

**Grant Agreement 297292**

***EUROPEANA INSIDE***

**Europeana Connection Kit –  
Technical Documentation**

<b>Deliverable number</b>	D5.3
<b>Dissemination level</b>	Public
<b>Delivery date</b>	September 2014
<b>Status</b>	V0.9
<b>Author(s)</b>	K-INT



This project is funded under the  
*ICT Policy Support Programme* part of the  
*Competitiveness and Innovation Framework Programme*.

## Revision History

Revision	Date	Author	Organisation	Description
V0.1	2014-09-29	Neil Smith	K-INT	First Draft
V0.2	2014-09-30	Naeem Muhammad	KUL	Additional mapping information included
V0.3	2014-10-03	Eva Van Passel	IBBT	First Review
V1.0	2014-10-06	Laura Miles	CT	Final Review

### Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

## Contents

<b>1. INTRODUCTION .....</b>	<b>4</b>
<b>2. ECK MODULES .....</b>	<b>5</b>
2.1. ECK Core.....	6
2.2. Metadata Definition .....	7
2.3. Data Mapping/ Transformation .....	8
2.4. Preview .....	9
2.5. PID Generation .....	10
2.6. Statistics .....	11
2.7. Validation .....	12
2.8. Set Manager .....	13
<b>ANNEX – PARTNERS PROVIDING ECK IMPLEMENTATION .....</b>	<b>14</b>

## 1. Introduction

This document provides details of the location of technical documentation relating to the software components which were developed during the Europeana Inside project, collectively known as the Europeana Connection Kit (ECK). The nature of this deliverable was redefined as 'Other' in an amendment to the Description of Work in order to reflect the preference of all technical partners for online delivery of documentation alongside the online delivery of the reference implementations of the software components (described in D5.1).

As described in D5.1, the software components are divided into 2 main categories. This document focuses on the common or shared modules which are used to some extent in all the implementations of the ECK. In addition to these components, each individual technical partner with a Collections Management System or similar system also embedded functionality within that system. Those partners are listed in the Annex to this report. Technical documentation for components embedded in third party systems is an integral part of the technical documentation of those systems. For commercial reasons, product technical documentation relating to third party products is not reproduced here.

The ECK modules have been developed by a number of partners, each of which has adopted its own approach to online documentation (for example some documentation is available as HTML whilst others are in PDF or MS Word format). No attempt has been made to standardise the format and delivery mechanism as the requirements of users of each module can differ. The test of whether the documentation is acceptable and sufficient was the feedback gained from other technical partners during the course of the project.

In terms of the sustainability of this documentation, we refer to the forthcoming Europeana Inside Forward Plan, D 5.4. The section on "Sustaining the Europeana Connection Kit" that will be elaborated in this Forward Plan is also applicable with regards to the documentation discussed in this deliverable.

## 2. ECK Modules

The table below shows the ECK modules and describes the functionality they deliver.

<b>Module</b>	<b>Functional Description</b>
CMS: ECK supporting functionality	All functionality that must be provided within or alongside a CMS in order to meet the functional requirements set out in D2.4, including Data Push and/or OAI-PMH repository and content re-ingestion.
ECK Core	The gateway to the various modules, so supports all interfaces that all the other modules support.
Metadata Definition	Provides definitions for metadata elements within profiles and error messages in the different languages
Data Mapping / Transformation	Converts between formats, e.g. Native format to LIDO or LIDO to EDM
Preview	Provides a preview of how the record will be rendered in the Europeana user interface
PID Generation	Provides a persistent id for the record
Set Manager	Manages the sets and records
Statistics	Provides statistics about the performance of modules
Validation	Validates the record to ensure it conforms to a provided profile (of either LIDO or EDM).
Aggregator: ECK supporting functionality	Accepts data via data push or OAI-PMH, supplies data to Europeana via OAI-PMH and provides management information and enriched records for re-ingestion

**Table 1: Summary of ECK functionality**

As reported in section 1, CMS:ECK supporting functionality (i.e. the software components that are specific to an individual partner’s implementation of the ECK) is not covered in this deliverable.

## 2.1. ECK Core

The reference version of the ECK core module is installed at <http://euinside.k-int.com/ECKCore2/help/core>.

The standard help page for ECK Core was included in D5.1. Help pages exist for each interface. The following screen shot shows the help page for the aggregator interface:

### ECKCore - Aggregator

#### Description

The Aggregator interface provides a way of talking to the aggregators through the ECKCore.

#### Invocation

The url for the interface takes the form /<aggregator>/<action>/<urlParameter1>/<urlParameter2>?parameters Where:

URL Part	Description
aggregator	The code for the aggregator that you want to retrieve the statistics from (eg. DarkAggregator, CultureGridTest, CultureGridLive, Europeana, SetManager, ...)
action	The action to be performed against the aggregator
urlParameter1	The purpose of this parameter varies depending on the action being performed, see the action for more details
urlParameter2	The purpose of this parameter varies depending on the action being performed, see the action for more details

The possible actions are:

- [enrichmentRecord](#)
- [statistics](#)

Note: Not all aggregators support all actions, when the aggregator does not support the action or support has not been added for an aggregator the http status 400 will be returned with an explanatory message

#### Parameters

There is one general parameter that the ECKCore intercepts and takes notice off and that is rawRequired, if this is set to yes then the raw data returned from the aggregator will be passed back to the caller, if it is not set or not set to yes then we will not map the data to the generic format unless xml is being returned from the server.

#### Response

The data for all actions is returned as json

#### Testing

A simple test form is available from each of the action pages

## 2.2. Metadata Definition

The reference implementation of the metadata definition module is installed at <http://euinside.k-int.com/ECKCore2/Definition>.

The standard help page for this module was reproduced in D5.1. For each of the three available operations more detailed on-line documentation is available. The following screen shot shows the detailed help screen for the Error Definitions action:

### Error Definitions

#### Invocation

The following are the possible ways to retrieve the definition of error message(s)

##### **/errors**

Retrieves the full list of errors in the default language

##### **/errors/<language>**

Retrieves the full list of errors in the specified language

##### **/errors/<language>/<error code>**

Retrieves the definition of the specified error in the requested language

If the error definition is not available in the specified language, then it will be returned in the default language

The content is returned as json

#### Response when /errors/en/error1 is specified as the url

```
{
  "error": "error1",
  "definition": "The first error code"
}
```

#### Response /errors or /errors/en is specified as the url

```
[
  {
    "error": "error1",
    "definition": "The first error code"
  },
  {
    "error": "error.with.dots",
    "definition": "An error code with dots in it"
  }
]
```

### 2.3. Data Mapping/ Transformation

The reference implementation of the data mapping and transformation modules are installed at:

Mapping: [http://www.heron-net.be/einside\\_test/dmt.php](http://www.heron-net.be/einside_test/dmt.php)

Transformation: <http://services.libis.be/euInside/dmt.php>

Documentation for the data mapping and transformation services is available as a MS Word document, downloadable from <http://euinside.k-int.com/ECKCore/static/docs/ManualDataMappingTransformationService.doc>.

The introductory section of this document is reproduced below:

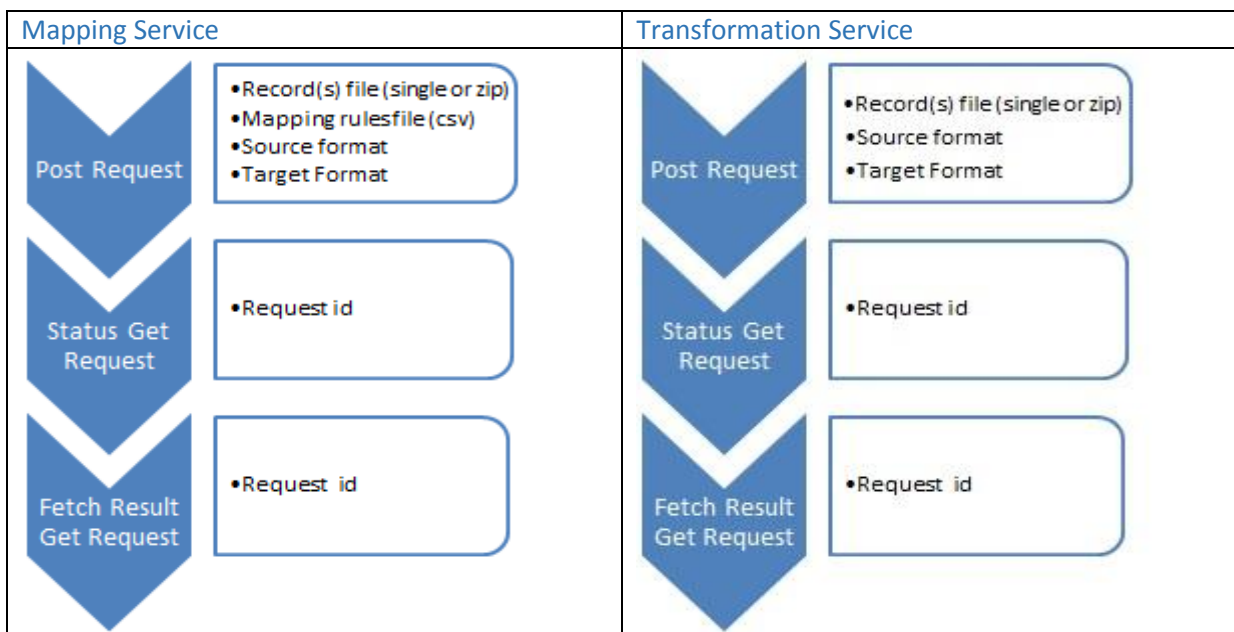
#### Introduction

This document describe the use of REST api developed to map and transform records from one format to another. The services of this api can be divide into two categories: Mapping Service and Transformation Service.

The mapping service can be used to convert XML to another XML. Currently we have implemented LIDO to EDM and MARC to EDM. The mapping is defined in a text file with the specified mapping rules. The mapping rules consists of an action, input and output. Sometimes there is also a fixed value. The rules will be applied during the transformation.

The mapping service is only necessary if you want to make your own mapping. If you use the mapping rules specified by Linked Heritage project, then you don't need this mapping service. In this case you need to use the transformation service instead of the mapping service.

Following diagram shows the workflow of both mapping and transformation service. It can be noticed that no mapping rules file is needed for transformation service.





## 2.4. Preview

The reference implementation of the preview module is installed at <http://euinside.asp.monguz.hu/eck-preview-servlet/index.html>.

Documentation for the Preview module is available as a PDF document, downloadable from <http://euinside.k-int.com/ECKCore/static/docs/preview-rest-api-MON.pdf>.

The preview generation section of the documentation is reproduced below:

### Preview Generation

Responsible for preview generation for EDM records. Clients are able to:

- Choose a template and apply it to a record supplied. Results vary according to mime type requested. The following mime types are supported:
  - text/html: Returns the preview in XHTML format.
  - application/xhtml+xml: See text/html.
  - application/zip: Returns preview in a zipped bundle.
- Upload a set of records for batch Preview.
- Retrieve the result of a batch preview as a zipped bundle.
- List active sets.

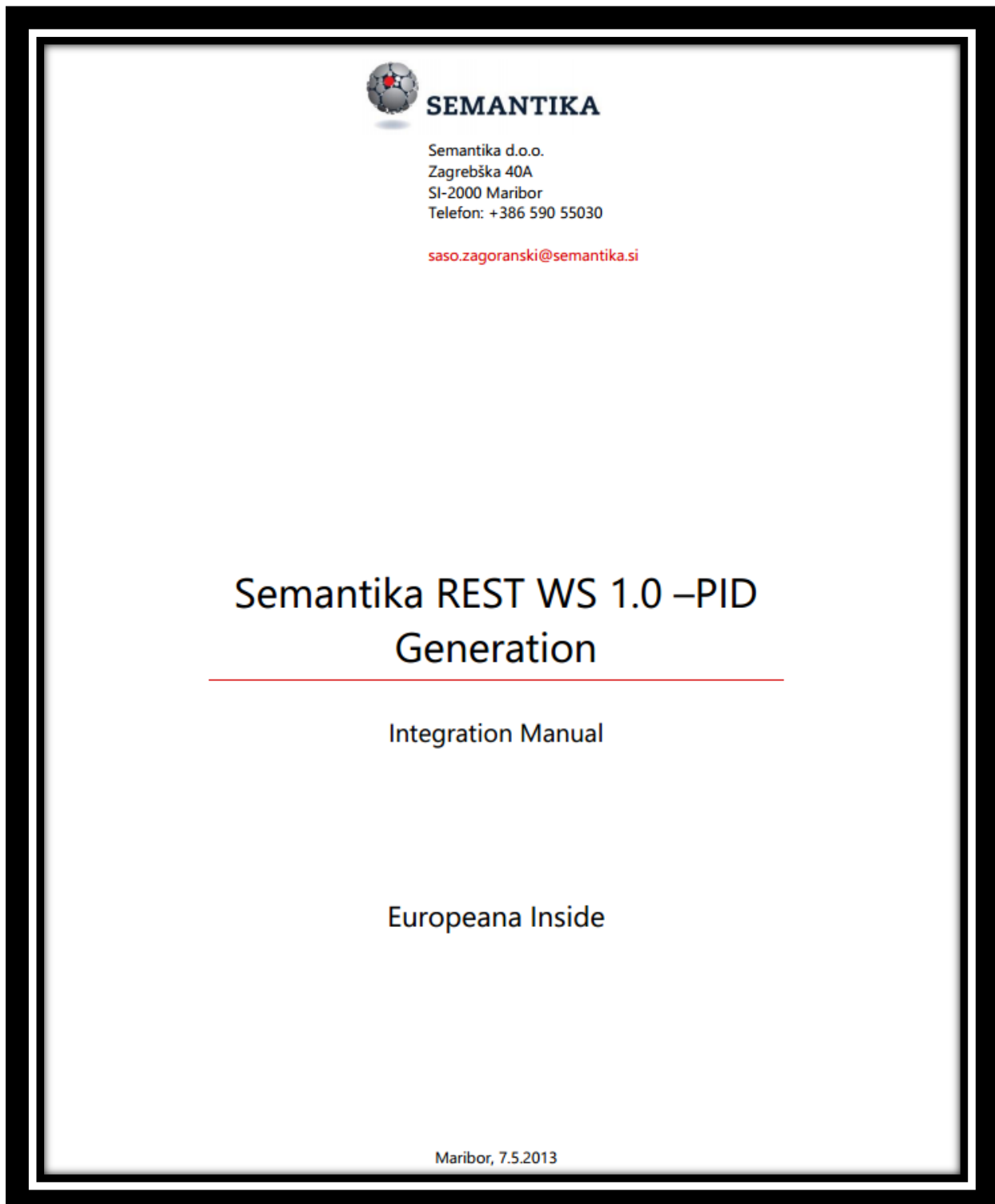
URL (pattern)	Request body	HTTP Method	Function	Response body
/Preview/{provider}/single/preview/{name}	Record in EDM (or other supported) format	POST	Generates a preview of the record using the template specified by name.	Preview as HTML or as a zipped bundle.
/Preview/{provider}/sets	Empty	GET	List available sets, with processing status	Sets as XML or JSON as determined by mime type
/Preview/{provider}/{set name}/preview/{name}	Record in EDM (or other supported) format	POST	Generates a preview of the record using the template specified by name (batch mode).	HTTP 202, empty body, HTTP 409 if set already exists for current provider
/Preview/{provider}/{set name}/result	Empty	GET	Retrieves the result of a batch.	Results as a zip, HTTP 200 if results are available, HTTP 102 if processing is still ongoing

## 2.5. PID Generation

The reference implementation of the PID generation module is installed at <http://euinside.semantika.si/pid>.

Documentation is available as a PDF document, downloadable from [http://euinside.k-int.com/ECKCore/static/docs/Semantika\\_EU\\_Inside\\_PID\\_Generation\\_WS.pdf](http://euinside.k-int.com/ECKCore/static/docs/Semantika_EU_Inside_PID_Generation_WS.pdf).

The cover page of the document is reproduced below:



## 2.6. Statistics

The reference implementation of the statistics definition module is installed at <http://euinside.k-int.com/ECKCore2/Statistics/>.

The standard help page for this module was reproduced in D3.1. That page links to more detailed help screens for each of the 2 available actions (query and update). The help screen for the Update action is reproduced below:

### Statistics - Update

#### Description

The update action allows you to add statistics for a module

#### Parameters

Parameters	Description
dateTime	Date and time of when the processing started, format: ISO8601, RFC3339
duration	The length of time in milliseconds that it took to process the records
itemsProcessed	The number of items that were processed
numberFailed	The number of items that failed to be processed
numberSuccessful	The number of items that were successfully processed

Note: If two off the three parameters itemsProcessed, numberSuccessful and numberFailed are supplied and the third is missing then the third one will be calculated, if we do not have at least two of these parameters then an error will be generated, the dateTime and duration parameters are mandatory and an error will be generated if either of these are missing.

#### Responses

If successful the response will return a http status code 202, if there is an error then a http status code 400 will be returned

#### Testing

In order to allow simple testing of the ECK Statistic interface, a test form is available [here](#) which performs actions against the default modules default group.

## 2.7. Validation

There are two reference implementations of the validation module. These are installed at <http://euinside.asp.monguz.hu/eck-validation-servlet/index.html> and <http://euinside.semantika.si/Validation>

Both validation services support the same API calls and therefore share the same online documentation, available as a PDF document downloadable from <http://euinside.k-int.com/ECKCore/static/docs/validation-rest-api-common.pdf>.

The screenshot below illustrates the documentation available for the Record Validation action:

### Record validation

Responsible for validating EDM records. Clients are able to:

- Choose a validation profile, and validate a record against it immediately.
  - A record can be in any supported format, default is LIDO.
- Validate a record set.
- Retrieve a validation result for a given set.
- List active sets.

URL (pattern)	Request body	HTTP Method	Function	Response body
/Validation/{provider}/single/validate/{name}	Record in LIDO (or other) format	POST	Validate a single record	Validation result in XML format, HTTP 200 if validation successful, 412 if there were errors.
/Validation/{provider}/sets	empty	GET	List available sets, with processing status	Sets as XML or JSON as determined by mime type
/Validation/{provider}/{set number}/validate/{name}	Records in LIDO format as a ZIP file (application/zip)	POST	Validate the record using the selected profile, batch mode	HTTP 202, empty body, HTTP 409 if set already exists for current provider
/Validation/{provider}/{set number}/result	empty	GET	Retrieve batch validation results	Results in XML format or as a zip (determined by header mime type), HTTP 200 if results are available, HTTP 102 if processing is still ongoing.

## 2.8. Set Manager

The reference implementation of the Set Manager module is installed at <http://euinside.k-int.com/ECKCore2/SetManager/>.

The standard help page for this module was reproduced in D3.1. That page links to more detailed help screens for the each of the ten available actions. The help screen for the Edit action is reproduced below:

### Set Management - Edit

#### Description

The edit action allows various details about the provider set to be modified, if a parameter is not supplied or is empty then the field is not modified, it returns the current values of the fields, except the password field or an error message if there was a problem saving the changes (eg. too much data for a field)

#### Parameters

The edit action accepts the following parameters

Parameter	Meaning
europeanaId	The id that europeana knows this provider as
providerDescription	The description to be given to the provider
collectionId	The collection identifier to be used for the push operation
password	The password required for the push operation
providerId	The provider identifier to be used for the push operation
username	The username required for the push operation
setDescription	The description to be used for the set
swordURL	The base sword URL to be used for the push operation
pushFormat	The format of the data to push LIDO or EDM (defaults to LIDO)

#### Response

The response is in json and returns the current values of the fields, except the password field

```
{
  "europeanaId": "20223",
  "provider": "chas",
  "providerDescription": "This be my provider not yours",
  "pushCollection": "Chas1",
  "pushProviderIdentifier": "chas99",
  "pushUsername": "username88",
  "set": "first",
  "setDescription": "This is the first set",
  "swordURL": "swordurl Possibly"
}
```

#### Testing

In order to allow simple testing of the ECK Set Manager interfaces, etc. a test form is available [here](#) which actions data for the default provider using the default set.

## **Annex – Partners providing ECK implementation**

Adlib Information Systems

iMinds

KE Software

Knowledge Integration

Libis (KU Leuven)

Mobydoc

Monguz

Postscriptum

Semantika

Skinsoft

System Simulation

Zetcom