

MediaHaven Pro : Search API version 1.2

Doc version: 9/03/2013



zeticon

Contents

1	Introduction.....	3
2	Versions.....	3
2.1	Version 1.1 of API	3
2.2	Version 1.2 of API	3
3	API.....	3
3.1	URL and search function.....	3
3.2	Specifying the query	3
3.3	Pagination of results.....	4
3.4	Format of results.....	4
4	Example of XML query and result.....	4
4.1	Basic example	4
4.2	More complex example: getting collections.....	5
4.3	Kortrijk live example	5

1 Introduction

This API makes it possible to do search queries on the MediaHaven Pro backend application and get in return a list of metadata (including a URL) of media assets in the MediaHaven Pro backend.

The results are dependent on the userlogin (a parameter) which is used for querying (depending on the permissions of users and assets).

This first version of the API focusses only on images. In a second phase of the API, other multimedia (video, audio) will also be included.

2 Versions

2.1 Version 1.1 of API

Initial release, upto release 3.x.x of MediaHaven.

2.2 Version 1.2 of API

Starting from release 4.x.x of MediaHaven.

- The parameter organisation is a dummy parameter for now as the login is unique (and determines the organization)
- The API returns an extra field 'thumbnailImagePath' which is the URL of a thumbnail of max 130x130 pixels

3 API

3.1 URL and search function

The URL of the webservice will be of the following format:

<https://IPADDRESS or hostname/mediahaven-public-ws/Search?wsdl>

The MediaHaven public webservice exports the search function :

```
public SearchResult search(String login, String password,
                          String organisation, String query,
                          int startIndex, int nrOfResults) {
}
```

3.2 Specifying the query

This function has 4 parameters which will be defined by the owner of the MediaHaven Pro installation.

- login = username of a MediaHaven Pro user (e.g. publicmuseumA)
- password = password for the user
- organisation = dummy parameter, may be used in the future

- query = the query string for the search: the most simple query is an empty string, which returns all assets visible for that user. More advanced queries can limit the results to a particular keyword or period or.... Videofragments or audiofragments are not returned, only the full MediaObject

3.3 Pagination of results

The other 2 arguments are used to limit the number of results (pagination):

- startIndex = used for pagination of searchresults, searchresults will be returned starting from this index. E.g. index=0 starts from the first result, index=1 from the second result.
- nrOfResults = the number of results that will be returned

So for using pagination, a first query will be: Startindex=0, nrOfResults=10, followed by Startindex=10, nrOfResults=10 for 2nd page and so on.

3.4 Format of results

This function responds with a SearchResult object, this object has 3 properties:

- totalNrOfResults: total number of search results for the query (with this number you can show the number of pages of results)
- startIndex : startindex of the search results (=equal to the parameter which was sent)
- List of mediaData : object containing information about the found media:
 - title: the title of the media asset
 - description: description of the media asset
 - date: creation date of the media asset. This date has the following format yyyy:mm:dd hh:mm:ss (EXIF format)
 - previewImagePath: URL to the linked image for this media asset
 - thumbnailImagePath: URL to a 130x130 pixel thumbnail of this media asset
 - List of keywords of this media asset
 - type: type of media asset can be [Video, Audio, Image, Document, Collection, Set]
-> in this version of the API, every asset will have a preview URL (image) independent of its type. Also video will have a URL to a thumbnail. So, the API makes it only possible to show still images.
 - List of authors of the media asset
 - mediaObjectId: unique ID of the mediaobject, e.g. useful to get contents of a collection, see further

4 Example of XML query and result

4.1 Basic example

Test environment at Zeticon:

- <https://193.191.148.145>
- login: kortrijk1
- password: kortrijk1
- organisation: empty parameter or skip the parameter

```
curl -k -d '<?xml version="1.0" ?><S:Envelope
xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"><S:Body><ns2:search
xmlns:ns2="http://ws.api.mediahaven.zeticon.com/"><login>kortrijk1</login><password>kort
rijk1</password><organisation></organisation><query></query><startIndex>0</startIndex>
```

```
<nrOfResults>5</nrOfResults></ns2:search></S:Body></S:Envelope> -H "Content-type:
text/xml" https://193.191.148.145/mediahaven-public-ws/Search
```

Video will only return a still thumb, no video URL.

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:searchResponse xmlns:ns2="http://ws.api.mediahaven.zeticon.com/">
      <return>
        <mediaDataList>
          <date/>
          <description/>
          <mediaObjectid>6c3a400d8269455795c586138beb32db1d597e5b341f445f823807ab538f436c</mediaObjectid>
          <previewImagePath>http://193.191.148.146/BEELDBANK-MGMT/6c3a400d8269455795c586138beb32db1d597e5b341f445f823807ab538f436c/browse.jpg</previewImagePath>
          <thumbnailImagePath>http://193.191.148.146/BEELDBANK-MGMT/6c3a400d8269455795c586138beb32db1d597e5b341f445f823807ab538f436c/browse-thumb.jpg</thumbnailImagePath>
          <title>Haven 0</title>
          <type>Image</type>
        </mediaDataList>
        <mediaDataList>
          <date/>
          <description/>
          <mediaObjectid>5955c31e5bf3476f8a234a003c726a56cdce4b1af3ef4d17b242bc3904bedfd</mediaObjectid>
          <previewImagePath>http://193.191.148.146/BEELDBANK-MGMT/5955c31e5bf3476f8a234a003c726a56cdce4b1af3ef4d17b242bc3904bedfd/browse.jpg</previewImagePath>
          <thumbnailImagePath>http://193.191.148.146/BEELDBANK-MGMT/5955c31e5bf3476f8a234a003c726a56cdce4b1af3ef4d17b242bc3904bedfd/browse-thumb.jpg</thumbnailImagePath>
          <title>Haven 1</title>
          <type>Image</type>
        </mediaDataList>
        <mediaDataList>
          <date>2012:05:25 10:56:24</date>
          <description/>
          <mediaObjectid>0908c7fa34e6467fb3af70b5460f3a1b0765d9df698460888950c6f4036e43c</mediaObjectid>
          <previewImagePath>http://193.191.148.146/BEELDBANK-MGMT/0908c7fa34e6467fb3af70b5460f3a1b0765d9df698460888950c6f4036e43c/browse.jpg</previewImagePath>
          <thumbnailImagePath>http://193.191.148.146/BEELDBANK-MGMT/0908c7fa34e6467fb3af70b5460f3a1b0765d9df698460888950c6f4036e43c/browse-thumb.jpg</thumbnailImagePath>
          <title>138 DC De Meere.jpg</title>
          <type>Image</type>
        </mediaDataList>
        <mediaDataList>
          <date>2012:05:25 10:49:26</date>
          <description/>
          <mediaObjectid>444ddb21d2aa49ce868f05612e0e92d0d0251266b46142cca6f621d305ccc97b</mediaObjectid>
          <previewImagePath>http://193.191.148.146/BEELDBANK-MGMT/444ddb21d2aa49ce868f05612e0e92d0d0251266b46142cca6f621d305ccc97b/browse.jpg</previewImagePath>
          <thumbnailImagePath>http://193.191.148.146/BEELDBANK-MGMT/444ddb21d2aa49ce868f05612e0e92d0d0251266b46142cca6f621d305ccc97b/browse-thumb.jpg</thumbnailImagePath>
          <title>16 DC De Meere.jpg</title>
          <type>Image</type>
        </mediaDataList>
        <mediaDataList>
          <date>2009:08:18 14:41:37</date>
          <description/>
          <mediaObjectid>761d02dd138f4f3da18b183414fbafd228e923663e84194a2ff8cbe47ae982f</mediaObjectid>
          <previewImagePath>http://193.191.148.146/BEELDBANK-MGMT/761d02dd138f4f3da18b183414fbafd228e923663e84194a2ff8cbe47ae982f/browse.jpg</previewImagePath>
          <thumbnailImagePath>http://193.191.148.146/BEELDBANK-MGMT/761d02dd138f4f3da18b183414fbafd228e923663e84194a2ff8cbe47ae982f/browse-thumb.jpg</thumbnailImagePath>
          <title>Carolus Borromeuskerk (1) © Antwerpen Toerisme & Congres.JPG</title>
          <type>Image</type>
        </mediaDataList>
        <startIndex>0</startIndex>
        <totalNrOfResults>823</totalNrOfResults>
      </return>
    </ns2:searchResponse>
  </S:Body>
</S:Envelope>
```

4.2 More complex example: getting collections

This example shows how collections can be retrieved:

- First we search for objects called: "Sinterklaas Kortrijk" (exact match)

```
<query>"Sinterklaas Kortrijk"</query>
```

- We receive results, and one of the results is of <type>Collection</type>, and with <mediaObjectid>999dca7b4542469ab42fdd8b78e1dc240cb59959aac84eba8a6a66845a221d73</mediaObjectid>

- We can get the objects in that Collection with the following query:

```
<query>Collection:999dca7b4542469ab42fdd8b78e1dc240cb59959aac84eba8a6a66845a221d73*</query>
```

- Those objects can be again images, videos, collections, ...

4.3 Kortrijk live example

The API runs at:

- <https://beeldbank.kortrijk.be:444>
- login: tinkertouch
- password: graven
- organization: kortrijk



For starting at the root collection Gravenkapel, we use
Collection:ad74ee9e6175467caf384692ec0f1995e0191731528f4198bfe0feaffc558e71* as
query:

```
curl -k -d '<?xml version="1.0" ?><S:Envelope
xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"><S:Body><ns2:search
xmlns:ns2="http://ws.api.mediahaven.zeticon.com/"><login>tinkertouch</login><password>g
raven</password><organisation>kortrijk</organisation><query>Collection:ad74ee9e6175467
caf384692ec0f1995e0191731528f4198bfe0feaffc558e71*</query><startIndex>0</startIndex>
<nrOfResults>5</nrOfResults></ns2:search></S:Body></S:Envelope>' -H "Content-type:
text/xml" https://beeldbank.kortrijk.be:444/mediahaven-public-ws/Search
```

This returns 2 results, a collection Gravenkapel_graven and a collection
Gravenkapel_geschiedenis.