

Twingly Api

Analytics API



Author: Oskar Skoog (oskar@twingly.com)
Rev: 1.0
2012-03-14

INTRODUCTION

Analytics API (<http://api.twingly.com/analytics>) is an XML over HTTP API that enables machine access to Twingly's blog search index. It is the API to use when querying for commercial purposes. Given an API key that grants access to Analytics, search results can be fetched with the same query language that is used on twingly.com.

When querying through the Analytics API, the result is given in XML. Walkthroughs of how to build queries and what is included in the response is given below. A major difference between results fetched from twingly.com and results fetched from the Analytics API is that the full text of the post is returned in the latter case.

QUERYING

The query language used by the Analytics API is exactly the same as on twingly.com. A reference for how to build queries can be found here: <http://www.twingly.com/help#search>

You can find a help page for the Analytics API at <http://api.twingly.com/analytics>. The form in that page (shown below) can be used as a help for building queries. The service is located at <http://api.twingly.com/analytics/Analytics.ashx> and the query is built by adding query parameters to the URL.

The screenshot shows a web form with a black background and white text. It contains several input fields and labels:

- API Key (required)**: A white text input field.
- Search pattern (required)**: A white text input field with an example: *eg. Primelabs AND Twingly*.
- Document Language**: A white text input field with an example: *A ISO twoletter languagecode, eg. sv*.
- Timestamp**: A white text input field with an example: *eg. 2007-05-01 12:00:00*.
- Timestamp To**: A white text input field with an example: *eg. 2007-05-01 13:00:00*.
- Output version**: A white text input field containing the number **2**, followed by the text: *0 - old format, 1 - current version (no blogrank), 2 - current version (with blogrank)*.
- Search**: A white button with the text "Search" in black.

What follows below is an explanation of the different fields and query parameters. Remember that parameter values must be URL encoded. The examples below are not encoded.

API KEY

When searching, an API key must be provided for authentication.

Parameter name: *key*

SEARCH PATTERN

This is the actual query. As mentioned earlier in this document, the query language is the same as on twingly.com and a reference can be found here: <http://www.twingly.com/help#search> .

Parameter name: *searchpattern*

DOCUMENT LANGUAGE

This field is used to restrict the search to a desired language. So if the search should only return results written in English, the string *en* should be used.

NOTE! An API key may grant access to one language, several languages or all languages. As an example, if the key is restricted to Swedish, this will be enforced even though no language code is given.

Parameter name: *documentlang*

TIMESTAMP

The *Timestamp* field is used to limit the search to posts that were published after the given timestamp.

Parameter name: *ts*

Parameter value (date and time): Example: *2007-05-01 14:45:58*

TIMESTAMP TO

The *Timestamp to* field is used to restrict the search to posts published before the given timestamp.

Parameter name: *tsTo*

Parameter value (date and time): Example: *2007-05-01 14:45:58*

OUTPUT VERSION

This dropdown can be used to get different version of the result. Older versions are supported for backward compatibility. When starting to use the service the latest version should always be used.

Parameter name: *xmloutputversion*

Parameter value (Integer):

- Old format (default if parameter is missing): 0
- Current format (without blogrank): 1
- Current format (with blogrank): 2

As marked in the form, the only two required fields are the *API key* and *Search pattern*. All other fields are optional and except for Output version, all of them can be expressed within the actual search query as well.

THE RESULT

When querying through the Analytics API, the search result will be returned as XML. Below is an explanation of the elements and their attributes.

The structure of the XML document returned looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<twinglydata
  numberOfMatchesReturned=""
  secondsElapsed=""
  numberOfMatchesTotal="">
  <post contentType="">
    <url></url>
    <title></title>
    <summary></summary>
    <languageCode></languageCode>
    <published></published>
    <indexed></indexed>
    <blogUrl></blogUrl>
    <blogName></blogName>
    <authority></authority>
    <blogRank></blogRank>
    <matchedLinks>
      <link></link>
    </matchedLinks>
    <tags>
      <tag></tag>
    </tags>
  </post>
</twinglydata>
```

<TWINGLYDATA>

The `<twinglydata>` element is the root element. It has three attributes:

- `numberOfMatchesReturned` – The number of matches returned in the xml result (the maximum number of results returned for a query is 1000).
- `secondsElapsed` – The time it took to perform the search.
- `numberOfMatchesTotal` – The total number of matches that the search rendered.

The two attributes `numberOfMatchesReturned` and `numberOfMatchesTotal` can be used to figure out if pagination is required to get the full search result. The pagination is then done by using the `timestamp` and `timestamp to` fields in the form mentioned above. Using these two fields a time window is created that can be moved over the full result set.

<POST>

The `<post>` is the root element for a post. The type of the post is given by the following attribute:

- `contentType` - The type of the post. Currently the only available type is blog.

<URL>

The `<url>` element contains the URL to the post.

<TITLE>

The `<title>` element contains the title of the post.

<SUMMARY>

The `<summary>` element contains the summary of the post. Through Analytics API, this contains the full content of the blog post

<LANGUAGECODE>

The `<languageCode>` element contains the ISO language code that represents the language that the post was written in.

<PUBLISHED>

The `<published>` element contains the time when the post was published. The time is given in UTC.

<INDEXED>

The `<indexed>` element contains the time when the post was indexed by Twingly. The time is given in UTC.

<BLOGURL>

The `<blogUrl>` element contains the URL to the blog.

<BLOGNAME>

The `<blogName>` element contains the name of the blog.

<AUTHORITY>

The `<authority>` element contains a value indicating how authoritative the blog is. The value is an increasing value, where a higher value means more authoritative.

<BLOGRANK>

The `<blogRank>` element contains the *BlogRank* value for the blog. The *BlogRank* value is a value from 1 to 10 indicating how important or influential the blog is.

<MATCHEDLINKS>

The `<matchedLinks>` element will be included if the *Show Matched Links* feature is turned on. It will only have values if a *link:* search is performed. The purpose of this field is to show the exact page that the post has linked to.

So if the search query looks like this:

link:twingly.com

then the `<matchedLinks>` element will contain one or more `<link>` elements containing the full URLs to twingly.com included in the post.

<TAGS>

The `<tags>` element contains zero or more `<tag>` elements that contain any tags specified by the author of the post.