

HIGH INTEGRATION OF RESEARCH MONOGRAPHS IN THE EUROPEAN OPEN SCIENCE INFRASTRUCTURE

Deliverable D6.1

Metrics services specification

Grant Agreement number	: 731102
Project acronym	: HIRMEOS
Project title	: High Integration of Research Monographs in the European Open Science infrastructure
Funding Scheme	: EINFRA-22-2016
Project's coordinator Organization	: CLEO-CNRS
E-mail address	: pierre.mounier@openedition.org
Website	: http://www.hirmeos.eu
WP and tasks contributing	: WP6
WP leader	: Ubiquity Press
Dissemination level	: CO
Due date	: 31/08/2017
Delivery date	: 31/08/2017



The project has received funding from European Union's Horizon 2020 research and innovation programme under grant agreement 731102

Contents

I.	Overview	4
II.	Metrics API Standards	4
	A. Geographical Data	4
	B. Chronological Data	5
	C. Response Standards	5
III.	Altmetrics Service	5
	A. Background	5
	B. Workflow	5
	C. DOI Scraping	6
	D. Multiple Resources and DOIs	7
	E. Solving edge cases	8
	F. Results	8
	a) By source	8
	b) By time	9
	c) By source	10
	d) By scrape ID	11
IV.	Citations Service	11
V.	OAMetrics Service	12
	A. OAMetrics Open Standard	12
	1. Measures	13
	B. Statistics Collection Agent	14
	1. Drivers	14
	2. Identifier Translation Service Back End	15
	3. Identifier Translation Service Front End	15
	C. Reader Analytics Service	18
	1. Database Structure	18
	2. POST Requests	19

3. GET Requests	19
4. DELETE Requests	20
Appendices	20
A. Reader Analytics Database Entity Relationship Diagram	20
B. Reader Analytics Example GET Requests	22
C. CrossRef XML response	25

I. Overview

Ubiquity Press (UP) is developing an altmetrics service (T6.2) and a citations service (T6.3). The software that collects the altmetrics and citations will run in their servers, and a public API will be provided (T6.1). Publishers will be able to register DOIs, and then perform requests to the API to obtain the metrics per DOI.

Open Book Publishers (OBP) will provide partners with: software to collect metrics from the different platforms (e.g. Google Books, Open Edition, etc.), an Open Standard format to enable the upload of statistics from other platforms (T6.4), and an API to enable metrics upload/query (T6.1).

The API will be used to query three types of metrics: altmetrics, citations, and readership; which must be queried separately specifying the type of metric as part of the URL (e.g., /altmetrics/).

UP will also provide a JavaScript widget that displays all of the metrics collected for a specific publication (T6.5).

II. Metrics API Standards

The common metrics API will provide access to all metrics collected within the project, hence a series of standards have been agreed in order to facilitate coordination between the parties involved in the development of itself.

A. Geographical Data

Some of the readership metrics include geographical information, thus a consistent way of recording this data is needed. For this purpose, the standard specification of choice has been ISO 3166-1 alpha-2¹. Although not included in a standard specification, there seems to be consensus in using the following codes to categorise continents:

- AF: Africa
- AN: Antarctica
- AS: Asia
- EU: Europe
- NA: North America

¹ See: <https://www.iso.org/iso-3166-country-codes.html>

- OC: Oceania
- SA: South America

B. Chronological Data

The standard "Complete date plus hours, minutes and seconds" as specified by W3C², e.g. 1997-07-16T19:20:30+01:00 will be used.

C. Response Standards

Along with the actual result of the query, the response should also incorporate some sort of **status** in order to let the client know whether the query was successful or not. These statuses could either be codes (e.g., 200), strings (e.g., success), or both (e.g., {"code": 200, "status": "success"}).

III. Altmetrics Service

A. Background

CrossRef offers a similar service called Event Data³, which is focused on finding references to a specific DOI in the web, more than the URL of the document.

B. Workflow

- The publisher registers an account on the proposed Altmetrics service.
- After logging in, the publisher loads a CSV containing the following columns
 - URL of the published document (article, book)
 - DOI of the document; used as primary key
- **URL scraping:** the service launches a scraper, which will query the appropriate public API for the following social network: Twitter, Facebook, Wikipedia; the scraper will (try to) match each of the URLs the publisher has uploaded.
 - If a match is found, the scraper will download and store the content of the match (tweet, facebook post, etc.) along with its unique identifier in the

² See: <https://www.w3.org/TR/NOTE-datetime>

³ See: <https://www.crossref.org/services/event-data/>

social network it belongs to. The match will also be associated with the DOI.

- **DOI scraping:** the scraper will also call the CrossRef API to collect information about the DOI; the "crossref_type" is particularly relevant, since it will provide information about the type of document (article, book, etc.)
- A public API will expose the DOIs and their matches.
- A JavaScript widget will be made available to the publisher; once loaded, it will query the metrics API for one or more DOIs, and receive a summary of the social shares of each.

C. DOI Scraping

CrossRef is a valuable source of information regarding DOIs. An example response for a query on a specific DOI can be seen in Appendix C. Some of the fields which should be imported in the Altmetrics service as part of the scraping are:

- doi type
- publication date
- cited by

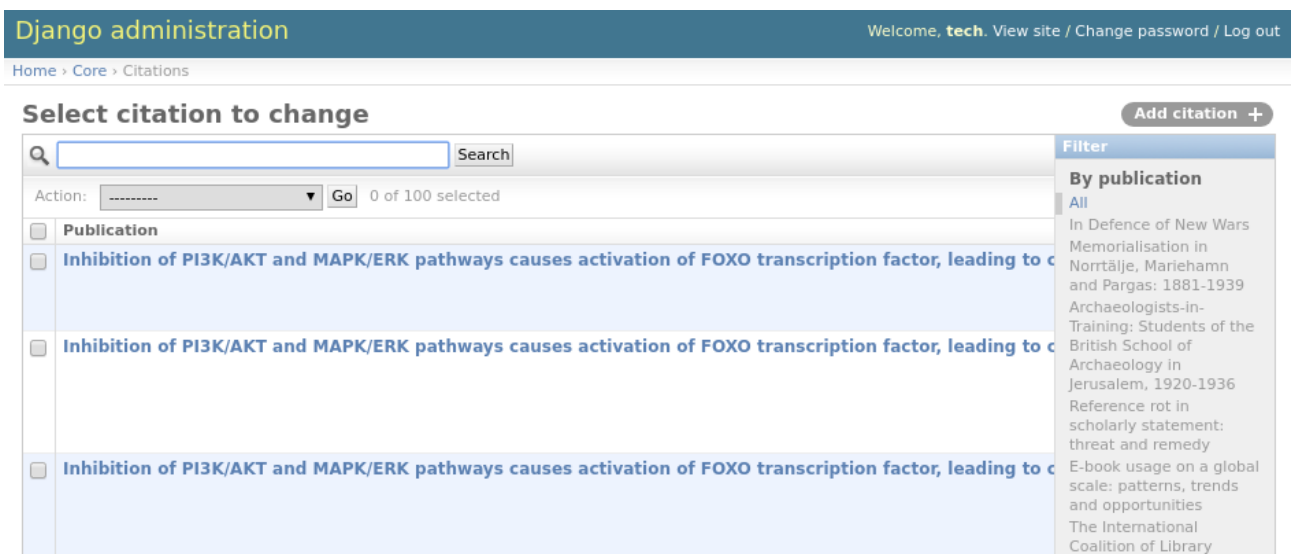


Fig. 1: Example of CrossRef citations collection for a DOI.

Django administration Welcome, tech. View site / Change password / Log out

Home > Core > Citations > Inhibition of PI3K/AKT and MAPK/ERK pathways causes activation of FOXO transcription factor, leading to cell cycle arrest and ...

Change citation History

Publication:
 Inhibition of PI3K/AKT and MAPK/ERK pathways causes activation of FOXO transcription factor, leading to cell cycle arrest and apoptosis in pancreatic cancer

Doi:

Journal title:

Article title:

Year:

Volume:

Issue:

[✖ Delete](#)

Fig. 2: Detail of the collected information from CrossRef.

D. Multiple Resources and DOIs

Multiple relationships between a document and a DOI are possible:

1. 1 DOI for each document
2. Multiple DOIs for the same document
3. A group of documents (book series) can have a DOI
4. Different formats of the same document can have different DOIs

The relationship between different DOIs could be represented using a hierarchical data structure (e.g. Modified Preorder Tree Transversal), already implemented in many frameworks and libraries in different languages. In the data structure, the level of each DOI will be set by information retrieved from Crossref (specifically, `doi` type).

The product should also be able to get around the issue of having the same resource (e.g. the same journal article), with the **same DOI, available at different URLs**. The way publishers structure URLs is not something which can be controlled, and the product must expect that many different URLs point to the same DOI. This issue will be solved with a **many-to-one** relationship

between URLs and DOI. The publisher will also be in charge of uploading a list of all the available URLs for each single DOI.

To have a better understanding of the URLs, the URL related to the DOI as referenced in CrossRef could also be marked as **primary** in the database, distinguishing it from all the other URLs pointing to the same DOI.

E. Solving edge cases

In some edge cases where the software will not be able to follow the rules defined above to associate URLs with DOIs and the relative metrics, the match will be marked as `to_review` in the database and a manual review process will be put in place. The publisher will be presented with a fast, intuitive admin screen where all the edge cases will be collected and the possible solutions will be suggested; the final call will be on the publisher. Some of the following cases will be considered in this context:

- a URL resolves to another URL (redirect): the latter is listed under a specific publisher, but the first is not in the list of URLs defined from the publisher;
- a URL listed by a publisher does not resolve any more to the listed DOI;

F. Results

The results could be exposed in different ways. Pagination will be available on all responses.

a) By source

The response contains an array of metrics sources (e.g. Twitter, Facebook, etc.) and the related number of shares at the datetime of the request; useful for a text-based widget.

```
GET: /altmetrics/{doi}?view=source (or by default
/altmetrics/{doi}?)
{
  "count": 2
  "next": "https://api.example.org/accounts/?page=3",
  "previous": "https://api.example.org/accounts/?page=1",
  "total_shares": 3,
  "results": [
    {
      "id": "41e4544f-2adf-418f-8871-8f7e2853ec15",
      "source": "twitter",
```



```

        "shares": 1,
        "scraped_on": "2017-08-08T13:42:55+00:00"
    },
    {
        "id": "aed09375-4fe1-4e75-9a03-d24387b1e653",
        "source": "facebook",
        "shares": 2,
        "scraped_on": "2017-07-08T13:42:55+00:00"
    },
]
}

```

b) By time

The response contains an array of datetimes representing the different scrapes, and the relevant metrics related to each source; useful for a graph-based widget.

```

GET:
/altmetrics/{doi}?view=time&start_date={dt}&end_time={dt}
{
    "count": 2
    "next": "https://api.example.org/accounts/?page=3",
    "previous": "https://api.example.org/accounts/?page=1",
    "results": [
        {
            "id": "41e4544f-2adf-418f-8871-8f7e2853ec15",
            "timestamp": "2013-11-18T13:42:55+00:00",
            "twitter": 0,
            "facebook" : 0
        },
        {
            "id": "aed09375-4fe1-4e75-9a03-d24387b1e653",
            "timestamp": "2013-11-19T13:42:55+00:00",
            "twitter": 0,
            "facebook" : 0
        },
    ],
}

```

c) By source

Expanding context: the response will contain the whole content of a citation on a social media (e.g. the tweet object or Facebook post).

GET: /altmetrics/{doi}?view=source&expand=context

```
{
  "count": 2
  "next": "https://api.example.org/accounts/?page=3",
  "previous": "https://api.example.org/accounts/?page=1",
  "total_shares": 2,
  "results": [
    {
      "id": "41e4544f-2adf-418f-8871-8f7e2853ec15",
      "source": "twitter",
      "shares": 1,
      "scraped_on": "2017-07-08T13:42:55+00:00",
      "objects": [
        {
          "ref": "894904696690941952",
          "content": "text http://ddd.com",
          "created_at": "2016-07-08T13:42:55+00:00"
        }
      ]
    },
    {
      "id": "aed09375-4fe1-4e75-9a03-d24387b1e653",
      "source": "facebook",
      "shares": 1,
      "scraped_on": "2017-11-19T08:18:31.809000",
      "objects": [
        {
          "ref": "383407285386411",
          "content": "post text http://ddd.com",
          "created_at": "2017-07-01T13:42:55+00:00"
        }
      ]
    }
  ]
}
```

d) By scrape ID

Using the unique identifier of a scrape process (which will be executed at regular intervals); items having the same scrape ID have been scraped together.

```
GET: /altmetrics/{scrape_id}
{
  "count": 2
  "next": "https://api.example.org/accounts/?page=3",
  "previous": "https://api.example.org/accounts/?page=1",
  "total_shares": 3,
  "results": [
    {
      "source": "twitter",
      "shares": 1,
      "scraped_on": "2017-07-08T13:42:55+00:00"
    },
    {
      "source": "facebook",
      "shares": 2,
      "scraped_on": "2017-07-08T13:42:55+00:00"
    }
  ]
}
```

IV. Citations Service

The purpose of this endpoint is to provide a response of all the citations for a specific DOI from CrossRef⁴. The default ordering of the response data is by time, and the result can be further filtered by time of publishing of the citing source. The main content of the response data will be an array of DOIs, each being a source citing the requested DOI, along with the publishing date of each citing DOI.

```
GET: /altmetrics/{doi}?view=citation&start_date={dt}&end_time={dt}
{
```

⁴ See: <https://support.crossref.org/hc/en-us/articles/214318946-Retrieving-cited-by-matches>

```
"count": 2
"next": "https://api.example.org/accounts/?page=3",
"previous": "https://api.example.org/accounts/?page=1",
"results": [
  {
    "doi": "10.1007/BF000155550",
    "published": "2017-07-08T13:42:55+00:00"
  },
  {
    "doi": "10.1007/BAA00016551",
    "published": "2015-07-09T13:42:55+00:00"
  }
]
```

V. OAMetrics Service

Open Book Publishers will provide partners with: software to collect metrics from different platforms, an Open Standard format to enable the upload of statistics from other platforms, and an API to allow the submission, and subsequent query, of such metrics.

The OAMetrics Service will be thus clearly divided in two software components: the **Statistics Collection Agent** and the **Reader Analytics API**, and the definition of a format -an **OAMetrics Open Standard**- to be used consistently across all readership and download metrics.

A. OAMetrics Open Standard

The proposed standard for the submission and aggregation of readership and download metrics places particular emphasis in the definition of the **measure**, which describes the type and origin of the collected metric. The other key element of this standard is the identifier that will allow mapping the values of the measures to a particular object. We have chosen the Digital Object Identifier (**DOI**) as such key, given its versatility⁵; even though the main object of this project will be the research monograph, by using DOIs to identify the object of study this standard could well be used to accommodate any type of metric for any object that can be assigned a DOI (e.g. chapters, figures, tables, etc.).

⁵ International DOI Foundation. 2017. Digital Object Identifier Handbook. URL: https://www.doi.org/doi_handbook/1_Introduction.html#1.5

Apart from the measurement and the object it identifies, geographic location becomes a very important factor when analysing the impact of a publication, and although many dissemination platforms do not provide such data, it is imperative to allow its inclusion in this standard. We have chosen to include **country** information, when available, in ISO 3166-1 alpha-2⁶ format.

Furthermore, in order to being able to uniquely identify these metrics, we must also include **chronological information** in this definition; for which we will use the complete date plus hours, minutes, and seconds, with time zone, as specified in ISO 8601⁷.

1. Measures

Although the list could potentially be extremely extensive, disseminating platforms tend to keep the number of measures offered to a minimum, normally exposing only book views, but sometimes also page views, book downloads, and/or unique book views⁸. One of the ambitions of defining this format is to achieve a sufficient level of flexibility that will allow its application to metrics collected from different platforms that are already using different criterion, but also to anticipate to future specifications. Consequently, there cannot exist a predefined list of available measures in this standard, but rather a set of rules that allow defining them.

It is therefore proposed that the concept of a measure is made of three components: a platform, a namespace, and a type; and that each measure will be identified by a unique name within its namespace.

The platform component will identify where the data was collected from (e.g. Google Books, Open Edition, World Reader, etc.), and the type will reflect what the measure represents (e.g., a visit, a download, a session, etc.). These two components alone are actually sufficient to define a measurement, however, they fail to identify those measures that can be aggregated together. Instead of imposing a unique point of view, it is desired to accommodate each different interpretation of measure aggregation allowing the users of this standard to create their own definitions.

Each set of measure definitions will be collected within a namespace (e.g. hirmeos.eu, or readership.hirmeos.eu) and will represent a standard of the different measures that can be

⁶ See: https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2, <https://www.iso.org/iso-3166-country-codes.html>

⁷ See: <https://www.iso.org/iso-8601-date-and-time-format.html>

⁸ See: https://support.google.com/books/partner/answer/3323499?hl=en-GB&ref_topic=3324029,
<https://logs.openedition.org/awstats.pl?config=narratologie>,
<https://support.google.com/analytics/answer/2731565?hl=en>

aggregated together, as agreed by the parties involved in the definition of that particular namespace. Consequently, namespaces represent not only a flag for technical purposes but also an indication of the organisation that takes responsibility of its definition. As far as the current implementation is concerned, each of the partners in HIRMEOS will be able to contribute to measures held under the `hirmeos.eu` namespace, but they will also be able to add measures to their own namespace, for which they will be solely in charge. Likewise, and for instance, OBP will be able to contribute to measures `@hirmeos.eu`, but also to `@openbookpublishers.com` and, subsequently, any subdomain of such.

B. Statistics Collection Agent

Metrics collection will occur locally within the publisher's infrastructure, and the resulting data will be uploaded to the common, remote, metrics database. This methodology will put each publisher in charge of their own statistics, and ensure that the credentials needed to obtain such data from disseminating platforms remain secured, since most platforms do not share these statistics publicly. Furthermore, running the agent in a local environment will significantly reduce the workload on the remote service, leaving external responsibility to a bare minimum, thus facilitating future migrations of the Reader Analytics API to a different service provider, if ever required.

1. Drivers

The Statistics Collection Agent will be developed in a modular fashion, where each of the modules constituting the software will be called drivers; one for each of the disseminating platforms which one wish to collect metrics from. These drivers will work independently of one another, minimising the risk of failure, and allowing future expansions and the implementation of new drivers; as well as publishers will not need to run, or even install, drivers for those platforms in which they do not possess any content. The role of these modules will be both the download of metrics from its disseminating platform and the upload to the remote database via the metrics API. Publishers will be encouraged to schedule drivers to run daily via cron jobs in order to keep metrics updated.

The list of drivers -subject to future expansion- that will be developed in this project includes: Google Analytics, Access Logs, Google Books, Open Edition, OAPEN, Wikimedia, Unglue.it, The Classics Library, and World Reader. The Google Analytics driver will be provided as a generic template that will allow disseminating websites (and publishers who also act as such) to develop a driver for their own online readers; those who do not wish to share their usage data with commercial third parties will be able to use the Access Logs driver to achieve the same purpose. An Access Logs based driver to provide COUNTER compliant metrics will also be provided; this driver will be modified slightly to use the more permissive definition of a session that COUNTER

entails; and even though publishers will not need to submit these metrics to the remote database if they do not wish to, they could do so by creating a subdomain within their namespace to store these (e.g. counter.openbookpublishers.com) thus avoiding its aggregation with other metrics.

Some of the disseminating platforms provide APIs that can be queried to obtain metrics, others require web scraping. Even though generic drivers for each use case cannot be provided given that each API or web interface will differ from one another, it is expected that publishers will be able to create new modules for platforms not covered in this project based on existing drivers that follow the same methodology they intend to produce. For instance, Google Books does not provide a metrics API, only a reporting interface where publishers and authors can download a CSV report; taking its driver as an example it would be trivial to develop a new module for a different platform which lacked an API but provided metrics reports.

2. Identifier Translation Service Back End

Each disseminating platform makes use of a different identifier for their content: Google Books using ISBN numbers, World Reader book titles, Open Edition URLs, etc.; however, partner publishers and platforms have chosen the DOI as the primary key for their publications and we will indeed wish to store the collected metrics against DOIs. The solution to this problem is to make all drivers communicate with the Identifier Translation Service, a common service that will enable drivers to convert the primary keys obtained from their platform to a DOI.

This service will store each mapping within a local database, for which an API will be provided, and will interact with Crossref REST API⁹ whenever it fails to obtain a DOI from this database.

There will be occasions where a single publication has been assigned multiple DOIs and the translation service will fail to programmatically return a single mapping. In order to avoid heuristic resolutions, which are very error prompting, publishers will be allowed to assign a canonical DOI for each publication; thus ensuring a conscious resolution of the conflict.

The translation API will therefore enable the storage of book metadata, and facilitate DOI conflict resolution through canonicalisation. Publishers will be able to perform these actions with the provided front end, or they can otherwise integrate them within their existing administration systems.

3. Identifier Translation Service Front End

The front end of the service will be developed in the form of a web application, a simple client to the API that will be notified of DOI conflicts and will allow resolving them. In order to avoid such

⁹ See: <https://github.com/CrossRef/rest-api-doc>

conflicts in the first place, users will be able to administer book metadata via CRUD interfaces. Bulk upload of metadata will also be permitted and, if provided, DOIs input by the publisher will be considered canonical.

An interface listing all drivers installed by the publisher will display whether they were successful last time they run and will allow fixing any potential errors if they were not; a mock-up of this interface can be seen in figure 3.

The error reporting interface will display exceptions caused by drivers that failed to run and will determine a possible solution to the error. Three example errors have been displayed in figure 4: a) a DOI conflict, where multiple DOIs have been found for the same publication; the solution to this problem has been determined correctly, the user needs to edit the publication, where they will be able to set a canonical DOI for that record; b) again correctly presented with the solution of adding/editing the global constants where credentials and other configuration settings will be stored; c) this example was introduced as a means to justify the need of human interaction in this service, the error indeed claims that a DOI cannot be found for the given title, however it may well be that such title is already in the system and that the exception was caused by a mere encoding problem; the solution in this case is not to add a new record, but to edit an existing one.

Statistics Collection Agent				Drivers Status	Errors	Publications	Admin ▾
Driver	Last Run	Status	Action				
ClassicsLibrary	2017-07-01	OK	Run				
OBP PDF Reader	2017-07-28	OK	Run				
OBP HTML Reader	2017-08-28	OK	Run				
Wikimedia	2016-09-01	Out of date	Run	Fix errors			
GoogleBooks	2017-07-01	OK	Run				
OpenEdition	2017-07-01	OK	Run				
Ungluit	2014-01-01	Stale	Run	Fix errors			
World Reader	2016-12-01	Out of date	Run	Fix errors			

Figure 3. Statistics Collection Agent: Driver Status Interface

Statistics Collection Agent					
		Drivers Status	Errors	Publications	Admin ▾
Driver	Timestamp	Type	Error	Action	
WorldReader	2017-08-22 02:46:23	DOI conflict	Multiple DOI for key: 978-1-78374-208-0 at IdentifierTranslation\$query.isbn(Translator.java:283)	Edit Record	Ignore
Unglueit	2017-08-01 01:32:15	Wrong credentials	\$UNGLUEIT_USER not defined! at driver.Unglueit\$Api.login(Unglueit.java:143)	Edit Credentials	Ignore
Wikimedia	2017-07-07 04:12:16	Missing DOI	Not found: Die Europaidee im Zeitalter der Aufklärung at IdentifierTranslation\$query.title(Translator.java:587)	Add Record	Ignore

Figure 4. Statistics Collection Agent: Error Reporting Interface

C. Reader Analytics Service

The Reader Analytics Service will constitute the component of the common metrics API that will enable the submission and posterior query of readership metrics. While POST requests to the API will be restricted to organisations that are part of a namespace included in the service, GET requests to query the database will be permitted publicly.

1. Database Structure

The entity relationship diagram of the reader analytics database is attached in [appendix A](#), and displays the different interactions that each object shares with one another. The event table represents the value of a measurement, where its composite primary key reflects the OAMetrics Open Standard detailed earlier in this paper. Given that “there is no defined limit on the length of the DOI name, nor of its prefix or its suffix elements”¹⁰, the doi attribute of the event table has been defined as character of varying length, hence leaving the validation of these identifiers to the API, rather than the database itself.

¹⁰ Norman Paskin. 2009. *Digital Object Identifier (DOI®) System*. URL: <http://www.doi.org/overview/DOI-ELIS-Paskin.pdf>

2. POST Requests

POST requests will be restricted by means of tokenization. Furthermore, only organisations that are registered as contributors of each namespace -those who possess a relationship in the namespace_uploader table- will be allowed to submit data to measures associated with such namespace, these requests will therefore be subject to a further check that will determine whether the uploaded possesses the right permission.

3. GET Requests

Queries for a particular DOI will return the complete list of metrics collected for such identifier, unless filtered by some of the attributes explained earlier; each request will be structured to follow this pattern: /readership/{doi}?filter={name}:{value}. Table 1 shows the list of filters that will be available - two example requests and their results can be seen in appendix B.

A combination of any of these filters should be possible in order to narrow the result, providing they do not contradict each other, e.g., `measure.id = obp-html@hirmeos.eu` and `measure.namespace = somethingelse` would not be possible since the `measure.id` already specifies a particular namespace.

Queries performed by the widget, or any other aggregation purpose, will therefore need to be filtered by the pertinent namespace using the `measure.namespace` filter.

Filter name	Filter value	Example
measure.id	measure_id	?filter=measure.id:obp-html@hirmeos.eu
measure.namespace	measure_namespace	?filter=measure.namespace:hirmeos
measure.type	measure_type	?filter=measure.type:sessions
start-date	date	?filter=start-date:2017-01-01
end-date	date	?filter=end-date:2017-07-31
country	iso2	?filter=country:GB

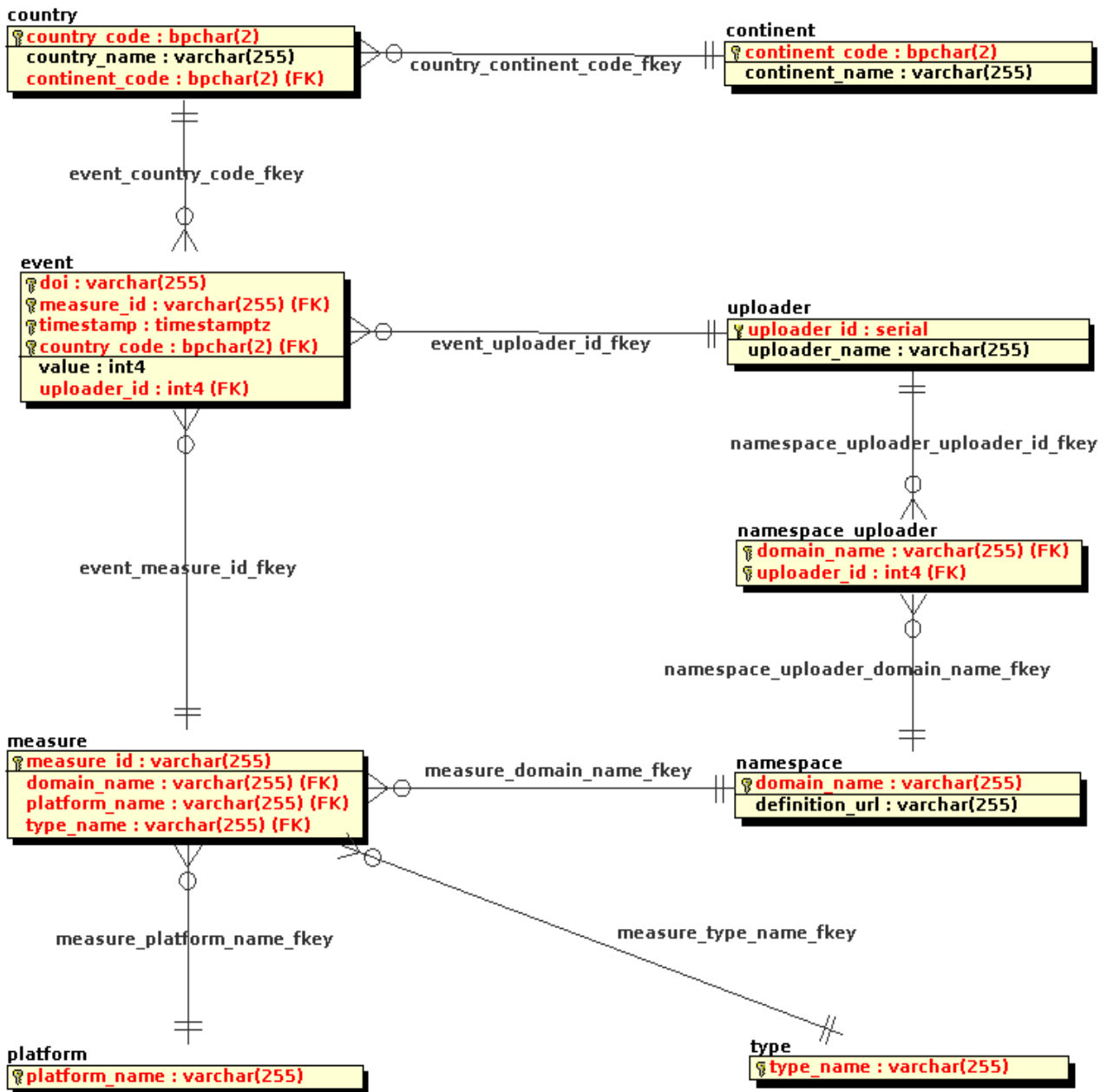
Table 1. Allowed filters in readership GET requests

4. DELETE Requests

In order to avoid inconsistency problems, and data loss, DELETE requests on an individual basis will not be allowed, instead, partners will be granted permission to delete all the metrics they have uploaded themselves within a date range for specific namespace.

Appendices

A. Reader Analytics Database Entity Relationship Diagram




```
    "namespace": "hirmeos.eu"
    "type": "visits"
  },
  "country": {
    "iso2": "ES",
    "name": "Spain"
  "continent": {
    "code": "EU",
      "name": "Europe",
    }
  },
  "value": 43
}
}]
}
```

GET: /readership/measures

```
{
  "status": "success",
  "data": {
    "count": 9,
    "objects": [{
      "id": "google-books@hirmeos.eu",
      "platform": "Google Books"
      "namespace": "hirmeos.eu"
      "type": "visits"
    }, {
      "id": "world-reader-binu@hirmeos.eu",
      "namespace": "hirmeos.eu"
      "platform": "World Reader biNu"
      "type": "visits"
    }, {
      "id": "world-reader-web@hirmeos.eu",
      "namespace": "hirmeos.eu"
      "platform": "World Reader Website"
      "type": "visits"
    }, {
      "id": "internet-archive@hirmeos.eu",
      "namespace": "hirmeos.eu"
      "platform": "Internet Archive"
      "type": "visits"
    }, {
      "id": "open-edition@hirmeos.eu",
```

```
"namespace": "hirmeos.eu"  
"platform": "Open Edition"  
"type": "visits"  
}, {  
  "id": "obp-html@hirmeos.eu",  
  "namespace": "hirmeos.eu"  
  "platform": "Open Book Publishers HTML Reader"  
  "type": "sessions"  
}, {  
  "id": "obp-pdf@hirmeos.eu",  
  "namespace": "hirmeos.eu"  
  "platform": "Open Book Publishers PDF Reader"  
  "type": "sessions"  
}, {  
  "id": "obp-downloads@hirmeos.eu",  
  "namespace": "hirmeos.eu"  
  "platform": "Open Book Publishers Downloads"  
  "type": "downloads"  
}, {  
  "id": "classics-library@hirmeos.eu",  
  "namespace": "hirmeos.eu"  
  "platform": "Classics Library"  
  "type": "sessions"  
}  
]  
}  
}
```


C. CrossRef XML response

```

1. <crossref_result
2.   xmlns="http://www.crossref.org/qrschema/3.0"
3.   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"                                version="3.0"
   xsi:schemaLocation="http://www.crossref.org/qrschema/3.0
   http://www.crossref.org/schemas/crossref_query_output3.0.xsd">
4.   <query_result>
5.     <head>
6.       <doi_batch_id/>
7.     </head>
8.     <body>
9.       <query key="" status="resolved">
10.        <doi type="journal_article">10.5334/sta.at</doi>
11.        <crm-item name="publisher-name" type="string">Ubiquity Press, Ltd.</crm-
item>
12.        <crm-item name="prefix-name" type="string">Ubiquity Press, Ltd.</crm-item>
13.        <crm-item name="member-id" type="number">3285</crm-item>
14.        <crm-item name="citation-id" type="number">61027341</crm-item>
15.        <crm-item name="journal-id" type="number">191085</crm-item>
16.        <crm-item name="deposit-timestamp" type="number">1408110993</crm-item>
17.        <crm-item name="owner-prefix" type="string">10.5334</crm-item>
18.        <crm-item name="last-update" type="date">2015-11-03T13:39:10Z</crm-item>
19.        <crm-item name="created" type="date">2013-03-07T15:47:06Z</crm-item>
20.        <crm-item name="citedby-count" type="number">0</crm-item>
21.        <doi_record>
22.          <crossref
23.            xmlns="http://www.crossref.org/xschema/1.1"
   xsi:schemaLocation="http://www.crossref.org/xschema/1.1
   http://doi.crossref.org/schemas/unixref1.1.xsd">
24.            <journal>
25.              <journal_metadata>
26.                <full_title>
27. Stability: International Journal of Security and Development
28. </full_title>
29.                <abbrev_title>sta</abbrev_title>
30.                <issn media_type="electronic">2165-2627</issn>
31.              </journal_metadata>
32.              <journal_issue>
33.                <publication_date media_type="online">
34.                  <month>02</month>
35.                  <day>22</day>
36.                  <year>2013</year>
37.                </publication_date>
38.                <journal_volume>
39.                  <volume>2</volume>
40.                </journal_volume>
41.                <issue>1</issue>

```

```

42.         </journal_issue>
43.         <journal_article publication_type="full_text">
44.             <titles>
45.                 <title>In Defence of New Wars</title>
46.             </titles>
47.             <contributors>
48.                 <person_name                contributor_role="author"
sequence="first">
49.                     <given_name>Mary</given_name>
50.                     <surname>Kaldor</surname>
51.                 </person_name>
52.             </contributors>
53.             <publication_date media_type="online">
54.                 <month>03</month>
55.                 <day>07</day>
56.                 <year>2013</year>
57.             </publication_date>
58.             <pages>...</pages>
59.             <doi_data>...</doi_data>
60.             <citation_list>...</citation_list>
61.         </journal_article>
62.     </journal>
63. </crossref>
64. </doi_record>
65. </query>
66. </body>
67. </query_result>
68. </crossref_result>
69.

```