Making Maps with the CONTENTdm API

CONTENTdm User Group Meeting
Nashville, TN - August 6-7, 2015
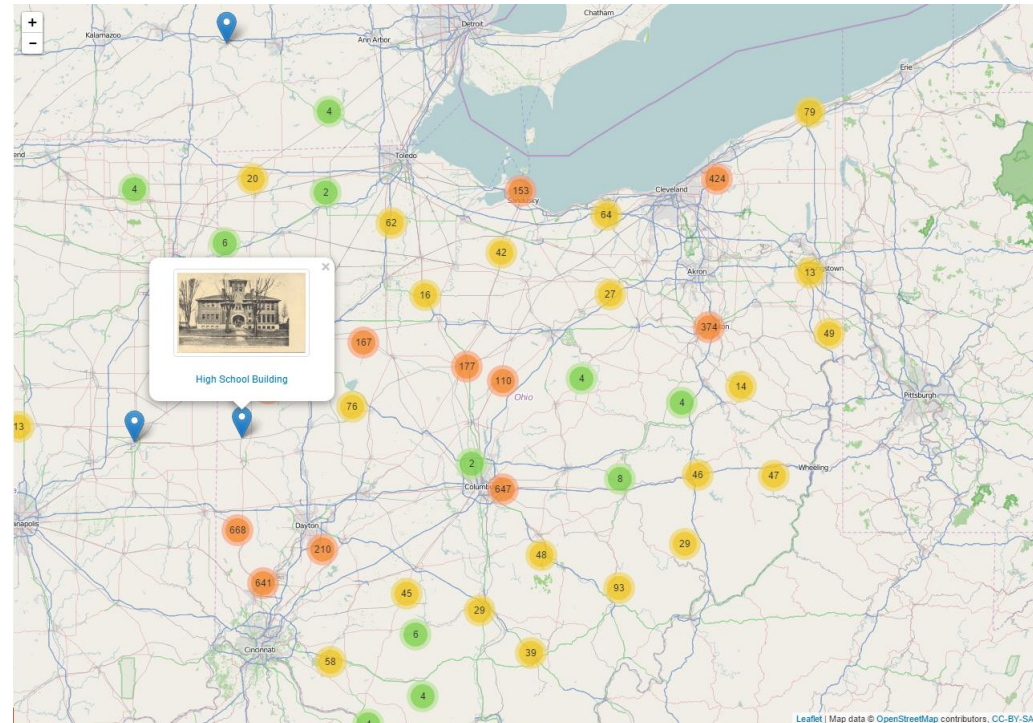
MIAMI UNIVERSITY LIBRARIES

Marcus Ladd
Special Collections Librarian

Easy, free method of displaying a CONTENTdm collection geographically using the API and publically available code

REQUIREMENTS
- SQL database
- Leaflet Javascript Library
- Leaflet.markercluster
- Customized query and map PHP

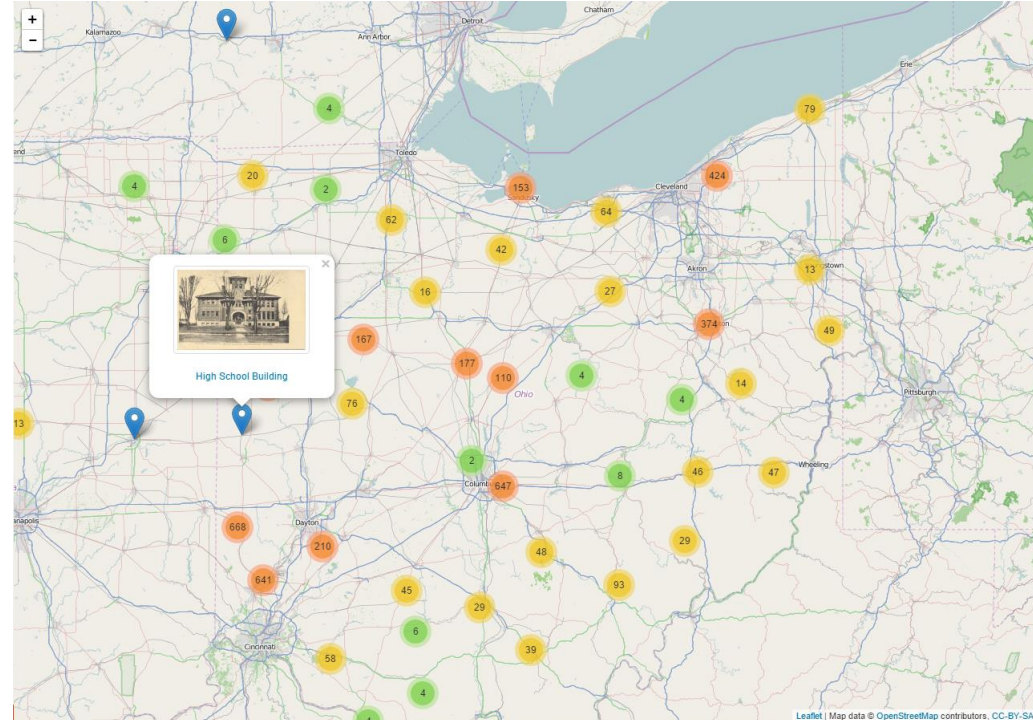Example: http://bit.ly/1HF5Z2R
GitHub: http://bit.ly/1IlyB1d

STRENGTHS

- Free
- Easily customizable
- Handles large collections well
- Mobile interface

WEAKNESSES

- apiquery script must be run to register collection updates
- Street addresses rely on Google Maps API which can be buggy

Example: http://bit.ly/1HF5Z2R
GitHub: http://bit.ly/1IlyB1d
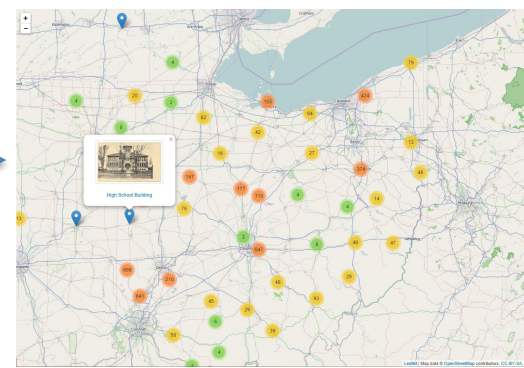
CONTENTdm API allows data to be harvested

Running apiquery.php pulls the data from the CONTENTdm API...

...and stores it in an SQL table

map.php uses the SQL table to generate the final map

WHY THE SQL TABLE?

WHY NOT JUST DRAW STRAIGHT FROM THE CONTENTdm API?

Developed to meet specific needs

- Server storing CONTENTdm data struggled to process a large number of requests
- SQL table led to faster loading
- Uses Google Maps API to create the table, which has a max daily query limit of 2,500

Google Maps has an API that will take address queries and convert them into coordinates, but it is buggy.

Bug #1: Dropping entries
- Even under 2,500 queries, sometimes a query will return empty. This breaks the Javascript used in map.php

- RESOLUTION: before adding new entries to the SQL table, apiquery.php will check for entries with an empty 'coords' field and delete any found

```php
82   $mapsstm = "Select coords FROM " . $dbtable;
83
84   $sqlstm = $pdo->prepare($mapsstm);
85   $sqlstm->execute();
86
87   $response = $sqlstm->fetchAll(PDO::FETCH_COLUMN, 0);
88
89   $cfail = ", ";
90
91   foreach ($response as $ctest) {
92
93       if ($ctest==$cfail) {
94
95           $cstm = "DELETE FROM " . $dbtabl . " WHERE coords=', '";
96
97           $sqlstm = $pdo->prepare($cstm);
98           $sqlstm->execute();
99
100          }
101      }
```

Google Maps has an API that will take address queries and convert them into coordinates, but it is buggy.

Bug #2: Bad coordinates
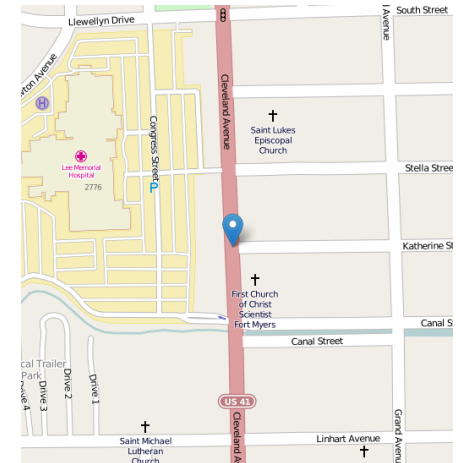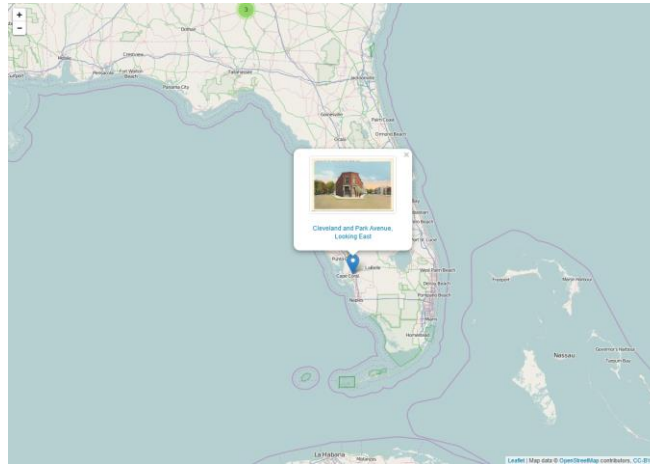- Sometimes the coordinates returned to a street address are inaccurate. No perfect resolution found

- STRATEGIES:
    - Minimize unnecessary information in metadata (e.g. building names)
    - Include postal codes in addresses
    - Constant quality control

Location

Ohio--Amherst

Questions?

GitHub
https://github.com/laddmm/CONTENTdm_Map

Marcus Ladd
laddmm@miamioh.edu