



FUNDING AGENCY DASHBOARD SERVICE API

V 1.0, October 2015

[Purpose](#)

[Basic Mechanism](#)

[Funding Agency Dashboard Service API](#)

[API Change Log](#)

[Change Notification Process](#)

[Issues/Questions](#)

Purpose:

This API documentation is intended to assist the user with understanding how the CHORUS Funder Dashboard Service delivers aggregated data and paths to actual articles, please see <http://www.chorusaccess.org/services/dashboard-service/> for more information regarding the Agency Dashboard Service.

Basic Mechanism:

To use this document, you should be familiar with API concepts. All APIs return response as JSON by default. In case a response is need in XML the Content-Type of the request header needs to be set to either text/xml or application/xml.

This API will only work for the FunderID's for those Funder Dashboards that were created.

Funder Agency Dashboard APIs:

Four APIs were created to return dashboard data.

API Endpoint	HTTP Method	Description
/agencies/:id/histories	GET	Returns data of indicator history table
/agencies/:id/publishers	GET	Returns data of publisher breakdown table
/agencies/:id/histories/:yyyy/:mm/:dd	GET	Returns works associated with numbers displayed in indicator history table

/agencies/:id/publishers /:memberId	GET	Returns work records corresponding to numbers in publisher table
--	-----	--

API 1: For fetching the data of indicator history table

API Endpoint: /agencies/:id/histories

Method: GET

Parameters to be passed:

Parameter	Type	Description
:id(<i>Required</i>)	URL	Funder Id of agency whose data is to be fetched

Headers:

In order to request data in XML format, following header needs to be set for the API call:

'Content-Type: text/xml'

OR

'Content-Type: application/xml' (JSON is returned by default)

Response Codes:

200 – Successful Request

404 – Requested agency not found in dashboard database

500 – Something went wrong with the server

Response:

- agency_id – ID of agency/funder whose data was requested
- agency_name – Name of Agency
- tallies – An array of object corresponding to all rows in indicator history table

Example cURL requests:

```
curl "https://api.chorusaccess.org/agencies/100000001/histories"
```

In case data is needed as XML:

```
curl -H "Content-Type: text/xml"
```

```
"https://api.chorusaccess.org/agencies/100000001/histories"
```

Sample Response:

```
{
  "agency_id": "100000001",
  "agency_name": "National Science Foundation",
  "tallies": [
```

```
{
"date": "12/31/2014",
"total_works": 23070,
"publicly_accessible_yes": 9930,
"publicly_accessible_unknown": 13140,
"reuse_terms_available_yes": 8142,
"reuse_terms_available_unknown": 14928,
"archived_yes": 7672,
"archived_unknown": 15398,
"acceptable_overall": 7566
}
]
}
```

API 2: For fetching the data of publisher breakdown table

API Endpoint: /agencies/:id/publishers

Method: GET

Note(s): The parameters, headers, and response codes section for this API are the same as in API 1 above.

Response:

- agency_id – ID of agency/funder whose data was requested
- agency_name – Name of Agency
- publishers – An array of object corresponding to all rows in publisher table

Example cURL requests:

```
curl "https://api.chorusaccess.org/agencies/100000001/publishers"
```

In case data is needed as XML:

```
curl -H "Content-Type: text/xml"
```

```
"https://api.chorusaccess.org/agencies/100000001/publishers"
```

Sample Response:

```
{
"agency_id": "100000001",
"agency_name": "National Science Foundation",
"publishers": [
{
"publisher": "American Diabetes Association",
"total_works": 1,
"publicly_accessible_yes": 0,
"publicly_accessible_unknown": 1,
"reuse_terms_available_yes": 0,
```

```

"reuse_terms_available_unknown": 1,
"archived_yes": 0,
"archived_unknown": 1,
"acceptable_overall": 0
}
]
}

```

API 3: For fetching the works associated with numbers displayed in indicator history table

API Endpoint: /agencies/:id/histories/:yyyy/:mm/:dd

Method: GET

Note(s): This API returns 50 records per request; setting limit query parameter to a higher value while making request can increase this limit. The headers and response codes section for this API are the same as in API 1 above.

Parameters to be passed:

Parameter	Type	Description
:id (Required)	URL	Funder Id of agency whose data is to be fetched
:YYYY (Required)	URL	Year of date of which data is to be fetched
:mm (Required)	URL	Month of date of which data is to be fetched
:dd (Required)	URL	Day of date of which data is to be fetched
category (Optional)	Query	This can be one of, publicly_accessible, archive, license and total_acceptable. Default value for this parameter is 'all'.
subcategory (Optional)	Query	This can be one of 'yes' or 'unknown'. If only category is passed as query parameter with request, subcategory defaults to 'yes'
limit (Optional)	Query	Number of records to be returned per request/page. Defaults to 50
offset (Optional)	Query	Pagination offset. Default value is 0

Response:

- agency_id – ID of agency/funder whose data was requested
- agency_name – Name of Agency
- breakdown_for – Date whose data is requested
- category – category parameter passed with request
- subcategory – subcategory parameter that was passed with request
- items – Array of work records
- limit – Number of records requested
- offset – Result offset
- total_results – Total number records matching the request

Example cURL requests:

curl

```
"https://api.chorusaccess.org/agencies/100000161/histories/2015/2/23?category=archive
&subcategory=yes&limit=100"
```

In case data is needed as XML:

```
curl -H "Content-Type: text/xml"
```

```
"https://api.chorusaccess.org/agencies/100000161/histories/2015/2/23?category=archive
&subcategory=yes&limit=100"
```

Sample Response:

```
{
"agency_id": "100000161",
"agency_name": "National Institute of Standards and Technology",
"breakdown_for": "2015/2/23",
"category": "archive",
"subcategory": "yes",
"items": [
{
"DOI": "10.1002/2014jd022555",
"URL": "http://dx.doi.org/10.1002/2014jd022555",
"publisher": "Wiley-Blackwell",
"authors": [
"Turnbull Jocelyn C.",
"Sweeney Colm",
"Karion Anna",
"Newberger Timothy",
"Lehman Scott J.",
"Tans Pieter P.",
"Davis Kenneth J.",
```

```

"Lauvaux Thomas",
"Miles Natasha L.",
"Richardson Scott J.",
"Cambaliza Maria Obiminda",
"Shepson Paul B.",
"Gurney Kevin",
"Patarasuk Risa",
"Razlivanov Igor"
],
"title": " Toward quantification and source sector identification of fossil fuel CO 2
emissions from an urban area: Results from the INFLUX experiment ",
"journal_name": "Journal of Geophysical Research: Atmospheres",
"publication_date": "1/12/2015",
"audited_on": "1/12/2015",
"publicly_accessible_date": null
}
],
"limit": 1,
"offset": 0,
"total_results": 61
}

```

API 4: For fetching work records corresponding to numbers in publisher table

API Endpoint: /agencies/:id/publishers/:memberId

Method: GET

Note(s): This API returns 50 records per request; setting limit query parameter to a higher value while making request can increase this limit. The headers and response codes section for this API are the same as in API 1 above.

Parameters to be passed:

Parameter	Type	Description
:id (Required)	URL	Funder ID of agency whose data is to be fetched
:memberId (Required)	URL	Member id of publisher whose data is to be fetched
category (Optional)	Query	This can be one of, publicly_accessible, archive, license and total_acceptable. Default value for this parameter is 'all'.
subcategory (Optional)	Query	This can be one of 'yes' or 'unknown'. If only category is passed as query parameter with request, subcategory defaults to 'yes'
limit (Optional)	Query	Number of records to be returned per request/page. Defaults to 50

offset (Optional)	Query	Pagination offset. Default value is 0
----------------------	-------	---------------------------------------

Response:

- agency_id – ID of agency/funder whose data was requested
- agency_name – Name of Agency
- breakdown_for – Publisher whose data is requested
- category – category parameter passed with request
- subcategory – subcategory parameter that was passed with request
- items – Array of work records
- limit – Number of records requested
- offset – Result offset
- total_results – Total number records matching the request

Example cURL requests:

```
curl
"https://api.chorusaccess.org/agencies/100000161/publishers/16?category=publicly_accessible&subcategory=yes"
```

In case data is needed as XML:

```
curl -H "Content-Type: text/xml"
```

```
"https://api.chorusaccess.org/agencies/100000161/publishers/16?category=publicly_accessible&subcategory=yes"
```

Sample Response:

```
{
  "agency_id": "100000161",
  "agency_name": "National Institute of Standards and Technology",
  "breakdown_for": "American Physical Society (APS)",
  "category": "publicly_accessible",
  "subcategory": "yes",
  "items": [
    {
      "DOI": "10.1103/physreva.83.032501",
      "URL": "http://dx.doi.org/10.1103/physreva.83.032501",
      "publisher": "American Physical Society (APS)",
      "authors": [
        "Pachucki Krzysztof",
        "Komasa Jacek"
      ],
      "title": "Magnetic dipole transitions in the hydrogen molecule",
      "journal_name": "Physical Review A",
    }
  ]
}
```

```
"audited_on": "11/1/2014"  
}  
],  
"limit": 1,  
"offset": 0,  
"total_results": 49  
}
```

API Change Log:

Revisions Date	Change	Reason for change

Change Notification Process:

Changes and/or updates will be communicated by notifying the technical representative noted in the CHORUS participant account details.

Issues/questions:

If you find a bug in this documentation or would like to propose an improvement to the API, please send an email to (support@chorusaccess.org). If you have a suggestion how to fix it, include that as well.