DH-publish Documentation
Release 2019

DARIAH-DE

December 04, 2019

A PFD version of the DH-publish Service API Documentation you can find HERE.

API DOCUMENTATION

The DH-publish service's API is not yet fully implemented, nevertheless it is written down here how it works at the moment.

For every API call of the DH-publish service where IDs are involved, you do need a storage token and (optionally) a transaction ID. The storage token you can get via the DARIAH-DE Publikator. Just go to the Publikator and click on **Start with the DARIAH-DE Publikator**. Then please do log in via your DARIAH or DFN-AAI account and finally click on **reveal storage token** and copy the token (of the form **252dad1a-7b78-4384-88d1-59ba4e0edd42**) or just click the **Copy the OAuth Token to your clipboard.** button (see tooltip).

1.1 #VERSION

HTTP POST /version

To check the current productive DH-publish version simply try:

http://repository.de.dariah.eu/1.0/dhpublish/version

and you get the currently deployed version of the productive and public DH-publish service.

1.2 #PUBLISH

HTTP POST /{Storage-ID}/publish

Request Parameters

Parameter	Position	Type	Description
Storage-ID	Path	String	The OwnStorage ID of the root collection (mandatory)
X-Storage-Token	Header	String	DARIAH-DE OAuth2 storage token (mandatory)
X-Transaction-ID	Header	String	ID used for logging (optional)

Response:

```
200 OK: Calling DH-publish service [{Storage-ID}]
400 Bad Request
401 Unauthorized
405 Method Not Allowed
500 Internal Server Error
503 Service Unavailable
```

Example Request:

```
curl -H "X-Storage-Token: 252dad1a-7b78-4384-88d1-59ba4e0edd42" -H "X-Transactio
```

Example Response:

```
Calling DH-publish service [https://de.dariah.eu/storage/EAEA0-A43B-574C-4D38-0]
```

Example of a root collection object (TTL):

```
@prefix dcterms:      <http://purl.org/dc/terms/> .
@prefix dariah:       <http://de.dariah.eu/rdf/dataobjects/terms/> .
@prefix dariahstorage: <https://de.dariah.eu/storage/> .
@prefix dc:           <http://purl.org/dc/elements/1.1/> .

dariahstorage:EAEA0-7113-BC10-9AF8-0
    a        dariah:Collection .

dariahstorage:EAEA0-15C8-2916-7718-0
    a           dariah:DataObject ;
    dc:creator  "fu" ;
    dc:format   "image/gif" ;
    dc:rights   "restricted" ;
    dc:title    "Dragon.gif" .

dariahstorage:EAEA0-A43B-574C-4D38-0
    a              dariah:Collection ;
    dc:creator        "fu" ;
    dc:rights         "free" ;
    dc:title          "This is a very nice root collection file!" ;
    dcterms:hasPart  ( dariahstorage:EAEA0-15C8-2916-7718-0 dariahstorage:EAEA0-
```

Example of a subcollection object (TTL):

```
@prefix dcterms:      <http://purl.org/dc/terms/> .
@prefix dariah:       <http://de.dariah.eu/rdf/dataobjects/terms/> .
@prefix dariahstorage: <https://de.dariah.eu/storage/> .
@prefix dc:           <http://purl.org/dc/elements/1.1/> .

dariahstorage:EAEA0-7113-BC10-9AF8-0
    a              dariah:Collection ;
    dc:creator        "fu" ;
    dc:rights         "free" ;
    dc:title          "Another very very nice subcollection file!" ;
    dcterms:hasPart  ( dariahstorage:EAEA0-9131-72D3-FA4E-0 ) .
```

```
dariahstorage:EAEA0-9131-72D3-FA4E-0
    a           dariah:DataObject ;
    dc:creator  "fu" ;
    dc:format   "image/gif" ;
    dc:rights   "restricted" ;
    dc:title    "A Magenta Worm" .
```

1.3 #STATUS

HTTP POST /{Storage-ID}/status

Request Parameters

Parameter	Position	Type	Description
Storage-ID	Path	String	The OwnStorage ID of the root collection (mandatory)
X-Storage-Token	Header	String	DARIAH-DE OAuth2 storage token (mandatory)
X-Transaction-ID	Header	String	ID used for logging (optional)

Response:

```
200 OK: XML or JSON response body
400 Bad Request
401 Unauthorized
405 Method Not Allowed
500 Internal Server Error
503 Service Unavailable
```

Example of a not yet published and not yet queued collection (XML status response):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<publishResponse objectListComplete="false">
    <PublishObject uri="https:/de.dariah.eu/storage/EAEA0-A43B-574C-4D38-0" stat
    <PublishStatus progress="0" processStatus="NOT_QUEUED" activeModule="de.lang
</publishResponse>
```

Example of a not yet published and not yet queued collection (JSON status response):

```
{
  "objectListComplete": false,
  "publishStatus": {
    "progress": 0,
    "processStatus": "NOT_QUEUED",
    "activeModule": "de.langzeitarchivierung.kolibri.webservice.dariahde.publish
  },
  "publishObjects": [
    {
      "uri": "https:/de.dariah.eu/storage/EAEA0-A43B-574C-4D38-0",
```

```
        "status": "NOT_YET_PUBLISHED"
    }
  ]
}
```

Example of a published collection (XML status response):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<publishResponse dryRun="false" objectListComplete="true">
    <PublishObject uri="https://de.dariah.eu/storage/EAEA0-A43B-574C-4D38-0" pid
    <PublishObject uri="https://de.dariah.eu/storage/EAEA0-15C8-2916-7718-0" pid
    <PublishObject uri="https://de.dariah.eu/storage/EAEA0-9131-72D3-FA4E-0" pid
    <PublishObject uri="https://de.dariah.eu/storage/EAEA0-7113-BC10-9AF8-0" pid
    <PublishStatus progress="100" processStatus="FINISHED" activeModule="de.lang
</publishResponse>
```

Example of a published collection (JSON status response):

```
{
  "dryRun": false,
  "objectListComplete": true,
  "publishStatus": {
    "progress": 100,
    "processStatus": "FINISHED",
    "activeModule": "de.langzeitarchivierung.kolibri.actionmodule.dariahde.publi
  },
  "publishObjects": [
    {
      "uri": "https://de.dariah.eu/storage/EAEA0-A43B-574C-4D38-0",
      "pid": "doi:10.20375/0000-001A-3887-D",
      "status": "OK"
    },
    {
      "uri": "https://de.dariah.eu/storage/EAEA0-15C8-2916-7718-0",
      "pid": "doi:10.20375/0000-001A-3888-C",
      "status": "OK"
    },
    {
      "uri": "https://de.dariah.eu/storage/EAEA0-9131-72D3-FA4E-0",
      "pid": "doi:10.20375/0000-001A-388A-A",
      "status": "OK"
    },
    {
      "uri": "https://de.dariah.eu/storage/EAEA0-7113-BC10-9AF8-0",
      "pid": "doi:10.20375/0000-001A-3889-B",
      "status": "OK"
    }
  ]
}
```

1.4 #MINISTATUS

HTTP POST /{Storage-ID}/ministatus

Request Parameters

Parameter	Position	Type	Description
Storage-ID	Path	String	The OwnStorage ID of the root collection (mandatory)
X-Storage-Token	Header	String	DARIAH-DE OAuth2 storage token (mandatory)
X-Transaction-ID	Header	String	ID used for logging (optional)

Response:

```
200 OK: XML or JSON response body
400 Bad Request
401 Unauthorized
405 Method Not Allowed
500 Internal Server Error
503 Service Unavailable
```

Example of a not yet published and not yet queued collection (XML ministatus response):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<publishResponse objectListComplete="false">
    <PublishObject uri="https://de.dariah.eu/storage/EAEA0-B52F-B2E3-1393-0" sta
    <PublishStatus progress="0" processStatus="NOT_QUEUED" activeModule="de.lang
</publishResponse>
```

Example of a not yet published and not yet queued collection (JSON ministatus response):

```
{
  "objectListComplete": false,
  "publishStatus": {
    "progress": 0,
    "processStatus": "NOT_QUEUED",
    "activeModule": "de.langzeitarchivierung.kolibri.webservice.dariahde.publish
  },
  "publishObjects": [
    {
      "uri": "https://de.dariah.eu/storage/EAEA0-B52F-B2E3-1393-0",
      "status": "NOT_YET_PUBLISHED"
    }
  ]
}
```

Example of a published collection (XML ministatus response):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<publishResponse dryRun="false" objectListComplete="true">
    <PublishObject uri="https://de.dariah.eu/storage/EAEA0-A43B-574C-4D38-0" pid
```

```
    <PublishStatus progress="100" processStatus="FINISHED" activeModule="de.lang
</publishResponse>
```

Example of a published collection (JSON ministatus response):

```json
{
  "dryRun": false,
  "objectListComplete": true,
  "publishStatus": {
    "progress": 100,
    "processStatus": "FINISHED",
    "activeModule": "de.langzeitarchivierung.kolibri.actionmodule.dariahde.publi
  },
  "publishObjects": [
    {
      "uri": "https://de.dariah.eu/storage/EAEA0-A43B-574C-4D38-0",
      "pid": "doi:10.20375/0000-001A-3887-D",
      "status": "OK"
    }
  ]
}
```

1.5 #INFO

HTTP POST /{Storage-ID}/info

Request Parameters

Parameter	Position	Type	Description
Storage-ID	Path	String	The OwnStorage ID of the root collection (mandatory)
X-Storage-Token	Header	String	DARIAH-DE OAuth2 storage token (mandatory)
X-Transaction-ID	Header	String	ID used for logging (optional)

Response:

```
200 OK: XML or JSON response body
400 Bad Request
401 Unauthorized
405 Method Not Allowed
500 Internal Server Error
503 Service Unavailable
```

Example of a published collection (XML info response):

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<infoResponse status="PUBLISHED" pid="21.T11991/0000-001A-3887-D" doi="10.20375/
```

Example of a published collection (JSON info response):

```
 {
    "status": "PUBLISHED",
    "pid": "21.T11991/0000-001A-3887-D",
    "doi": "10.20375/0000-001A-3887-D",
    "uri": "https://de.dariah.eu/storage/EAEA0-A43B-574C-4D38-0",
    "module": "de.langzeitarchivierung.kolibri.actionmodule.dariahde.publish.Publ
    "progress": 100,
    "rdf": "EAEA0-9AE3-F007-A5C2-0"
}
```

Example of a not yet published and not yet queued collection (XML info response):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<infoResponse status="DRAFT" pid="" doi="" uri="https://de.dariah.eu/storage/EAE
```

Example of a not yet published and not yet queued collection (JSON info response):

```
{
    "status": "DRAFT",
    "uri": "https://de.dariah.eu/storage/EAEA0-B52F-B2E3-1393-0",
    "progress": 0
}
```

SOURCES

See dhpublish_sources

THREE

BUGTRACKING

See dhpublish_bugtracking

LICENSE

See LICENCE