will perform a validation of all subsequent web pages (the site) of that web-course. Performing a check of the site will guarantee a more thorough accessibility validation of the web-course, and we will avoid checking of only the start page. Since the complexity and number of web-pages related to a web-course is not known, validating all these pages could

⁴ UWEM - http://www.wabcluster.org/uwem1_2/

⁵ WCAG2 - <http://www.w3.org/TR/WCAG20/>

take some time, and measures will be established to avoid that the user have to wait for the result of the evaluation.

3.3.5 Validator results

When the validation is completed, the validator checker provides a UVEM number that indicates the overall accessibility of the web site tested. This number is indicating the accessibility of the web page or the web site. We will use this number to inform the user of the level of accessibility of the learning resource.

The validator results will also be added to the existing metadata so that the metadata is enriched. The validator service will also send a more complete report of accessibility problems to the content provider.

3.3.6 OpenScout accessibility validator API

The accessibility validator service will accept the following input variables:

- URL to be checked
- Type of document (Page, Site, PDF)

The output from the service will be the following:

- String, which will be formatted according to the JSON specification. The value will indicate a URL for a more detailed report, and the accessibility value.

A call to the service will look like the following:

- `http://openscout.net/services/accessibility-checker?url=http://openscout.net/resource/01&type=Page`

The results would look like the following:

```
{ report-url:  
"http://report.egovmon.org/reports/12837aihlds1  
92783", report-value: "0.25" }
```

The accessibility-checker servlet will first check if the URL have already been checked, and if so retrieve the value from the enriched metadata from the OpenScout metadata store. If this URL has not been checked before, the proper accessibility checker will be invoked. If the checking of the URL takes to long, the user will be notified that this will take some time and be encouraged to continue exploring the OpenScout portal.

When the URL is checked the result will be presented to the user, and a report will be sent to the content provider, and the metadata will be enriched with the result of the accessibility check.

3.3.7 Search results

When doing a search for resources within OpenScout the search results with the OpenScout accessibility services enabled could look like the following:

Title	Content	Description	Accessibility
Studying the arts and humanities	Text/html	This unit is an introduction...	Partly Accessible
History as commemoration	Text/html	Commemoration – remembering...	Check Accessibility
David Hume	Text/html	This unit examines Hume's...	Not Accessible
French Revolution	Text/html	This unit provides basic...	Check Accessibility

In the next version of the OpenScout system, the OpenScout accessibility service will be invoked on all new resources that are identified and collected in the repository, and the metadata will be enriched with the accessibility results when the resource is made available in the OpenScout portal.

4 Discussion

As mentioned in the introduction the current version of the competence service library is enabling basic functionality to enrich learning resources from the OpenScout federation with competence metadata and accessibility information. This first version covers the following elements described in the DoW: skill-based search of learning resources, taxonomy/folksonomy building, metadata generation & accessibility metadata. In the next phase the focus of our work will shift from services to enrich resources to services that support end users of these resources during their learning process. This advanced competence service library adds the following components: competence based rankings, accessibility tools & services.

References

- Golder, S. A., & Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science*, 32(2), 198-208.
- Lohmann, S.; Thalmann, S.; Harrer, A.; Maier, R. (2008): Learner-Generated Annotation of Learning Resources – Lessons from Experiments on Tagging. In: Tochtermann, K.; Maurer, H. (Hrsg.): Proceedings of I-KNOW '08. 8th International Conference on Knowledge Management. Eigenverlag, S. 304 - 312.
- Strohmaier, M. "Purpose tagging: capturing user intent to assist goal-oriented social search." In Proceeding of the 2008 ACM workshop on Search in social media, 35–42. ACM, 2008.
- Vuorikari, R. (2009). Tags and self-organisation: a metadata ecology for learning resources in a multilingual context. Doctoral thesis. November, 13, 2009, Heerlen, The Netherlands: Open University of the Netherlands, CELSTEC.