

# OPENER OAI REPOSITORY

September 14<sup>th</sup> 2007

version 0.4

Seek You Too B.V.

[info@cq2.org](mailto:info@cq2.org)

## Document History

<b>Date</b>	<b>Version</b>	<b>Author</b>	<b>Remarks</b>
11 September 2007	0.1	J.B. Jonkers	Initial version
	0.2	J.B. Jonkers	Modifications after peer review
13 September 2007	0.3	J.B. Jonkers	Completed initial version
14 September 2007	0.4	J.B. Jonkers	Modifications after peer review

## **Summary**

This document describes in detail how the RSS feed provided by the Open Universiteit is used to create and maintain an OAI-PMH Interface. The software is based on the Meresco Core Component Library.

## **Table of Contents**

1.Intended audience.....	5
2.Introduction.....	6
3.Functional Description.....	7
3.1.Reading.....	7
3.2.Crosswalking.....	7
3.3.Storing.....	7
3.4.Accessing.....	7
4.Application DNA.....	8
5.Installation.....	9
5.1.Installation script.....	10
5.2.Operating.....	10
6.Configuration.....	11
7.Abbreviations.....	12

## **1. *Intended audience***

This manual is intended for those interested in knowing how the Meresco Core Component Library is used to setup and maintain an OAI-PMH Interface based on the EduCommons RSS feed provided by the Open Universiteit. It describes how, using the Meresco Application DNA, the RSS feed is read, crosswalked and stored. Furthermore this manual describes how the Application DNA also provides means to submit OAI-PMH queries to the server to retrieve the processed data.

## **2. Introduction**

The Open Universiteit provides a data stream based on their EduCommons system which contains Dublin Core (DC) records encapsulated in a RDF Site Summary (RSS) v1.0. These records need to be extracted from this RSS stream and stored in both Open Archives Initiatives Dublin Core (OAI-DC) format and the Institute of Electrical and Electronics Engineers (IEEE) Learning Objects Metadata (LOM) format v1.0.

The stored records then have to be able to be retrieved using the Open Archives Initiatives Protocol for Metadata Harvesting (OAI-PMH).

### **3. Functional Description**

Based on the requirements, the Opener OAI system has four tasks to perform: read, crosswalk, store and access. Most of these tasks are performed by components from the Meresco Core Component Library.

#### **3.1. Reading**

The EduCommons RSS feed is read and the data is fed to the processing part of the software. The contents of the RSS feed will be fetched at the predefined interval. This ensures that changes to the original data will propagate to the OAI-PMH interface in a timely fashion.

#### **3.2. Crosswalking**

After receiving the complete RSS feed, the processing part of the software will extract all items present in the RSS feed. Each item is then converted into OAI-DC and IEEE LOM. The transition from source records into OAI-DC and IEEE LOM records is done using the Crosswalk Component of Meresco. This component converts the given record into the specified metadata formats by applying rewrite rules. These crosswalk rules are provided by the Opener OAI system.

#### **3.3. Storing**

The newly created records are stored in a Storage provided by storage software. All records are stored in plain text format with human-readable filenames. The storage has built-in version control, which is used to tell new record from old ones.

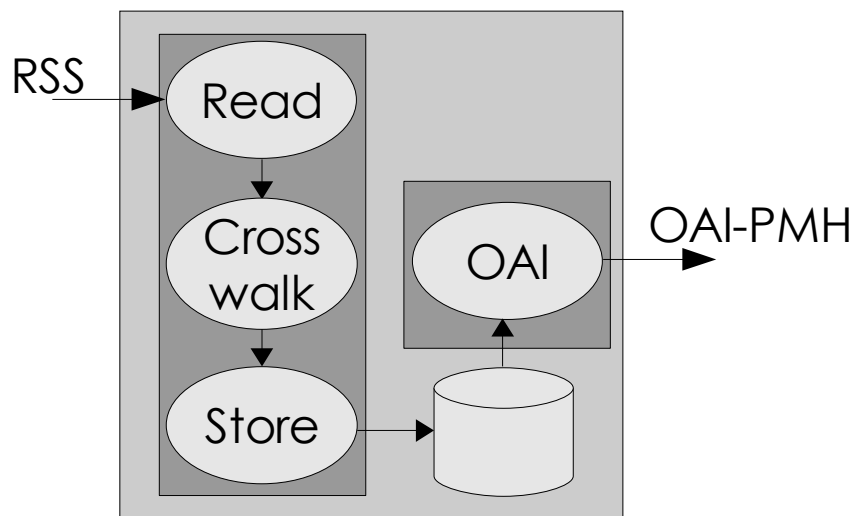
In order to provide OAI-PMH access on the records, additional information is stored for every record. This information is stored in separate files.

#### **3.4. Accessing**

The OAI-PMH access on the data is provided by the Meresco OAI-PMH component. This component provides OAI-PMH version 2.0 compliant access to the stored data.

## 4. Application DNA

The so called Application DNA is a description of strands of components that together make up a complete system that is set to perform one or more tasks. In this case, the Application DNA describes two strands which work independently of each other. Although they are capable of working independently, there is a connection between the two strands. One strand determines how to read, perform the crosswalks and store the information, while the second strand describes how the user can access the stored data.



*Illustration 1: Application Overview*

The illustration above shows the two strands of Application DNA (dark gray blocks) within the application. On the left side is the input (RSS) which flows through the components configured by the Application DNA strand into the storage. From there it can flow through the components configured in the second strand to the user, in the form of OAI-PMH.



## 5. Installation

The Opener OAI software has been developed to operate on a unix based operating system. Development has taken place on Debian based Linux computers and therefore, Debian is the preferred Linux Distribution to operate the Opener OAI software.

The Opener OAI software has various dependencies. Dependencies maintained by CQ2 are distributed with the Opener OAI Repository software in so called unix tarballs.

Dependencies:

Software component	Required version
Meresco Component Library	Version 2.1
Storage	Version 5.0
Weightless	Version 0.19
CQ2Utils	Version 5.0

The Opener OAI Repository also has 3<sup>rd</sup> party dependencies. These are not automatically installed and are not included in the Opener OAI Repository distribution.

Software component	Required version	Location
Python	2.4	<a href="http://www.python.org">http://www.python.org</a>
Python-lxml	1.2.0	<a href="http://codespeak.net/lxml">http://codespeak.net/lxml</a>
RCS	5.7	
Amara	> 1.1.7	<a href="http://uche.ogbuji.net/tech/4suite/amara/">http://uche.ogbuji.net/tech/4suite/amara/</a>
4Suite	1.0.2	<a href="http://cheeseshop.python.org/pypi/4Suite-XML">http://cheeseshop.python.org/pypi/4Suite-XML</a>
PyLucene	2.0.0	<a href="http://pylucene.osafoundation.org">http://pylucene.osafoundation.org</a>
PyXML	0.8.4	<a href="http://pyxml.sourceforge.net">http://pyxml.sourceforge.net</a>

The system administrator of the system the Opener OAI Repository software is being installed on has to take care for the installation of the packages listed above. Without the packages listed above the Opener OAI Repository software developed will not operate. The supplied installation script will perform checks if the third party packages are installed and are operating correctly. If the Debian distribution for Linux is used, these packages can be installed using the aptitude software package manager.

## 5.1. Installation script

The installation procedure for the Opener OAI Repository is simple and straightforward:

1. Create a user account which will be used to run the Opener OAI Repository:

```
$ useradd -m oairepo
```

2. As that user, extract the supplied `opener_oai-1.0.tar.gz` file:

```
$ su - oairepo
```

```
$ tar xzf opener_oai-1.0.tar.gz
```

```
$ exit
```

3. As the root user, run the `install.sh` script located in the `install` directory:

```
$ cd ~oirepo/opener_oai-1.0/install
```

```
$ ./install.sh
```

If desired, the installation script can install the PyLucene package. This also means that a required library, `libgcsj5` will be installed in the `deps.d` directory. This directory needs to be added to the `LD_LIBRARY_PATH` environment variable in order for the Opener OAI Repository software to operate properly.

**NOTE:** The Opener OAI Repository software can be installed in any desired directory. The directory in which the software is extracted will be the directory in which the installation script will install the software.

## 5.2. Operating

After installing the Opener OAI Repository software it can be started using the `startserver.sh` script located in the `opener_oai` directory. This script will start the server side of the Opener OAI Repository and will start the download cycle for the EduCommons data source.

The Opener OAI Repository uses the standard out device for logging. This way it can be redirected to whichever location is desired.

There are various ways to start the server automatically. There is no default mechanism supplied with the Opener OAI Repository software. The system administrator can therefore use the `startserver.sh` as a basis to set up the Opener OAI Repository software as desired.

**Recommendation:** For running the Opener OAI Repository server in a simple but very effective manner, have a look at the `daemontools` package. See <http://cr.yip.to/daemontools.html> for more details.

The URL at which the Opener OAI Repository can be reached is:

```
http://<servername>:<port>/oai/?<OAI-arguments>
```

## **6. Configuration**

The Opener OAI software requires a few parameters in order to operate. The values for these parameters will be asked of the administrator at the start of the installation script. Below is a short explanation of those requested parameters.

### **In what directory should the storage be put?**

The Opener OAI Repository requires a directory in which it can store the metadata required to operate. Also, the DC and LOM records will be stored in this directory. The directory can be at any location desired, as long as the user that runs the Opener OAI Repository software has read and write access.

### **At what port number should the server be listening for OAI requests?**

The Opener OAI Repository needs a port number at which the server application will listen for OAI-PMH requests. NOTE: if the software is installed on a server shielded by a firewall, make sure the given port number is accessible from the outside.

### **What is the URL for the EduCommons data source?**

The Opener OAI Repository checks the given EduCommons data source URL to see if there is any new data available that needs to be fetch, crosswalked and stored. The specified URL needs to comply with Request For Comment (RFC) 1738 (<http://www.faqs.org/rfcs/rfc1738.html>)

### **At which interval should the EduCommons data source URL be checked? (in minutes)**

This is the number of minutes that has to pass before the Opener OAI Repository software checks if there is any new data available at the specified EduCommons data source URL.

## 7. **Abbreviations**

CQ2	Seek You Too
DC	Dublin Core
IEEE	Institute of Electrical and Electronics Engineers
LOM	Learning Objects Metadata
OAI	Open Archives Initiatives
OAI-DC	Open Archives Initiatives Dublin Core
RDF	Resource Description Framework
RFC	Request For Comments
RSS	RDF Site Summary
URL	Uniform Resource Locator