Elearning, Communication and Open-data: Massive Mobile, Ubiquitous and Open Learning

Deliverable Title	D3.6 Report on implementation of the ECO Federated Search infrastructure
Deliverable Lead:	OUNL
Related Work package:	WP3
Author(s):	Stefaan Ternier, Kjeld Loozen, Javier Viñuales, Sara Tejera, Alessandra Tomasini, Sören Unruh
Dissemination level:	Public (PU)
Due submission date:	31 January 2016
Actual submission:	31 January 2016

Abstract	This document describes the implementation of the ECO federated search architecture. It presents an analysis of various architectures and defines based on this analysis an architectural solution for federated search in ECO.
Keywords	Federated search architectures, metadata harvesting, OAI-PMH, distributed architectures

Project funded by ICT Policy Support Programme as part of the Competitiveness and Innovation framework Programme (CIP)

Disclaimer

This document has been produced in the context of the ECO Project, which has received funding from the European Community's CIP Programme under grant agreement n° 621127.

This document contains material, which is the copyright of certain ECO consortium parties, and may not be reproduced or copied without permission.

In case of ***Public (PU)****:*
All ECO consortium parties have agreed to full publication of this document.

In case of ***Restricted to Programme (PP)****:*
All ECO consortium parties have agreed to make this document available on request to other framework programme participants.

In case of ***Restricted to Group (RE)****:*
All ECO consortium parties have agreed to full publication of this document. However this document is written for being used by <organisation / other project / company etc.> as <a contribution to standardisation / material for consideration in product development etc.>.

In case of ***Consortium confidential (CO)****:*
The information contained in this document is the proprietary confidential information of the ECO consortium and may not be disclosed except in accordance with the consortium agreement.

The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the ECO consortium as a whole, nor a certain party of the ECO consortium, warrant that the information contained in this document is capable of use, or that use of the information is free from risk, and accept no liability for loss or damage suffered by any person using this information.

The user thereof uses the information at its sole risk and liability. For the avoidance of all doubts, the European Commission has no liability in respect of this document, which is merely representing the authors view.

Versioning and Contribution History

Version	Date	Main modification or revision	Contributions by
v0.1	02-09-2015	First version proposed by OUNL	OUNL
v0.2	08-12-2015	Meeting between Stefaan Ternier and Kjeld Loozen on refining architecture and implementation details	OUNL, REIMERIT
v0.3	18-12-2015	Minor comments by Javier Viñuales + resolution by Stefaan Ternier	GEO
	27-01-2016	Comments by Sören Unruh	HUMANCE
	29-01-2016	Comments by Antonio Alfaro	Antonio Alfaro
v0.4	30-01-2016	Resolution of comments by Stefaan Ternier	OUNL
v1.0	30-01-2016	Final review by Javier Viñuales	GEO

Main contributors to the document:

Partner	Contributors
UNED	Sara Tejera
REIMERIT	Kjeld Loozen
HUM	Sören Unruh
OUNL	Stefaan Ternier
POLIMI	Alessandra Tomasini
GEO	Javier Viñuales

Reviewers to the document:

Partner
HUM, Antonio Alfaro

Table of contents

Executive summary

ECO features an ecology of MOOC providers. This deliverable illustrates how through the use of the OAI-PMH protocol for metadata harvesting, MOOC metadata is harvested from the providers into the ECO Backend. In this way, the EcoPortal already offers federated 'MOOC' search over the different MOOCs in the ECO network. In addition, this deliverable covers the process of disseminating ECO courses to external federated networks of repositories by creating one OAI-PMH access point for external harvesters.

Historically, two architectures for federated search have been implemented in the Technology Enhanced Learning (TEL) community: federated queries and federated metadata. First an analysis is made for federated query architectures. The deliverable presents how in the 'federated queries' case, queries are federated to repositories (MOOC providers) as the queries are formulated by users. This analysis presents some important drawbacks for this solution.

Next, federated metadata (OAI-PMH) is presented. Here metadata is federated to a search index, prior to executing a query. When a user formulates a federated query, the query is executed on the search index. This approach has important benefits for scalability but has the drawback that queries are not executed on "live" data. When course data is altered at a MOOC provider it takes some time (i.e. the time between two harvesting sessions) before the search index is updated.

In the next chapter, this deliverable presents how third party federations (such as globe) will be able to query ECO content via the OAI-PMH standard. By offering these federations access to our MOOCs, we hope to attract more users.

In the final chapter the implementation and planning of this architecture on the ECO context is discussed. An OAI-PMH mechanism is specified to label the different MOOC providers in ECO, so that third party federations will be able to selectively harvest e.g a single MOOC provider. Provenance is discussed so that records that are federated outside of ECO can be linked back to their origin. This is important when implementing multiple layers of OAI-PMH and enables identifying the original MOOC provider.

1. Introduction

This deliverable contributes the ECO framework for federated search. The goal of this infrastructure is twofold. On the one hand side it gives third party federations of learning resources access to the MOOCs offered in ECO. On the other hand, it defines a set of rules to create a MOOC platform where MOOCs can be flexibly ingested from third party MOOC providers.

First an analysis is made of different architectural patterns that facilitate such infrastructure for federated search. This includes an analysis of search interfaces at the one hand side and harvesting solutions at the other hand. Next, the architecture for ECO is presented. This consists of a solution for metadata and services.

2. Federated Search

Architecturally, a federated search system component captures functionality that provides unified search access to many digital preservation layers (a MOOC platform in the case of ECO). Such a system component can be realized in two ways.

The federated search system component uses a search protocol to broker queries to third party MOOC providers. (Ternier et al., 2005) illustrates how such search protocol based solution can be realized in peer-to-peer networks and client/server based solutions.

Figure 1 illustrates a typical federated search infrastructure. The federated search engine is the core component and gives transparent search access to a collection of repositories. This component offers a search interface through which a search client can send queries into the network.

A registry component typically contains all the search targets into which the query will be distributed. This is a self-contained component that offers its functionalities via web services. Apart from the location of the repository it can also list the type and version of the search protocol that is used or document other services that are offered by the MOOC platform. At the bottom of this figure, MOOC platforms are represented which the federated search engine can search.

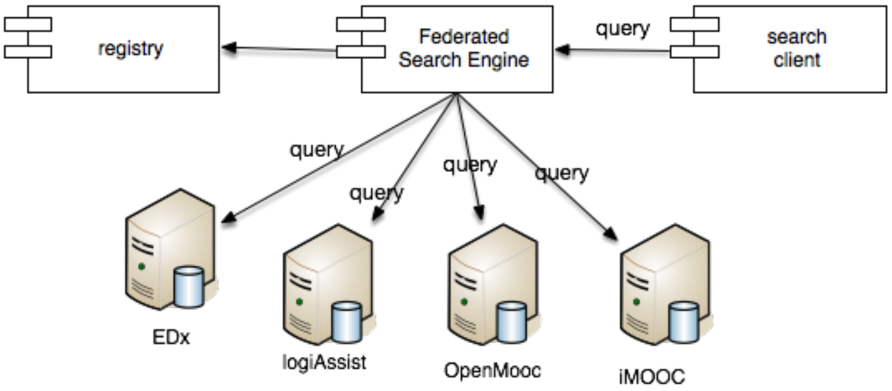

Figure 1: Federated Search Architecture that federates queries

A federated search infrastructure operates in the following steps:

1. A user formulates a query using the search client component. E.g. “sMooc”.
2. The search client encodes this query and sends the query to the federated search engine.
3. The federated search engine consults the registry and fetches a list of MOOC providers that provide a search API.
4. The federated search engine, sends the query to all MOOC providers
5. All MOOC providers execute the query and send a list of results back to the federated

search engine

6. Upon completion of all queries, the federated search engine merges and ranks the results.
7. The federate search engine sends the list of results to the search client that presents the results to the user.

In the past, such a solution for federated search was favourable for many practical reasons including storage capacity and network bandwidth. A single server was not capable of storing millions of cached metadata records. Also the process of synchronizing metadata over a network was considered too expensive. In addition to this, federated search infrastructures were favorable as they always searched in the most recent version of the metadata. When a user updates a metadata instance in e.g. the eDX component, this change is immediately visible for the search client.

However, federated search installation suffer from some important disadvantages:

1. Scalability. The amount of queries that a single platform has to execute becomes potentially very large. Typically, within a network the users of various platforms exchange queries. Before, users of e.g. iMOOC were only searching the iMOOC platform. After implementing federated search, iMOOC users can search OpenMOOC, weMOOC and eDX in addition. This leads to a situation where every MOOC provider has to serve queries for the sum of all users of the different participating MOOC providers. This leads to a much higher query load on every individual platform.
2. Speed. Since Google, many users expect queries to result in an answer within seconds. A federated search engine can implement two result merging strategies. (1) Fastness. Return a result as soon as one MOOC provider answers (2) completeness. Return results as soon as all MOOC providers have replied. Option 1 leads to less optimal results as slower repositories will never show up in the user result list (even if the quality is better). Option 2 leads to a system that is as slow as the slowest provider.

3. Metadata harvesting

The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) is a protocol that allows for making local copies of metadata records. This protocol stems from the digital libraries domain and was established in the late 1990s. With OAI-PMH a repository or more specifically, a MOOC provider, implements a simple REST XML based web service through which a third party application "the harvester" can obtain copies of the metadata describing the resources.

Selective harvesting involves obtaining a subset of metadata that a repository offers. For instance, using a date span, a third party application can obtain all metadata records for which a MOOC was changed within the given date span. A server can define multiple sets that enable filtering metadata records. Through selective harvesting, the supported sets can be queried ('ListSets' in OAI-PMH terminology). Next, only records related to one of the sets can be retrieved.

ECO leverages the OAI-PMH protocol to internally publish MOOCs to EcoPortal. That is, all MOOC platforms publish their resources as an OAI-PMH feed. The ECO Backend (the "harvester") checks these feeds hourly for new content. That actual list of offered ECO MOOCs is reflected on the EcoPortal. The EcoPortal implements a search interface through which users can search through the harvested MOOC records.

The advantages of metadata harvesting are:

- Scalability. A central harvester contacts and harvest the individual repositories. A new MOOC providers entering the network does not lead to extra load for the existing providers
- Speed. Queries are executed centrally on a metadata cache. This cache is easy to upgrade for performance or can be duplicated for scalability reasons.

The disadvantage compared to federated queries is that search results can be stale. If a harvester synchronizes every hour, a query result is potentially stale for one hour.

4. Dissemination MOOCs to third party applications

The DOW details that

"the OAI-PMH harvesting protocol will be implemented on the MOOC platforms used in ECO environment. This will enable ECO MOOC metadata to be disseminated to third party search engines such as Globe, Laclo and Wikiwijs"

Architecturally there are two ways in which this can be realized with OAI-PMH:

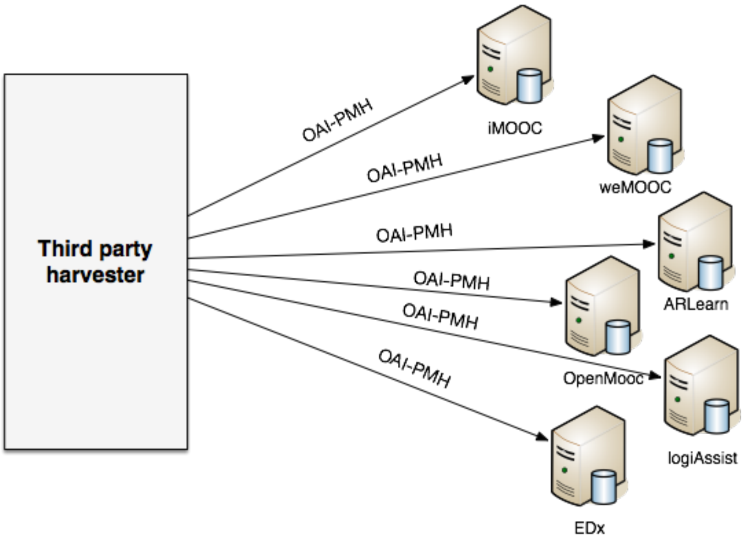

Figure 2: Third party application direct harvesting to ECO partners

A first 'naive' option is to give a third party applications such as Laclo or Globe direct access to the MOOC partners. This option is much in line with way standard harvesting architecture work. Standard architectures typically rely on a registry of OAI-PMH targets and harvest from all targets. Some architectures (e.g. Globe) implement a custom registry that is specialized in a domain (e.g. learning object repositories). An advantage of using this architecture is that the harvester gets direct access to each repository. When a record is changed or altered in this repository, the harvester will receive the modifications on the next harvesting session.

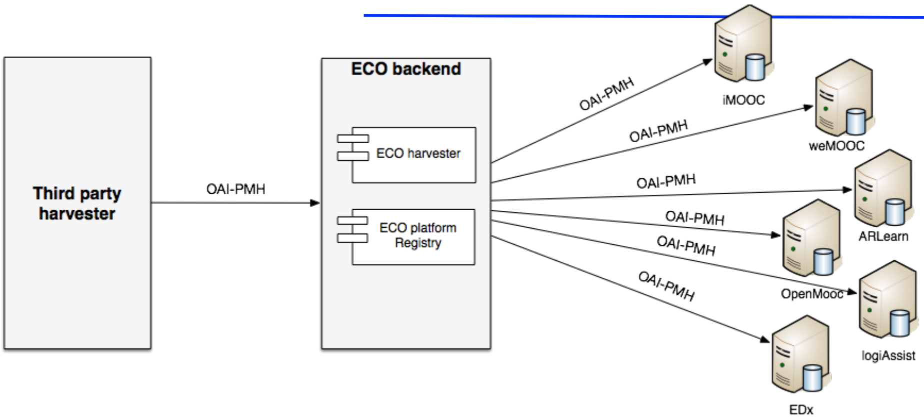

Figure 3: Third party applications harvests via ECO proxy

In Contrast to direct harvesting, the architecture in Figure 3 offers indirect harvesting to the ECO MOOC providers. The ECO backend component in this architecture is responsible for harvesting the MOOC providers and offers an OAI-PMH target to third parties through which they can obtain an aggregation of all ECO MOOC metadata.

There are many advantages of this harvesting approach. The ECO backend provides uniformity in metadata and can check if the MOOC providers deliver metadata that is inline with the ECO metadata application profile. When metadata is missing, it can provide this. E.g. implicit metadata like “this course is a MOOC” can be added. Such metadata often doesn’t make sense within the ECO MOOC federation, but does make sense outside the federation for third party harvesters that are unaware of the MOOC context. ECO has defined its own classification scheme, an agreement between ECO partners. Again, the backend can provide a mapping here to existing classification schemes that are relevant outside. Defining a new mapping is easier on a central component compared to implementing a new mapping distributed at every MOOC provider.

This architecture has limitations. Third party providers usually already harvest a network of repositories. This list of repositories should not overlap with the list of repositories that the ECO backend is connected to. The ECO backend must also introduce metadata that details when a record was harvested from the MOOC providers. If this is not done, the following scenario illustrates how metadata harvesting would fail:

1. At 08:00am, november 16th, a metadata instance is altered at OpenMOOC.
2. At 12 o’clock the same day, the third party harvester (e.g GLOBE) harvests new metadata instances from the ECO backend. This harvester harvests every 12 hours and specifies it wants data that was altered between 00:00AM and 12:00AM this day. This harvester will not harvest the change in step 1 as the ECO backend harvester has not run yet.
3. Next at 1PM, the ECO backend starts harvesting changes between 00:00AM and 12AM. As a result, the backend copies and stores the modified instance. Now the ECO backend contains the instance that was modified in step 1.
4. A bit later, the GLOBE harvester again selectively harvests instances that were changed between 12AM and 12PM. It again misses the instance altered in step 1 as it was changed before this time.

The solution for this is, that the ECO backend defines a last modification date per record. When a the backend makes a modification to the metadata, it sets the last modification timestamp to the current time (and not to the time when it was altered at the origin). Applied to the previous example, this results in a last modification of “1PM” for the record changed in step 1. When the ECO backend harvests a metadata record it will set the last modification date to “now”. As a result, GLOBE will not miss the record in step 4 as 1PM is contained within the range 12AM - 12PM.

5. Implementation of federated search in ECO

Taking the above analysis into account, the ECO OAI-PMH target will be implemented as follows.

5.1. OAI-PMH Sets

Every OAI-PMH target can arbitrarily define sets. A set enables selective harvesting.
The ECO backend OAI-PMH target will implement a set label for every participating MOOC:

- OpenMOOC
- iMOOC
- ARLearn
- weMOOC
- eDX
- LogiAssist

When a third party provider specifies “OpenMOOC” as a set, only openMOOC records will be retrieved. In this way, a third party provider can access the content of one MOOC provider.
In addition ECO makes a distinction between

- EcoBase. A list of ECO approved MOOCs. These are MOOCs that are created by the ECO partners and listed in the DOW
- EcoUser. A list of user generated MOOCs in ECO.

An ECO MOOC can only belong to either one of these two sets.

Furthermore, the target will support the concatenation of these two set definitions. E.g. “EcoBase:OpenMOOC” will result in the the OpenMOOC records that were created by ECO partners.

5.2. Provenance

OAI-PMH defines a provenance record to define the origin of a metadata record. E.g.

```
<provenance
      xmlns="http://www.openarchives.org/OAI/2.0/provenance"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/provenance
      http://www.openarchives.org/OAI/2.0/provenance.xsd">
    <originDescription harvestDate="2002-02-02T14:10:02Z" altered="true">
      <baseURL>http://eu.ecolearning.hub8/oai-pmh</baseURL>
      <identifier>oai:eu.ecolearning.hub8:56</identifier>
      <datestamp>2015-12-01</datestamp>
      <metadataNamespace>http://www.openarchives.org/OAI/2.0/oai_dc/</metadataNamespace>
    </originDescription>
  </provenance>
```

This record will define the originating repository of the metadata statement. Linking this “reharvested” statement to its origin.

5.3. Planning

As stated in the DOW, Task 3.6 started in M19 and ends on M36.
The fact that the ECO learning resources are visible to external sources via the OAI-PMH interface can help the dissemination of the courses. Especially when users start creating their own courses. These activities are planned to start in Q2 of 2016. WP3 plans to have the OAI-PMH backend interface ready at that time.

The development of this interface consists of two steps:

1. Adding a caching mechanism to the existing MOOC harvesting logic, which will keep track of changes in content by using the last modification timestamp as described in chapter 3. This functionality will be ready in Q1 of 2016
2. Implementing an official OAI endpoint following the official OAI-PMH specification. This specification can be found here:
   http://www.openarchives.org/OAI/openarchivesprotocol.html
   This functionality is planned to be ready in Q2 of 2016

6. Conclusion

This deliverable presents an open and extensible framework for federated search. This deliverable has presented a comparison between an architecture that federates queries and an architecture that federates metadata into a metadata cache. The conclusion is that federating metadata offers many (scalability) advantages over federating queries.

This deliverable furthermore defines how third party stakeholders will be able to obtain the MOOC description so that they can include them into their proprietary federated search system. An ECO harvesting interface is defined that aggregates all ECO MOOCs. Via the use of OAI-PMH sets, selectively harvesting becomes possible.

References

Ternier, S., Olmedilla, D., and Duval, E. (2005). Peer-to-Peer versus Federated Search: towards more Interoperable Learning Object Repositories. In Kommers, P. and Richards, G., editors, Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2005, pages 14211428, Montreal, Canada. AACE