



Engineering

[Back](#)

WOEIDs in Twitter's Trends

By

[raffikorian](#) 🧑

Thursday, 4 February 2010

How do you represent a “place”? That’s what we were wondering when we were putting together our API for [Trends on Twitter](http://blog.twitter.com/2010/01/now-trending-local-trends.html) (<http://blog.twitter.com/2010/01/now-trending-local-trends.html>). We needed a way to represent a place in a permanent and language-independent way - we didn’t want to be caught in using an identifier that may change over time, and we didn’t want to be caught in internationalization issues. Where we landed was on using Yahoo!’s Where On Earth IDs, or WOEIDs, as our representation.

[WOEID](http://developer.yahoo.com/geo/geoplanet/guide/concepts.html) (<http://developer.yahoo.com/geo/geoplanet/guide/concepts.html>)s are 32-bit identifiers that are “unique and non-repetitive” — if a location is assigned a WOEID, the WOEID assigned is never changed, and that particular WOEID is never given to another location. In addition, a WOEID has certain properties such as an implied hierarchy (a particular city is in a particular state, for example), and a taxonomy of categories (there is a “language” to categorize something as a “town” or a “suburb”). Finally, there is a standard endpoint to query to get more information about a place. A program that wanted to get data about San Francisco, CA would first determine that it has a WOEID of 2487956, and with that query <http://where.yahooapis.com/v1/place/2487956> (<http://where.yahooapis.com/v1/place/2487956>).

What this all means is that our Trends API is now interoperable with anybody else who is building a system on top of WOEIDs — you could easily mash-up [Flickr’s places](http://www.flickr.com/places/) (<http://www.flickr.com/places/>), with our Trend locations, for example. If you want to give something like that a try, check out the [trends/available](http://apiwiki.twitter.com/Twitter-REST-API-Method%3A-trends-available) (<http://apiwiki.twitter.com/Twitter-REST-API-Method%3A-trends-available>) endpoint, as that will

let you know what WOEIDs we are exposing trending information for. With those WOEIDs, you can then hit up trends/location (<http://apiwiki.twitter.com/Twitter-REST-API-Method%3A-trends-location>) to get the actual trending data.

There are two niceties to the API: pass in a latitude/longitude when querying trends/available, and remember that there is a hierarchy of WOEIDs. If you pass in a lat and a long parameter to trends/available, then all the locations that are returned are sorted by their haversine distance (http://en.wikipedia.org/wiki/Haversine_formula), from the coordinate passed in. Application developers can use this to help get trends from “around where you are”.

And second, like I mentioned above, WOEIDs form a hierarchy (that’s mostly correct). Here is the hierarchy of the locations that we have as of today:

```

1 ("Terra")
|---- 23424775 ("Canada" - Country)
|---- 23424803 ("Ireland" - Country)
|---- 23424975 ("United Kingdom" - Country)
| \---- 24554868 ("England" - State)
|   \---- 23416974 ("Greater London" - County)
|     \---- 44418 ("London" - Town)
|---- 23424900 ("Mexico" - Country)
|---- 23424768 ("Brazil" - Country)
| \---- 2344868 ("Sao Paulo" - State)
|   \---- 12582314 ("São Paulo" - County)
|     \---- 455827 ("Sao Paulo" - Town)
\---- 23424977 ("United States" - Country)
|---- 2347572 ("Illinois" - State)
| \---- 12588093 ("Cook" - County)
|   \---- 2379574 ("Chicago" - Town)
|---- 2347567 ("District of Columbia" - State)
| \---- 12587802 ("District of Columbia" - County)
|   \---- 2514815 ("Washington" - Town)
|---- 2347606 ("Washington" - State)
| \---- 12590456 ("King" - County)
|   \---- 2490383 ("Seattle" - Town)
|---- 2347579 ("Maryland" - State)
| \---- 12588679 ("Baltimore City" - County)
|   \---- 2358820 ("Baltimore" - Town)
|---- 2347563 ("California" - State)
| |---- 12587707 ("San Francisco" - County)
| | \---- 2487956 ("San Francisco" - Town)
| \---- 12587688 ("Los Angeles" - County)
|   \---- 2442047 ("Los Angeles" - Town)
|---- 2347580 ("Massachusetts" - State)
| \---- 12588712 ("Suffolk" - County)
|   \---- 2367105 ("Boston" - Town)
|---- 2347591 ("New York" - State)
| \---- 2459115 ("New York" - Town)
|---- 2347569 ("Georgia" - State)
| \---- 12587929 ("Fulton" - County)
|   \---- 2357024 ("Atlanta" - Town)
|---- 2347602 ("Texas" - State)
| |---- 12590226 ("Tarrant" - County)
| | \---- 2406080 ("Fort Worth" - Town)
| |---- 12590107 ("Harris" - County)
| | \---- 2424766 ("Houston" - Town)
| |---- 12590063 ("Dallas" - County)
| | \---- 2388929 ("Dallas" - Town)
| \---- 12590021 ("Bexar" - County)
|   \---- 2487796 ("San Antonio" - Town)
\---- 2347597 ("Pennsylvania" - State)
| \---- 12589778 ("Philadelphia" - County)
|   \---- 2471217 ("Philadelphia" - Town)

```

Right now, even though we don't expose this information using trends/available, you could ask for any of those WOEIDs, and we'll choose the nearest trend location (or locations!) that we have data for.

And, of course, we have a few more things in the pipeline...

—[@raffi](https://twitter.com/intent/user?screen_name=raffi) (https://twitter.com/intent/user?screen_name=raffi).

Share:



Link copied successfully