

MORE: A Micro-service Oriented Aggregator

Dimitris Gavrilis¹, Vangelis Nomikos¹, Konstantinos Kravvaritis¹, Stavros Angelis¹,
Christos Papatheodorou^{1,2(✉)}, and Panos Constantopoulos^{1,3}

¹ Digital Curation Unit – IMIS, Athena Research Center, Athens, Greece
{d.gavrilis,v.nomikos,k.kravvaritis,s.angelis,c.papatheodorou,
p.constantopoulos}@dcu.gr

² Department of Archives, Library Science and Museology,
Ionian University, Corfu, Greece

³ Department of Informatics, Athens University of Economics and Business,
Athens, Greece

Abstract. Metadata aggregation is a task increasingly encountered in many projects involving data repositories. The small number of specialized software for this task indicates that in most cases customized software is used to perform aggregation, which in turn relates to the highly complex tasks and architectures involved. In this paper, the metadata and object repository aggregator (MORE) is presented, which has been effectively used in numerous projects and provides an easy and flexible way of aggregating metadata from multiple sources and in multiple formats. Its flexible and scalable architecture exploits cloud technologies and allows storing content into different storage systems, defining workflows dynamically and extending the system with external services. One of the most important aspects of MORE is its curation/enrichment services which allow curation managers to automatically apply and execute enrichment plans employing enrichment micro-services in order to aggregated data.

Keywords: Metadata aggregation · Metadata interoperability · Enrichment · Scalable architectures · Micro-services · Metadata quality

1 Introduction

In many research, as well as industrial, data management applications an aggregation step is performed whereby data (or metadata) are aggregated from multiple sources into one database/system and from multiple formats into a unified/common one. The plethora of sources and formats can be explained by the diversity of technologies and requirements that exist. Metadata aggregation is the special case where the main resources being aggregated comprise metadata (not data). In the past years organizations like the European Library (Europeana) or the Digital Public Library of America (DPLA) have aggregated large amounts of content (measured in many tens of millions of metadata records) from different formats [1] and a large number of different sources into one format (EDM [2] or DPLA respectively). The handling of multiple formats (metadata schemas) that

are either custom or based on standards but used in a custom manner requires significant effort in order to:

- map them properly to the target schema (e.g. EDM);
- validate the incoming content (structural validation, schema validation, link checking, etc.);
- curate and enrich content with poor quality.

Moreover, efficiency presents another crucial challenge as in many cases millions of records are aggregated periodically and in a short amount of time. It is clear that traditional monolithic approaches would not work against the above challenges whereas a scalable and elastic architecture could stand better chances.

One of the systems used by organizations like Europeana for the aggregation task is the Metadata & Object Repository (MORE)¹ developed by the Digital Curation Unit/IMIS – Athena Research Centre. The large number of different projects, formats and applications, such as CARARE, LoCloud, 3DIcons, ARIADNE², that a single instance of MORE has proved capable of serving is an indication that the system, featuring an innovative architecture to deal with several complexity and efficiency issues, has addressed the challenges of the task in a cost-effective way. This paper presents the architectural modules of MORE aggregator and its functionalities for performing aggregation workflows and curating information in accordance with commonly accepted standards and conventions of the aggregation workflows.

The next section briefly presents the state of the art for the aggregation process and the existing systems, while Sect. 3 presents the MORE architectural modules. In Sect. 4 the information enrichment functionalities are presented and Sect. 5 presents some results from the usage of MORE. Finally Sect. 6 concludes the main results of the paper.

2 Related Work

The traditional approach to aggregate metadata and links to digital resources involves an aggregator [3], which implements a crosswalk to transform original metadata records to records following a common output schema. Three architectural views are relevant to the aggregator system discussed in this paper:

- a scalable, distributed and elastic architecture
- a micro-services oriented architecture
- a pluggable enrichment services architecture

The above distinct architectural features have been explored in the literature with the majority of papers focusing on curation micro-services. Enrichment micro-services can be considered as a category of curation micro-services.

¹ <http://more.dcu.gr/>.

² CARARE: <http://www.carare.eu/>, LoCloud: <http://www.loccloud.eu/>, 3DIcons: <http://3dicons-project.eu/>, ARIADNE: <http://www.ariadne-infrastructure.eu/>.

In [4], news items are automatically enriched with information from Linked Open Data (LOD) Datasets and use an ontology-based browser to demonstrate the advantages of LOD enabled navigation. In [5] the authors use an annotation tool to help users annotate records with information drawn from LOD thesauri. Regarding the micro-services approach, this is demonstrated in [6] where the authors propose and present a curation micro-services infrastructure in order to demonstrate the powerful characteristics and flexibility of such an approach. In [7] a micro-services architecture is presented which focuses on digital curation and preservation. Curation micro-services have also been used for enriching content. In [8] curation micro-services were used in a thematic aggregator to improve the quality and information of content.

MORE upgrades the current state of the art by integrating the traditional aggregation workflow with curation functionalities. It is a metadata aggregator that integrates several services for supporting metadata managers to (a) perform and monitor complex workflows, (b) handle huge volumes of million metadata datasets (c) validate and curate metadata with flexibility, i.e. according to enrichment plans that reflect different needs of metadata curation and (d) publish metadata in various schemas.

3 MORE Architecture

The architecture of MORE is based on established cloud technologies and focuses on three main requirements: (i) *scalability* that refers to both storage and services, (ii) *elasticity*, that refers to handling efficiently high bursts of requests and (iii) *flexibility* that refers to addressing different requirements with the use of services that are deployed in a distributed manner and are applied per case dynamically. All of the above requirements are discussed extensively in the following sections.

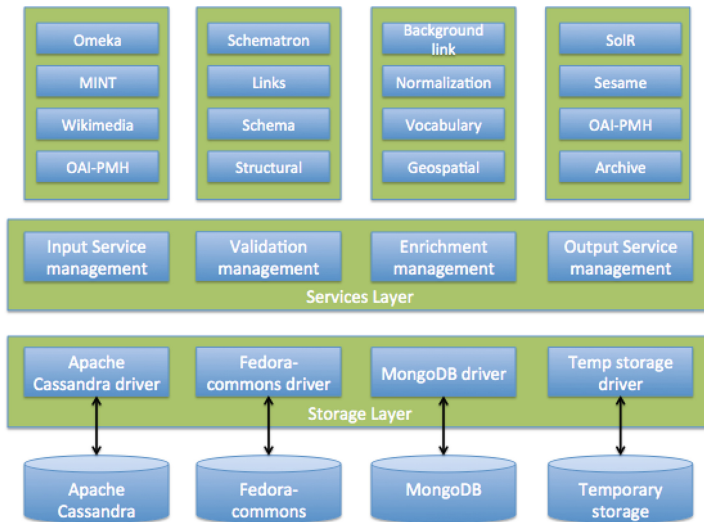


Fig. 1. Overall architecture

The overall architecture of MORE (Fig. 1) includes the following major components: (i) a storage layer, (ii) a services layer that provides a set of core services and (iii) a set of micro-services (Table 1).

Table 1. Curation/Enrichment micro-services integrated in MORE

Geo-normalization	A geo-normalization micro-service.
Geo-coding	A geo-coding micro-service based on Geo-names.
Reverse geo-coding	A reverse geo-coding micro-service based on Geo-names.
Rule based thematic enrichment	A subject collections micro-service that allows the user to create thematic collections of concepts from 27 standard vocabularies (encoded in SKOS).
Automatic thematic enrichment	An automatic vocabulary matching micro-service that identifies SKOS concepts from 27 standard vocabularies (based on title, descriptions and subject related information found in each metadata record).
Wikipedia & DBPedia automatic enrichment	A background links service that automatically identifies Wikipedia and DBPedia entries (based on title, descriptions and subject related information found in each metadata record).
Language identification	A language identification service, which identifies languages based on a title or description employing Apache Tika.
Historic place names enrichment	A historic place names micro-service.
Thesauri mappings	The thesauri mappings service allows loading and managing SKOS concepts mappings from SKOSified subject terms to a target SKOS thesaurus.

The storage layer provides an API that allows attaching virtually any “create, read, update and delete” (CRUD) based storage technology. For each storage technology a driver implementation is required and currently the Apache Cassandra, Fedora-commons and Temporary storage have been implemented.

The services layer consists of a number of core services. Some of them, such as enrichment services, follow the micro-services approach, increasing the flexibility of certain tasks:

- Harvest: harvest content from multiple sources.
- Ingest: ingest content into the appropriate storage.
- Validation: validate content.
- Indexing: index specific elements.
- Quality: measure metadata quality.
- Transform: transform content from one schema to another.

- Enrichment: enrich content using specific enrichment micro-services.
- Publish: publish aggregated content to a specific target.

Inter-service communication involves the communication between micro-services. For that, JMS Queues are used (see section below) which provide elasticity, routing and scalability. Each core service and micro-service consumes a separate queue thus enabling multiple instances to operate without any race conditions. The messages published to queues describe Jobs. A job (e.g. a transformation from one format to another) may contain additional information (e.g. the XSLT document that should be used for the transformation). This information can be part of the message. Core services (e.g. the enrichment management service) usually have to streamline tasks to micro-services and this process involves some kind of business logic (or workflow). This business logic is handled by the core service itself. The generation of jobs among core services (this essentially constitutes the workflow) is handled by a Dispatcher (or workflow management service) which is responsible for interpreting the user's input and enforcing the appropriate workflow.

3.1 Information Organization

Content aggregation is inherently a data driven task. This raises the importance of the content model, which needs to be robust, flexible and most important: domain agnostic. The latter is necessary in order to be able to aggregate information for different domains, schemas and for different purposes.

In order to address the above requirements and be able to cope with multiple users, content providers, metadata schemas and aggregation projects, information is organized in a simple hierarchical structure which can be seen from the data management perspective, as well as from the administrative perspective.

Regarding data management, all incoming information is organized in datasets. Each dataset always falls under an aggregation project. Hence a metadata provider, who participates to one or more aggregation projects, provides one or more datasets in an aggregation project. Each dataset contains one or more items which all belong to one metadata schema. MORE represents dataset items as complex items that comprise versionable datastreams. An item comprises seven datastreams:

1. The administrative metadata stream, which contains information about the provider, package, and general the history of the item.
2. The technical metadata, which contains technical metadata regarding the contents of the item.
3. The native metadata, which contains the source representation (e.g. the native metadata as they were initial harvested).
4. The enriched native metadata, which contains a representation of the enriched version of the native metadata.
5. The target metadata which contains the representation to the target schema
6. The enriched target metadata, which contains a representation of the enriched version of the target metadata.

7. Preservation metadata, which is a log of events of the PREMIS [9] metadata standard.

Regarding the administrative perspective, users are divided into four distinct roles:

1. Administrators that can setup the aggregation flows, define schemas, and system parameters.
2. Project managers that can have project scope access (e.g. see all information aggregated from different providers into a project).
3. Content providers that can initiate harvests and run the entire aggregation flow for their own organizations horizontally across projects.
4. Developers that can use MORE's RESTful API and deploy enrichment micro-services.

Each user can participate in different projects and assume different roles in each one.

3.2 Aggregation Workflows

One of the most important design considerations of MORE concerns workflows that can be adapted to particular aggregation scenarios. This is essential in order to be able to cope with the diverse needs that are found in large aggregation projects. The main aggregation workflow can be seen in Fig. 2; as indicated in the figure, some stages incorporate validation and indexing services.

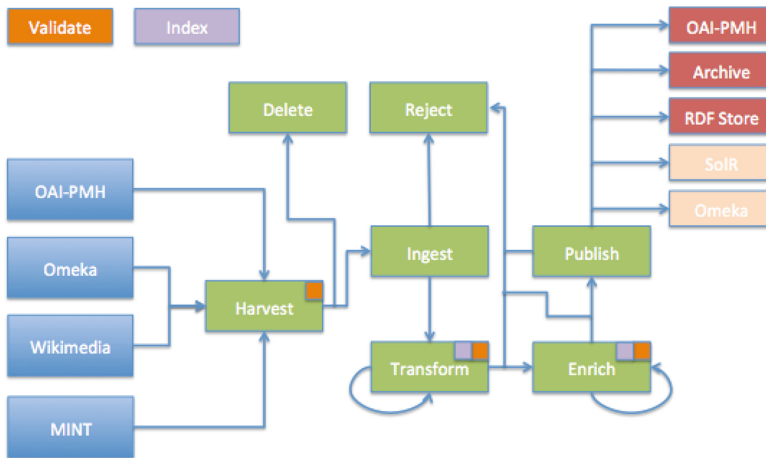


Fig. 2. Aggregation workflow

The user starts by initiating a harvest from one of the available input sources e.g. an OAI-PMH server or other intermediate aggregators, such as MINT, Omeka [10], etc. After a harvest is completed, the incoming dataset is validated and then ingested into the system. After ingest it is transformed into a common schema (e.g. EDM) and, if needed, it can be enriched using various enrichment micro-services (see next section).

After each transformation and enrichment operation, the dataset is validated and indexed. Finally, if the dataset is accepted for publication, the publishing service can publish it to one or more targets.

The overall content aggregation process involves a number of systems besides the aggregator itself, such as the content providers' native repositories and the publish targets. In order to provide interoperability so that these systems can be directly linked to the aggregation infrastructure, a RESTful API is provided alongside the human interface. The human interface is provided through a responsive Bootstrap based template that provides an improved user experience and intuitiveness.

3.3 Validation

When aggregating large amounts of content, validation and metadata quality measurement is essential in order to allow the aggregation manager to make informed decisions and ensure that the publication will be accepted. The validation of datasets in MORE is handled by the validation service that comprises four distinct micro-services:

- Schema validation (based on a given XSD)
- Structural checking of an XML record
- Link checking (broken links)
- Schematron rule based validation

The above four types of validation services are streamlined into validation schemes allowing for different schemes to be configured for different projects/sources or providers. This enables skipping heavy duty tasks such as link checking for very big packages that come from reliable sources.

3.4 Metadata Quality

Alongside with the validation results, a metadata quality service evaluates the quality of a metadata record (after each transformation or enrichment action) and produces quality related information that currently includes:

- Completeness per element for each schema
- Completeness per element set (mandatory and recommended element sets are used)

In order for the aggregation manager to make informed decisions on whether and how to enrich an information package, indexes (except metadata completeness) concerning spatial, thematic, temporal and rights information, are computed and presented by the system. These indexes are configured per schema, similarly to the metadata completeness index. In the case of spatial information indexing, a small map widget is used to project the data directly on the map thus offering a better, more intuitive user experience.

In the cases of thematic, temporal and rights information, a small widget presents the entries for each index on a list view and the number of their occurrences in the metadata records (items) per dataset.

3.5 Publish Targets

A dataset can be published after it has been transformed into a predefined metadata format, according to the specific project target schema, and it has passed validation and has been enriched. Traditionally in most aggregator infrastructures, publication either means exposing the published items through an OAI-PMH provider or pushing them to a SolR index server. In order to provide more flexibility MORE's publishing service supports a number of different publish targets, which can be extended to include more publish targets as they also follow a micro-service architecture. Currently, the publish service supports the following targets:

- Internal OAI-PMH repository (published under a specific Set per provider and project)
- Sesame RDF store (if the target schema is RDF formatted)
- OpenLink Virtuoso RDF store
- Elastic Search index server
- Archive (dump in tar.gz format for all published items)

Furthermore, it is possible for a project to have multiple targets, for example to publish all content through an OAI-PMH provider and also download them through an XML archive. The administrator can define these targets for each project separately and the aggregation manager can choose in which of the available targets to publish each dataset. This implies that the same content can be published simultaneously in multiple targets.

Some publish targets require the content to be in a format different than that of the project's target schema. For example, when aggregating content in XML and the goal is to transform into a common XML schema (e.g. EDM) the transformed EDM representations of each item can be directly published to the internal OAI-PMH provider or to an archive, but they cannot be published to an RDF store as this requires an RDF format. Similarly, Elastic Search accepts JSON format and, although it can automatically convert from XML to JSON in most cases it is more practical and efficient to encode information in specific JSON format.

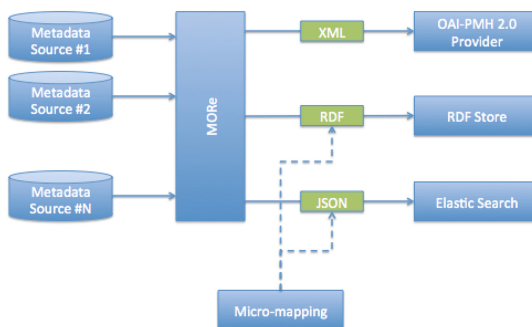


Fig. 3. Publishing to multiple targets using micro-mappings

In order to enable publishing to multiple targets simultaneously, a micro-mapping mechanism is employed which allows mapping directly and on the fly during the publication process to the target's desired output (Fig. 3). Micro-mappings are realized using XSLT transformations and the parameters are defined per project and partner. For example, in the case of Elastic Search, the server URL, credentials and index name are provided along the micro-mapping XSLT document.

3.6 Elasticity and Scalability

Elasticity is a characteristic very common to big data architectures (along with scalability) as bursts of requests that need not to be processed in real time can occur. An elastic architecture can ensure that all requests can be received, acknowledged and dispatched when a worker node becomes available. MORE, provides an elastic architecture which offers all of the above and is based on message queues.

Scalability is a critical characteristic of an aggregation process as it is resource consuming both in terms of storage and processing power. These two aspects are addressed in MORE in the following two ways:

- At the storage level by using a cloud based storage (like Apache Cassandra) that can scale out using a clustered architecture.
- At the data processing level by adopting scalable services architecture that allows services to scale out in a clustered environment (such as Apache Storm).

4 Curation

One of the most important aspects of MORE is that it is curation aware. This means that apart from simple XSLT based transformations from a native schema to a target schema, MORE employs a number of curation/enrichment micro-services that perform various curation/enrichment actions on the metadata. Examples of such micro-services that have been integrated/developed are listed in Table 1.

It is apparent that the above micro-services are heterogeneous from several aspects:

- they have been developed using different programming languages and frameworks;
- they require and produce different information (e.g. spatial coordinates, links, language codes, etc);
- they are encoded in different ways (e.g. json, xml, etc);

Some of them are self-sustained, some others, such as the Geo-names gazetteer rely on external databases and services.

4.1 Micro-services de-Coupling/Micro-schemas

The heterogeneity of micro-services presents a challenge for the system to be extensible with a minimum amount of effort and to take advantage of the richness of innovative services that are freely available. To this end, two main methodologies/technologies were employed:

- (a) a service oriented architecture (SOA) relying on HTTP REST to facilitate communication; and
- (b) an abstraction layer that de-couples the logic of the enrichment services from that of the aggregator.

The communication through REST enables simple and efficient remote invocation of micro-services while retaining the ability to scale them.

The abstraction layer enables to dynamically map parts of the target schema (e.g. part of metadata that provide only geographical data and coordinates) to the inputs of each enrichment service (e.g. Geo-coding micro-service). This technique employs *micro-schemas* to perform the mappings dynamically and thus enabling MORE to apply the same enrichment micro-service to multiple target schemas without having to adapt/code. This technique is illustrated in Fig. 4.

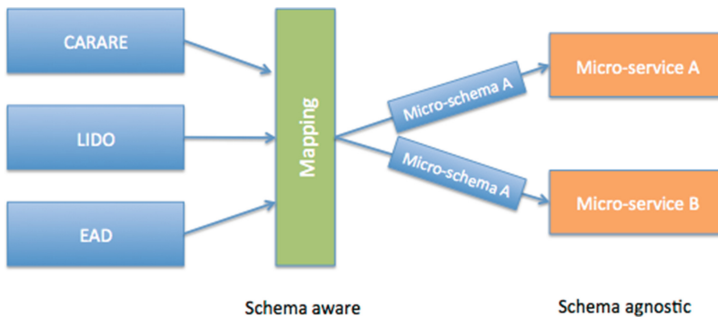


Fig. 4. Use of micro-schemas in enrichment micro-services

4.2 Streamlining Enrichment Micro-services: Enrichment Planning

In order for these services to be integrated and put into effective use in the aggregation workflow, service orchestration is employed through the *Enrichment Management Service*. This service is responsible for enriching a dataset through streamlining the execution of enrichment micro-services. The tasks it performs are:

- Iterating through all the valid items of a dataset.
- Executing a set of enrichment micro-services (called *Enrichment Plan* – see next section) for each one of the valid items by feeding the output of an enrichment micro-service to the input of the next enrichment micro-service.
- Handling errors in the enrichment process.
- Applying specific project/provider or run-time parameters to each enrichment service.
- Dispatching and monitoring the health of each enrichment micro-service.
- Compiling a report on the output of the enrichment.

All of the above tasks are provided through an API so that configuration and new micro-services integration tasks are provided easily.

Enrichment planning is an important and innovative feature of MORE as it allows each content provider or aggregation manager to easily create complex and powerful enrichment workflows by combining simple enrichment micro-services through an intuitive graphical interface.

Each enrichment plan operates on a single metadata schema and apart from streamlining the execution of enrichment micro-services, it defines configuration parameters for each one of them (if and when available). After the execution of an enrichment plan, a report is compiled and presented to the user so that he/she can see which items were enriched by which service etc.

5 Experimental Results

MORE manages huge volumes of data and provides a diversity of services to its users. Therefore it is obvious that the system performance depends on two parameters: (a) the number of micro-services that are processing data concurrently and (b) the number of concurrent users that call and apply micro-services on datasets. Regarding the second parameter it was observed that when the number of workers increase from 1 to 8, the response time decreases by 63.47 %.

Therefore, in order to evaluate the performance of the micro-service architecture, experimental results have been carried out using a fixed number of datasets coming from the CARARE project. The datasets included 1.3 million records approximately encoded in the CARARE schema [1]. The same datasets were used in the different experiments as well as the same environment (debian linux server).

The experiments investigated two scenarios: in the first scenario, the core service package contains all micro-services whereas in the second one the micro-services are completely distributed (running in different machines on the same subnet). It is clear

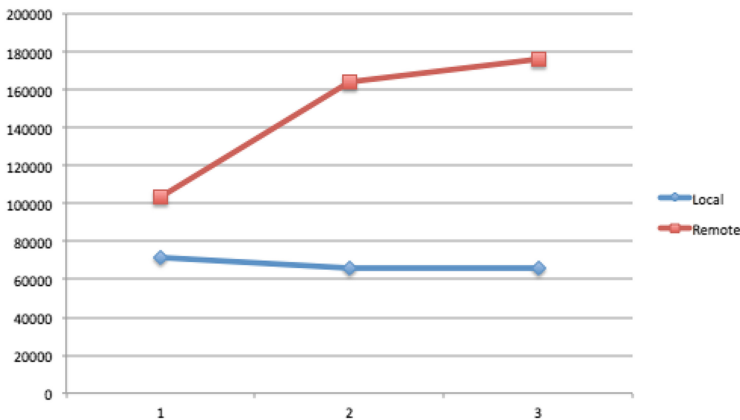


Fig. 5. Execution time in ms (vertical) versus the number of enrichment micro-services executed (horizontal) for an enrichment task. The blue line (circle) refers to a scenario where all the micro-services are packaged in the same worker. The red line (square) refers to a scenario where the micro-services are distributed. (Color figure online)

that when the enrichment plan becomes more complex and contains more micro-services the response time greatly decreases in the first case. In the example shown in Fig. 5, for 3 micro-services this increase reaches 62.42 %.

6 Conclusions

In this paper, the innovative architecture of the MORE metadata aggregator has been presented. MORE addresses the complexities found in metadata aggregation tasks through a micro-service oriented architecture that provides elasticity, flexibility and scalability. MORE has been effectively used to aggregated millions of records in various projects in different domains involving different formats and targets. It is accessible both as a Web application and through a RESTful API and it allows developers to extend it with curation/enrichment micro-services easily. The performance evaluation experimental results were encouraging since they indicate that MORE is a stable system even when it manages huge volumes of datasets.

References

1. Papatheodorou, C., Dallas, C., Ertmann-Christiansen, C., Fernie, K., Gavrilis, D., Masci, M.E., Constantopoulos, P., Angelis, S.: A new architecture and approach to asset representation for europeana aggregation: the CARARE way. In: García-Barriocanal, E., Cebeci, Z., Okur, M.C., Öztürk, A. (eds.) MTSR 2011. CCIS, vol. 240, pp. 412–423. Springer, Heidelberg (2011). doi:[10.1007/978-3-642-24731-6_41](https://doi.org/10.1007/978-3-642-24731-6_41)
2. Isaac, A.: Europeana data model primer. Europeana Project (2011)
3. Reis, D., Freire, N., Manguinhas, H., Pedrosa, G.: REPOX – a framework for metadata interchange. In: Agosti, M., Borbinha, J., Kapidakis, S., Papatheodorou, C., Tsakonas, G. (eds.) ECDL 2009. LNCS, vol. 5714, pp. 479–480. Springer, Heidelberg (2009). doi:[10.1007/978-3-642-04346-8_65](https://doi.org/10.1007/978-3-642-04346-8_65)
4. Mannens, E., Troncy, R., Braeckman, K., Van Deursen, D., Van Lancker, W., De Sutter, R., Van de Walle, R.: Automatic metadata enrichment in news production. In: 10th Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2009), pp. 61–64. IEEE Press (2009)
5. Rainer, S., Haslhofer, B., Jung, J.: Annotations tags and linked data. Metadata enrichment in online map collections through Volunteer-Contributed Information. *e-Perimtron* **6**, 129–137 (2011)
6. Clair, K.: Metadata for a micro-services-based digital curation system. In: International Conference on Dublin Core and Metadata Applications, pp. 58–62 (2011)
7. Abrams, S., Kunze, J., Loy, D.: An emergent micro-services approach to digital curation infrastructure. *Int. J. Digit. Curation* **5**, 172–186 (2010)
8. Gavrilis, D., Dallas, C., Angelis, S.: A curation-oriented thematic aggregator. In: Aalberg, T., Papatheodorou, C., Dobрева, M., Tsakonas, G., Farrugia, C.J. (eds.) TPD 2013. LNCS, vol. 8092, pp. 132–137. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-40501-3_13](https://doi.org/10.1007/978-3-642-40501-3_13)
9. Gavrilis, D., Angelis, C., Papatheodorou, C.: MOPSEUS: a digital repository system with semantically enhanced preservation services. In: 7th International Conference on Preservation of Digital Objects (iPRES 2010), pp. 135–143 (2010)
10. Kucsma, J., Reiss, K., Sidman, A.: Using Omeka to build digital collections: the METRO case study. *D-Lib Mag.* **16**(3–4) (2010)

Metadata and Semantics Research

10th International Conference, MTSR 2016, Göttingen,

Germany, November 22-25, 2016, Proceedings

Garoufallou, E.; Subirats Coll, I.; Stellato, A.; Greenberg,
J. (Eds.)

2016, XX, 378 p. 110 illus., Softcover

ISBN: 978-3-319-49156-1