

Implementing enhanced OAI-PMH requirements for Europeana

Nikos Houssos¹, Kostas Stamatis¹, Vangelis Banos², Sarantos Kapidakis³,
Emmanouel Garoufallou⁴ and Alexandros Koulouris⁵

¹ National Documentation Centre, Greece

² Veria Public Library, Greece

³ Laboratory on Digital Libraries and Electronic Publishing, Department of Archive and Library Sciences, Ionian University, Greece

⁴ Technological Educational Institute of Thessaloniki, Greece

⁵ Technological Educational Institute of Athens, Greece

nhoussos@ekt.gr, kstamatis@ekt.gr, vbanos@gmail.com, sarantos@ionio.gr,
mgarou@libd.teithe.gr, akoul@teiath.gr

Abstract. Europeana has put in a stretch many known procedures in digital libraries, imposing requirements difficult to be implemented in many small institutions, often without dedicated systems support personnel. Although there are freely available open source software platforms that provide most of the commonly needed functionality such as OAI-PMH support, the migration from legacy software may not be easy, possible or desired. Furthermore, advanced requirements like selective harvesting according to complex criteria are not widely supported. To accommodate these needs and help institutions contribute their content to Europeana, we developed a series of tools. For the majority of small content providers that are running DSpace, we developed a DSpace plug-in, to convert and augment the Dublin Core metadata according to Europeana ESE requirements. For sites with different software, incompatible with OAI-PMH, we developed wrappers enabling repeatable generation and harvesting of ESE-compatible metadata via OAI-PMH. In both cases, the system is able to select and harvest only the desired metadata records, according to a variety of configuration criteria of arbitrary complexity. We applied our tools to providers with sophisticated needs, and present the benefits they achieved.

Keywords: OAI-PMH, Europeana, EuropeanaLocal, Tools, DSpace Plug-in, Interoperability, Information integration, Metadata harvesting, Europeana Semantic Elements

1 Introduction

Europeana is an evolving service, which will constitute an umbrella of European metadata from distributed cultural organisations. Europeana currently gives access to more than 14 million items representing all Member States including film material, photos, paintings, sounds, maps, manuscripts, books, newspapers and archival papers.

The Europeana service [1] is designed to increase access to digital content across Europe's cultural organisations (i.e. libraries, museums, archives and audio/visual archives). This process will bring together and link up heterogeneously sourced content, which is complementary in terms of themes, location and time. Europeana's active partner network consists of 180 organisations till now.

In order to achieve these goals, the European Union launched in June 2008 the EuropeanaLocal project in the framework of the eContentPlus program. Up to June 2011, the EuropeanaLocal partners aim to make available to Europeana more than 20 million items, held across 27 countries. At the same time, they are committed to exploring and developing efficient and sustainable processes and governance procedures so that the growing numbers of regional and local institutions can easily make their content available to Europeana in the future by adopting and promoting the use of its infrastructure, tools and standards [2].

Greece is participating in EuropeanaLocal with content providers and the Hellenic Aggregator created and supported by the Veria Central Public Library (VCPL). Since March 2010, 10 content providers, from which 7 use DSpace, have followed closely the Europeana standards, thus implementing full support for Europeana Semantic Elements (ESE) and have been harvested successfully by the VCPL Aggregator (<http://aggregator.libver.gr>) and Europeana [3]. In March 2011, the Hellenic Aggregator provided 130.000 items to Europeana.

One of the most important aspects in the process of creating a Europeana Compliant digital repository is the support for ESE, which is virtually a new Dublin Core Profile, developed by Europeana in order to fulfill its operational requirements. Existing digital repository software in general does not support ESE by default as it is the case with Dublin Core. Nevertheless, the nature of the formats makes it feasible to alter existing software and data in order to add support for ESE. Specific information about the process can be found at the DSpace plugin for Europeana Semantic Elements webpage [4], developed by the Veria Central Public Library (VCPL) and the Hellenic National Documentation Centre (EKT).

The first step in the process is to use the Europeana XML Namespace <http://europeana.eu/schemas/ese/> and augment existing systems' configuration in order to support the additional ESE elements. After implementing ESE support, the repository has to be populated with the appropriate metadata values. This task can be either performed manually through the appropriate user interface of each digital library or automatically by using special software tools developed for this purpose. It must be noted that due to the wide usage of the DSpace software internationally and in Greece, the focus has been the implementation of tools for this specific platform.

Except from DSpace and other modern digital repository platforms, there are also numerous digital libraries built with older or closed source technologies or legacy software which do not support OAI-PMH or any other form of automatic metadata exchange. In these cases, special techniques should be applied in order to extract metadata through plain HTTP requests, for example the DEiXTo tool.

DEiXTo (or ΔEiXTo) [5] is a powerful freeware web data extraction tool, based on the W3C Document Object Model (DOM), created by an independent software developer. It allows users to create highly accurate "extraction rules" (wrappers) that describe what pieces of data to scrape from a web page. When used appropriately, DEiXTo can extract meaningful metadata from web pages of non standards compliant

digital content collections and generate appropriate Dublin Core and ESE records. These records can be utilised by any standards compliant metadata harvester in order to be included in Europeana.

This paper analyses a toolset for data providers that mainly targets owners of small collections that are running DSpace (i.e. the DSpace plug-in, which converts and augments the DSpace metadata according to Europeana ESE requirements) as well as systems with different software, incompatible with OAI-PMH. Focus is also on the system ability to select and harvest only the desired metadata records, according to a variety of configuration criteria of arbitrary complexity that is applied in both cases.

The structure of the rest of the present text is as follows: Section 2 describes the advanced harvesting requirements addressed by our solution and the motivation based on practical needs of data providers. Section 3 presents related work and section 4 elaborates on the actual solution. Section 5 describes the application of the proposed approach in real use cases, while the last section of the article provides summary, conclusions and plans for further work.

2 The Case for Enhanced OAI-PMH Compliant Data Providers

The ubiquitous OAI-PMH protocol provides an interoperability framework based on metadata harvesting. Two types of entities exist in a typical OAI-PMH interaction: the data provider that exposes metadata to interested clients and the service provider that offers value-added services on top of metadata collected from data providers.

The recent proliferation of repositories worldwide has created a favourable environment for the emergence of content aggregators that act as OAI-PMH service providers collecting metadata-only records from individual data sources. Aggregators provide unified search and browse functionality as well as the foundation and infrastructure for advanced value-added services that become particularly meaningful when provided over content of substantial size. A number of important aggregators with international coverage and diverse scope have entered the scene in the last few years. Distinctive examples are Europeana, the European digital heritage gateway, DRIVER and OpenAIRE (repositories of peer-reviewed scientific publications) and DART Europe (European portal to research theses and dissertations).

Compatibility with aggregators is nowadays a *sine qua non* pre-requisite for repositories, since it provides increased visibility, enables content re-use and allows participation of individual collections to the evolving global ecosystem of interoperable digital libraries. In this context, it is becoming an increasingly common requirement for repositories to provide for retrieval by an aggregator only a subset of the metadata records it contains, essentially enabling *selective harvesting*. This may be needed for various reasons; certain indicative use cases include the following:

- The aggregator collects only records that meet specific criteria concerning IPR, copyright and open access:
 - Records are included in the harvesting set only when there is a freely accessible digital item (eg full text articles, books, etc.). Such policies are followed by Europeana, DRIVER, OpenAIRE and DART Europe.

- Only metadata records which are themselves freely available for various uses, ideally through appropriate licensing (e.g. Creative Commons). This is required, for example, by Europeana.
- Thematic aggregators collect only records for content in specific subject areas, while individual repositories can be interdisciplinary. Such is the case with the VOA3R aggregator on Agriculture and Aquaculture. Europeana can be also considered an analogous example, since in initial stages of development concentrates on collecting mainly cultural heritage content (e.g. peer-reviewed journal articles are not included).
- The aggregator collects only records for content of a specific type (e.g. theses, like DART Europe), while individual repositories may contain different types.

The above indicate the complexity of supporting selective harvesting. This requirement becomes more difficult to achieve when you consider that a repository is likely to provide records to more than one aggregators, each with different requirements. Typically, OAI-PMH sets are implemented within repository platforms in a static fashion, through the creation of one set per individual collection in the repository. This approach is clearly not sufficient because, as is evident from the above examples, the desired sets to harvest may contain records spread over different collections. For practical needs to be satisfied and capabilities provided by the OAI-PMH sets specifications to be fully exploited, more sophisticated mechanisms are required, for example “virtual” sets that are dynamically formed per request based on specific conditions – a solution perfectly compatible with OAI-PMH.

Another important aspect and use case of selective harvesting is the retrieval of records from systems that are not compliant with OAI-PMH. These might include legacy systems like custom, non-standard databases, bibliographic catalogs of Integrated Library Systems connected with the corresponding digital material, etc. A common case is that such systems contain an array of diverse records, many of them not relevant for particular aggregators. Therefore, filtering needs to be applied, possibly according to complex criteria with a local, collection-dependent character. Crucial aspects for the success of this task are the adoption of a systematic way of implementing and injecting into the harvesting logic the filtering functionality, as well as repeatability of this procedure that enables periodic updates of metadata in the aggregator that reflect changes of records within the source systems. It is worth noting that the optimal option for content providers of this kind would be to provide their digital content through a repository platform, so that a holistic, standards-compliant solution is applied for the management of their digital material and metadata, enabling advanced services such as digital files preservation, curation, persistent identification, full-text indexing, etc.; however, this might not be feasible in the near term (e.g. due to lack of resources).

Addressing the above requirements and issues constitute the main aims of the system and approach presented in this paper, elaborated in Section 4.

3 Related Work

Mazurek et al [6], present the idea, role and benefits of a selective harvesting extension of the OAI-PMH protocol, developed and applied in Polish digital libraries in frame of the ENRICH project. Specifically, they describe the OAI-PMH protocol extension developed by the Poznan Supercomputing and Networking Center, which allows harvesting of resources based on a search query specified in the Contextual Query Language. This selective harvesting extension is being used by the Polish national aggregator, which enables extended selective harvesting at the national level. It is notable that in this approach filtering criteria are specified directly from the side of the aggregator.

The concept, implementation and practical application of the OAI-PMH protocol extension is also presented at the Mazurek, Mielnicki and Werla [7] JCDL 2009 poster.

Finally, Sanderson Young and LeVan [8], briefly contrast the information retrieval protocols SRW/U (the Search/Retrieve Web service) and OAI (Open Archives Initiative), their aims and approaches, and then, they describe ways in which these protocols have been or may be usefully co-implemented.

A common limitation of the aforementioned approaches is that data is retrieved from data sources through queries in standard query languages like CQL. In practical situations it is frequently the case that such queries cannot fulfill the custom and complex selective harvesting requirements for data providers, as demonstrated also in the use case of paragraph 5.2. Furthermore, this solution requires a full-fledged query language to be implemented against a variety of back-end systems / data sources, while the approach proposed in this paper requires from data providers to implement only the specific bulk data loaders and filters that are necessary / useful in their particular case.

The University of Minho has developed an OAI Extended AddOn for DSpace [9], which enables selective harvesting through the incremental, piece-wise addition of objects like filters in the OAI-PMH server. The solution is bound to DSpace and does not support retrieval from legacy, non OAI-compliant sources, since, compared with our approach, there is no abstraction neither of the data records nor the data loading and output generation functionalities.

4 An Innovative Approach to Implementing Enhanced Data Providers

The main idea of our approach is to enhance an OAI-PMH server (data provider) with a number of important capabilities particularly related to selective harvesting, while maintaining full compatibility with the protocol and respecting the OAI-PMH “contract” towards clients. These capabilities are the following:

- Dynamic definition of sets and their membership, possibly based on complex criteria that do not correspond to the coarse-grained and static classification of repository records in pre-defined sets and cannot be expressed with typical query languages used by systems like federated search platforms.

- A systematic way to introduce to an OAI-PMH server implementation advanced logic necessary for selective harvesting such as transformations among different formats and schemata, filtering and updating of data. Incremental development and piece-wise enhancement of selective harvesting logic at fine levels of granularity are important relevant requirements as is the simplicity and separation of concerns among developers of different parts of the OAI-PMH data provider. For example, the technical person creating or updating filters and crosswalks for the implementation of harvesting use cases should not need to be aware of harvesting or OAI-PMH specific technology and can thus concentrate on improving the filtering or update functionality per se.
- Support of a modular implementation that enables retrieval of metadata records from a variety of non OAI-PMH sources via simple extensions to the core architecture for data loading, transformation and exporting in the desired formats and schemata. This is highly important, since vast sets of important content are “hidden” behind legacy, custom-made applications that do not follow state-of-art interoperability standards and are thus deprived of their potentially significant impact for end users and other stakeholders like value-added services developers.

To achieve the above, we have designed according to these principles and developed a modular component called transformation engine. This component has been successfully incorporated in OAI-PMH server implementations for two types of systems: (a) OAI-PMH-compliant repositories, in particular running the DSpace platform, that have been enriched with selective harvesting functionality and (b) Z39.50-compliant bibliographic catalogs of metadata records, possibly with links to digital material, that have been enhanced with OAI-PMH data providers which enable pre-processing, mapping metadata entries to OAI-PMH clients requirements and also support repeatability of the procedure at periodic time intervals, as is common for OAI-PMH compliant sources.

The rest of this chapter is structured as follows: First, a detailed description of the transformation engine is provided, followed by a report on the implementation of the two aforementioned distinct use cases.

4.1 The Transformation Engine

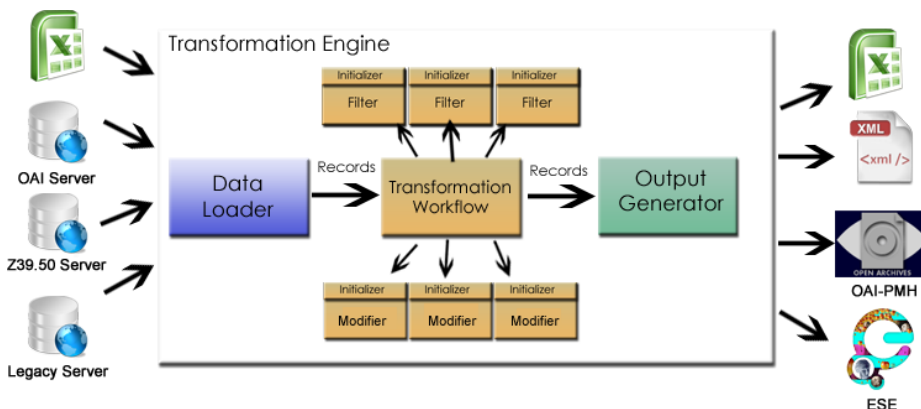


Figure 1. Architecture of the transformation engine.

The transformation engine is a generic framework for implementing data transformation workflows. It allows the decoupling of communication with third party data sources and sinks (e.g. loading and exporting/exposing data) with the actual tasks that comprise the transformation. Furthermore, it enables the decomposition of a workflow into autonomous, modular pieces (transformation steps), facilitating the continuous evolution/re-definition of workflows to constantly changing data sources and the development of fine-grained workflow extensions in a systematic way. It is worth noting that the transformation engine is an independent component that is used in a modular fashion in the proposed toolset. It has been used by EKT as an autonomous module in a variety of contexts, for example for the population of digital repositories of Greek public libraries [10] with metadata from ILS catalogues.

A key aspect of the engine's design is the *Record* abstraction. Metadata records are represented by a hierarchy of classes extending the abstract Record class. A simple common interface for all types of records proved adequate to allow complex transformation functions. Examples of record implementations that have been implemented and used until now concern UNIMARC, MARC21, Dublin Core, ESE, various structured formats for references (e.g. BibTex, RIS, Endnote) while there is also a more general abstraction for XML records. The main methods of the Record interface are shown in the following:

```
public abstract List<String> getByName(String elementName);
public abstract void removeField(String fieldName);
public abstract void addField(String fieldName,
ArrayList<String> fieldValues);
public void updateField(String fieldName, ArrayList<String>
fieldValues)
```

As depicted in Figure 1, data loaders are used to read data from external sources (e.g. files, repository databases, Z39.50 servers, even OAI-PMH data providers) and forward it to the transformation workflows in the form of a certain sub-type of Record. The output generators undertake the exporting / exposing of records to third party systems and applications. The transformation workflow(s) is the place where the actual tasks are executed. A workflow consists of processing steps, each of which falls most of the time into one of the two following categories: *Filters* determine whether an input record will make it to the output. *Modifiers* can perform operations on record fields and their values (e.g. add/remove/update field). *Initializers* initialize data structures that are used by processing steps. By using the record interface in the implementation of entities like filters and modifiers a great degree of separation of concerns is achieved (for example, knowledge of the specifics of MARC is not necessary for a developer to create a modifier that performs some changes on an input MARC record).

A workflow is defined as a series of processing steps in a configuration file outside the source code of the engine, in particular using the dependency injection mechanisms of the Spring framework. Thus, a transformation engine system can include many data loaders, output generators and transformation steps, but a specific scenario (being described a Spring configuration XML file) can make use of only some of them according to the user needs.

4.2 Extending the OAI-PMH-compliant Harvesting Server of a Repository

An obvious use case of the proposed mechanism is the enhancement of modern repository platforms that already support OAI-PMH with the aforementioned advanced functionality. In particular, we have incorporated the transformation engine in the OAI-PMH module of the DSpace platform, which is the most popular repository platform in Greece (also among the contributors to Europeana Local).

In the vanilla DSpace platform, the harvesting server receives requests through the DSpaceOAI Catalog module, where record filtering is performed, if required, according to the specifications of OAI-PMH, based on time stamps or set membership. Following this stage and before sending results to the client, the DSpaceOAI Crosswalk addresses adaptation of the returned records (e.g. modification of the exposed metadata schema, appropriate adjustments in field values).

This procedure is carried out by the DSpaceOAI Catalog and the DSpaceOAI Crosswalk classes depicted in Figure 2.

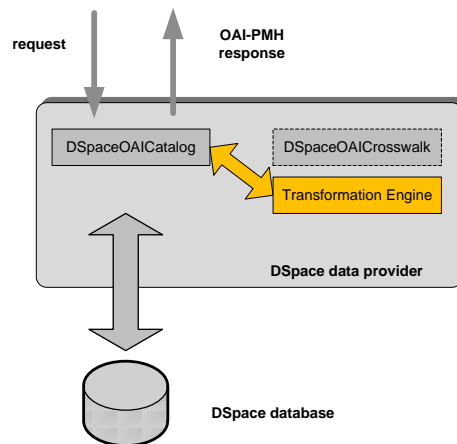


Figure 2. Enhanced DSpace data provider.

In the proposed enhanced version, the architecture of the DSpace data provider is modified as depicted in Figure 2. The tasks of record filtering and record adaptation according to the desired output schema (e.g. ESE) are handled by the Transformation Engine that is injected into the OAI-PMH server implementation, with Filters undertaking selection of records and Modifiers the work of the metadata crosswalk. Selective harvesting is based on virtual, dynamic sets. A virtual set is essentially defined as the set of repository records that results from a distinct transformation workflow, i.e. a series of specific filters and modifiers applied on repository metadata records, as specified in a Spring configuration file. If a particular record is not filtered during the workflow it is considered a member of the virtual set and is included in the record set returned to the client.

For the case of Europeana /ESE, specific user-defined classes have been developed and injected into the transformation engine (e.g. ESERecord, ESEOutputGenerator, ESEMappingModifier) in a straightforward manner, demonstrating the ease of system

customisation for developers which are due to the separation of concerns enforced by the engine's modular design.

4.3 Enabling OAI-PMH-compliant Harvesting of MARC/Z39.50 Data Sources

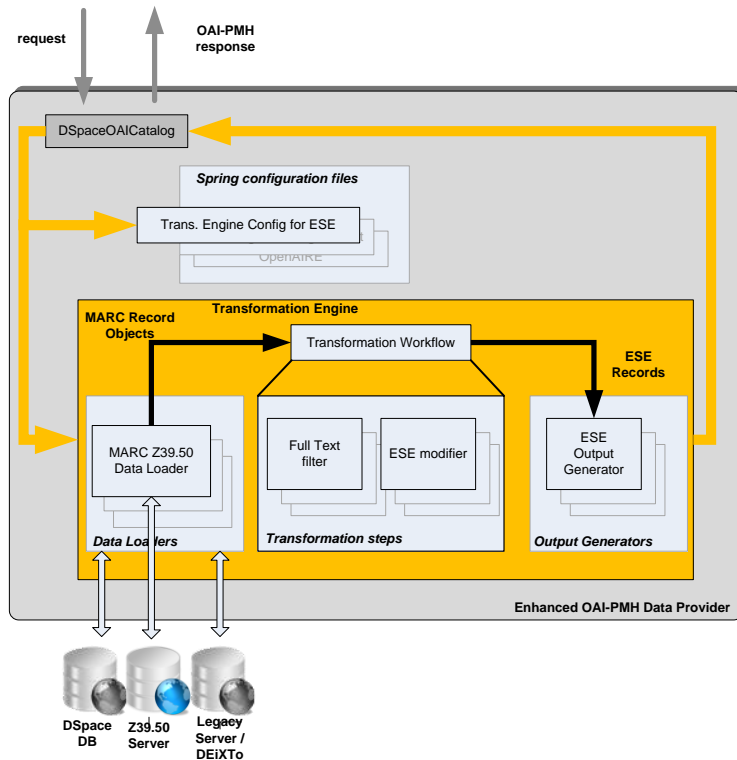


Figure 3. Architecture for OAI-PMH compliant harvesting of non OAI-PMH compliant data sources.

Large volumes of valuable content are hosted today in systems that are not compliant with OAI-PMH and thus providing them to aggregators like Europeana is a challenging task. In this use case, based on the DSpace OAI-PMH module, we have developed an OAI-PMH server that reads UNIMARC data records from Z39.50 data sources and serves them to OAI-PMH clients (and in particular Europeana), as depicted in Figure 3. To achieve this, we modified the DSpaceOAI Catalog so that upon receiving a request it triggers the transformation engine. A MARC/Z39.50 data loader is invoked first to get UNIMARC records (in ISO 2709 or MARCXML format) from a standard Z39.50 server, using the JZKit open source library, and transform them, based on the MARC4J tool, into MARCRecord objects (MARCRecord is an abstraction for MARC records following the aforementioned Record interface). These objects are relayed to the transformation workflow where filters are applied for tasks like rejection of records that do not have associated digital files (e.g. bibliographic records where full text is not available), de-duplication of

records (in real-life cases, duplicate records may result from retrieval from different collections, even within the same data source) and modifiers are executed to transform records to the ESE format and perform various modifications to field values (e.g. normalisation, adjusting value encoding to Europeana standards). Finally, an ESE output generator provides the output in the format prescribed by Europeana.

Moreover, as Figure 3 depicts, the *Transformation Engine* can include a pool of data loaders, output generators and transformation steps allowing the system to use any of them for providing data to dissimilar aggregators. And this is possible due to the system configuration which can be done outside the source code, through XML configuration files. These files are responsible to initialise the Transformation Engine with a specific set of transformation steps that will finally produce the right outcome for the specific aggregator. Thus, the same engine instance can produce totally different results depending on the needs of a particular aggregator / harvesting case.

It is worth noting that this approach makes the harvesting process periodically repeatable even when the underlying data sources are not OAI-PMH compatible. Furthermore, evolution and requirement changes are easily catered for due to the fine-grained extensibility and modifiability of the transformation engine (e.g. a change in requirements can be normally easily addressed by writing new filters / modifiers and including them in the processing workflow and/or by updating existing ones, without any modification of the core system).

A similar architecture but with more complex logic for data loading and mapping needs to be applied in the case of data sources not following standard metadata schemata, for example custom databases of digital material or even unstructured information in static web pages. Addressing the latter case can be assisted by tools like DEiXTo, which has been employed also within Europeana Local for collecting metadata from Greek sources.

5 Real use cases

5.1 The Environment and Data Sets

The Technical Chamber of Greece wants to contribute to Europeana collections that contain all their current publishing work (TEE digital library), some historical editions (1932-1980), and their multimedia content on engineers, buildings and posters.

The descriptions of these objects are in the UNIMARC format, mixed with descriptions without online objects, which are inappropriate for Europeana. Additionally, their own content management system provides the above 5 collections together with other content, from their own regional subdivisions, their journal subscriptions, etc. The right selection or records has to be performed before they become available to Europeana.

The metadata records that could be finally contributed to Europeana are approximately 6800. The most frequent metadata field is dc:subject, which is usually repeated at least 4 times, and the 28284 subjects that appear, contain 4669 unique values. The longest field is dc:title with 18 words on average and follows

dc:description and dcterms:isPartOf with 15, while the dcterms:isPartOf is used in the 97% of the records, and most fields are included once on each record.

Another case, corresponding to enhancing already OAI-PMH compatible data sources, has been the ability to provide virtual sets/collections of metadata records in the Greek National Archive of Doctoral Dissertation repository (<http://www.didaktorika.gr> / HEDI – a service operated by the National Documentation Centre) to harvesting clients. The respective repository contains more than 23.500 thesis records – each of them is assigned to one or more disciplines according to the Frascati classification. More than 1.000 of them belong to Agricultural Sciences class or its sub-classes and have been contributed to the VOA3R thematic aggregator (virtual repository) covering the areas of agriculture and aquaculture [11].

5.2 Two Practical Applications of the Approach

The most interesting and challenging case of application of the proposed system has been the delivery of ESE-compliant metadata from UNIMARC records in Z39.50 sources, which was done for the Technical Chamber of Greece. The retrieval of the desired sets of records was not possible using only queries (e.g. PQF or CQL) to the Z39.50 server, since the criteria for filtering were quite custom and complex, (e.g. availability of full-text that was specified in a non-standard way in the metadata records, filtering of records that are present in the database but are not published by the Technical Chamber of Greece, etc.) and also de-duplication of records was required. Using appropriate queries our data loader retrieves an unfiltered super-set of the appropriate record set, applies the filters, applies the mapping to ESE and produces and provides to clients the metadata in ESE format. The whole procedure is repeatable and transparent to harvesting clients, which receive the ESE data through OAI-PMH without being aware of the underlying complexity. Furthermore, development of filters and modifiers does not require any knowledge of the MARC and Z39.50 standards and the structure of MARC records.

In the second case, that of VOA3R, there has been the ability to provide virtual sets/collections of metadata records in the HEDI repository to harvesting clients. One virtual set is provided for each field of science and technology as specified in the Frascati classification – a relevant field exists in each metadata record. This scheme is being used to provide metadata from this repository to the VOA3R virtual repository.

6 Summary – Conclusions and Future Work

Global efforts, like Europeana, that address many small and heterogeneous content providers, have indicated the need for advanced tools, to handle common, or less common, content provider problems. We identified several of those needs, and developed appropriate tools, to facilitate the harvesting setup and configuration.

With the proposed approach, their OAI-PMH server can apply advanced logic for selective harvesting such as transformations among different formats and schemata, filtering and updating of data. Content providers can define dynamic sets to contribute to Europeana and memberships, without altering their collections. Even when their

software does not support OAI-PMH, they can use our modular implementation that enables retrieval of metadata records from a variety of non OAI-PMH sources.

We implemented these tool and extensions and used them in the context of Europeana providers, to cover their practical needs. This way, they do not have to perform such task manually, or re-implement functionality that others also implement or need, and their participation to Europeana will be easier and more flexible, according to their own collection setup and requirements.

Further work is being planned along various paths. The case studies provided clear indications that the proposed approach leads to very good performance both in terms of harvesting speed and consumption of computing and memory resources. A detailed investigation of performance issues is an interesting extension of the present work. Other plans include the incorporation of the developed modular tools into various open source OAI-PMH servers, as well as the application of the proposed approach with more content providers and a systematic user study to capture their experiences with the tools in terms of utility and ease of configuration and extension.

References

1. Koninklijke Bibliotheek: Europeana, <http://www.europeana.eu> (2009)
2. McHenry, O.: EuropeanaLocal – its role in improving access to Europe’s cultural heritage through the European Digital Library. In: 11th Annual International Conference «EVA 2008 Moscow», Moscow (2008), http://conf.cpic.ru/upload/eva2008/reports/dokladEn_1509.pdf
3. Koulouris, A., Garoufallou, E., Banos, E. (2010). Automated metadata harvesting among Greek repositories in the framework of EuropeanaLocal: dealing with interoperability. Proceedings of the 2nd Qualitative and Quantitative Methods in Libraries International Conference (QQML2010), Chania (2010)
4. Banos, E.: DSpace plugin for Europeana Semantic Elements (ESE), <http://vbanos.gr?p=189> (2010)
5. Donas. K.: DEiXTo, <http://www.deixto.com> (2010)
6. Mazurek, C., Mielnicki, M., Parkola, T., Werla, M.: The role of selective metadata harvesting in the virtual integration of distributed digital resources. In: ENRICH Final Conference, pp. 27--31 (2009)
7. Mazurek, C., Mielnicki, M., Werla, M.: Selective harvesting of regional digital libraries and national metadata aggregators. In: 9th ACM/IEEE-CS Joint Conference on Digital libraries (JCDL 2009), pp. 429--430, New York (2009)
8. Sanderson, R., Young, J., LeVan, R.: SRW/U with OAI: Expected and Unexpected Synergies. D-Lib Magazine, 11(2), (2005), <http://www.dlib.org/dlib/february05/sanderson/02sanderson.html>
9. OAI Extended AddOn, University of Minho (2011), <http://projecto.rcaap.pt/index.php/lang-en/consultar-recursos-de-apoio/remository?func=fileinfo&id=337>
10. Digital repositories of the public libraries of Serres and Levadia, <http://ebooks.serrelib.gr>, <http://ebooks.liblivadia.gr>
11. VOA3R EU project, www.voa3r.eu.