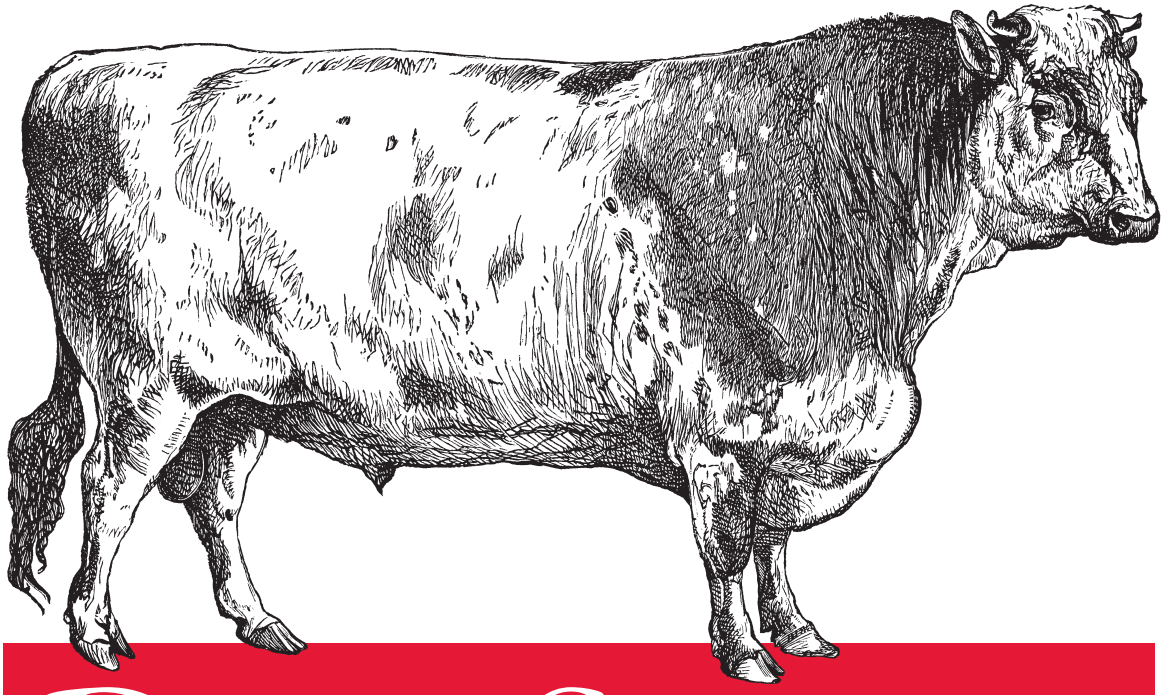
A Guide to Public Data



Data Source Handbook

easy computing

O'REILLY®

Pete Warden


easy
computing

Data Source Handbook

Yahoo! Music

This service lets you query both a large back catalog of music and information on the current charts using YQL. You're limited to 5,000 queries a day, and the information you get back on artists isn't very extensive, but it's simple to access, as it doesn't require any authentication:

```
curl "http://query.yahooapis.com/v1/public/yql?\
q=select%20*%20from%20music.artist.search%20where%20keyword%3D'Rihanna'&format=json"

{"query":{"count":"4","created":"2011-01-09T16:45:20Z","lang":"en-US","results":{
  "Artist":[{"catzillaID":"1927869111","flags":"124547","hotzillaID":"1809845326",
 "id":"19712698","name":"Rihanna","rating":"-1","trackCount":"453",
 "url":"http://new.music.yahoo.com/rihanna/",
 "website":"http://rihannanow.com","ItemInfo":{"Relevancy":{"index":"465"}}},
...
```

Musicbrainz

This site has assembled a large collection of information on music artists and works, and it has made the results available for download under an open license. It's reusable under a mixture of public domain and Creative Commons licensing, depending on the attributes you're looking at.

You can also access the same data through the online REST/XML API, where you can look up artists and works and get back quite a lot of information, not only about the people and albums, but also about their relationships to one another:

```
curl "http://musicbrainz.org/ws/1/artist/?type=xml&name=Tori+Amos"

<?xml version="1.0" encoding="UTF-8"?>
...
<artist type="Person" id="c0b2500e-0cef-4130-869d-732b23ed9df5" ext:score="100">
<name>Tori Amos</name><sort-name>Amos, Tori</sort-name>
<life-span begin="1963-08-22"/></artist>
...
```

The Movie DB

A rival site to the well-established IMDB, Movie DB has an API that gives you access to details on a wide range of movies. The terms of service aren't too constraining, with no requirement that your project be immediately end-user-facing, though Movie DB does discourage caching of data. I couldn't find any information on rate limits:

```
curl "http://api.themoviedb.org/2.1/Movie.search/en/xml/<key>/Transformers"

<?xml version="1.0" encoding="UTF-8"?>
<OpenSearchDescription xmlns:opensearch="http://a9.com/-/spec/opensearch/1.1/">
  <opensearch:Query searchTerms="transformers"/>
  <opensearch:totalResults>8</opensearch:totalResults>
  <movies>
```

```
    <movie>
      <score></score>
      <popularity>3</popularity>
      <translated></translated>
      <adult>false</adult>
      <language>en</language>
      <original_name>Transformers</original_name>
      <name>Transformers</name>
      <alternative_name>The Transformers</alternative_name>
      <type>movie</type>
      <id>1858</id>
      <imdb_id>tt0418279</imdb_id>
      <url>http://www.themoviedb.org/movie/1858</url>
      <votes>61</votes>
      <rating>7.4</rating>
      <certification>PG-13</certification>
      <overview>Young teenager Sam Witwicky becomes involved in the ancient struggle
 between two extraterrestrial factions...
      <released>2007-07-04</released>
...
```

Freebase

Freebase has a large collection of user-contributed information on films, TV shows, and other media. Its coverage is a bit patchy, since it depends on the enthusiasm of volunteers, but no authentication is required to access its REST/JSON interface and it has a flexible query language. You do need to be a little careful crafting your queries, since it appears to restrict you to a single query at a time, so if you accidentally create a long-running one by selecting too many rows, you'll be blocked until it completes.

If you want to run your own offline analysis, you can also grab snapshots of entire database as a multigigabyte file: *http://download.freebase.com/datadumps/latest/*:

```
curl "http://api.freebase.com/api/service/mqlread?\
query=\{%22query%22%3A%5B\{%22id%22%3Anull%2C%22name%22%3A%22The+Wicker+Man\
%22%2C%22type%22%3A%22%2Ffilm%2Ffilm%22\}%5D\}"

{ "code": "/api/status/ok",
  "result": [
    {
      "id": "/en/the_wicker_man",
      "name": "The Wicker Man",
      "type": "/film/film"
    },
    {
      "id": "/en/the_wicker_man_2006",
      "name": "The Wicker Man",
      "type": "/film/film"
    }
  ],
  "status": "200 OK",
  "transaction_id": "[illegible]2011-01-09T17:35:11Z;0012"
}
```