

Leaflet | Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox, Population data © US Census Bureau

Web Mapping Interfaces

Dr Hai Nguyen

Department of Geography and Planning



UNIVERSITY OF
LIVERPOOL

Overview and Structure

- Overview of Web Mapping Stack
- What are slippy maps?
- Interfaces
 - OpenLayers
 - Leaflet
 - D3
- Leaflet Example

Software Solution Stack

- A set of complete software systems that allows building applications.
- Example: LAMP, MEAN...
- Why do we need software stack?
 - Easy to install, maintain and deploy
 - No additional software for most applications
- Types of solution stacks:
 - General: LAMP, MEAN, etc.
 - Specific web application framework: Wordpress, Drupal, Moodle, MediaWiki

Try it yourself

- Create a full software stack in your own computer? => Bitnami.com

bitnami

Applications Cloud Enterprise Support What is Bitnami?

Log in Create Free Account

All

Free Trials

Following

Popular

Recently Updated

Infrastructure

Cloud Tools

Accounting

Analytics

Application Server

WordPress Joomla! Redmine Drupal Cloud9 WAMP Stack

Blog CMS Bug Tracking CMS Developer Tools Infrastructure

Neos Moodle SEO Panel Magento Let's Chat ownCloud

CMS eLearning Analytics e-Commerce Chat Media sharing

The screenshot shows the Bitnami website interface. At the top, there's a navigation bar with links for Applications, Cloud, Enterprise, Support, and What is Bitnami?, along with Log in and Create Free Account buttons. Below the navigation, there's a sidebar with filters: All (selected), Free Trials, Following, Popular, Recently Updated, Infrastructure, Cloud Tools, Accounting, Analytics, and Application Server. The main content area displays a grid of software stacks, each represented by a hexagonal icon and a name. The visible stacks include WordPress (Blog), Joomla! (CMS), Redmine (Bug Tracking), Drupal (CMS), Cloud9 (Developer Tools), WAMP Stack (Infrastructure), Neos (CMS), Moodle (eLearning), SEO Panel (Analytics), Magento (e-Commerce), Let's Chat (Chat), and ownCloud (Media sharing). Each stack has a small curved arrow underneath its icon.

General Web-stacks

- Provide an infrastructure for building web applications.
- Serve content as web-pages
- Typical components:
 - An operating system (Linux, Windows, MacOS)
 - Database Servers: (MySQL, PostgreSQL)
 - Web Servers (Apache/Nginx/Node.js)
 - Server software/technologies: PHP, Python, Node.js, etc.
 - Client-side software/technologies: JavaScript, AngularJS, etc.

A General Web Stack

LAMP – WAMP - MAMP

- L(W|M)AMP: Linux (Windows/MacOS) + Apache + MySQL + PHP



Specific Solution Stacks

- Usually built on-top of a general stack
- Contain a specific framework
- Examples:
 - Content Management System (CMS): Drupal/Joomla/Wordpress
 - E-learning: Moodle
 - Wiki: Media-wiki
 - E-commerce: Magento/Presta Shop

A Web-Mapping Stack?

- To serve maps instead of traditional contents
- Reminder: Web-stack components:
 - A database server
 - A web-server
 - Server-side technologies/languages
 - Client-side technologies/languages

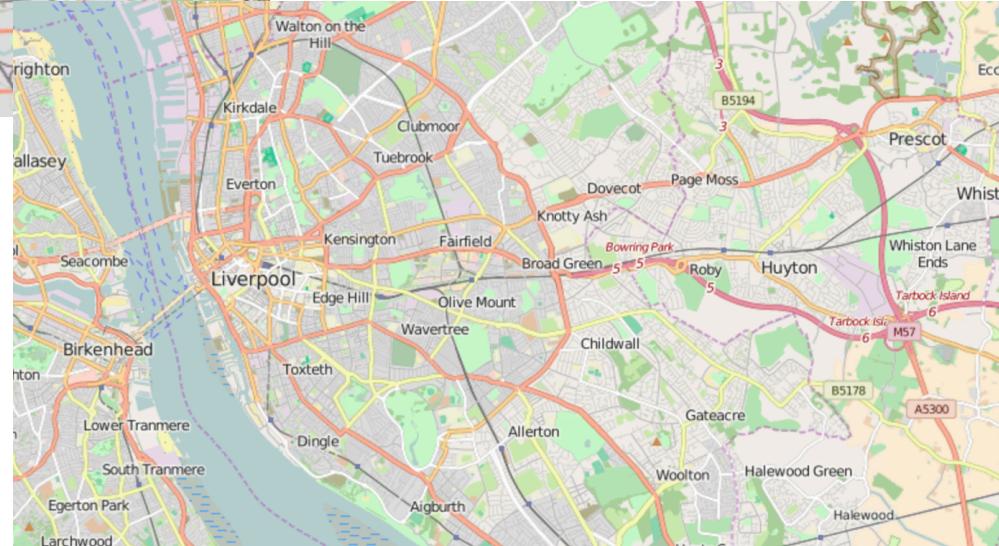
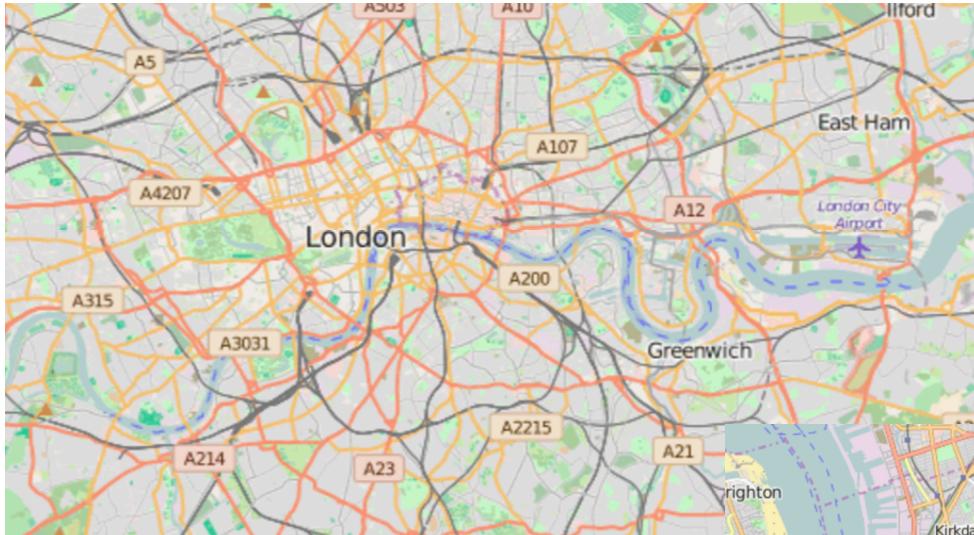
A Web-Mapping Stack?

- A **database server**: needs to store and process geospatial data as well
- A **web-server**: yes but not enough
- Server-side technologies/languages: yes
- Client-side technologies/languages: yes

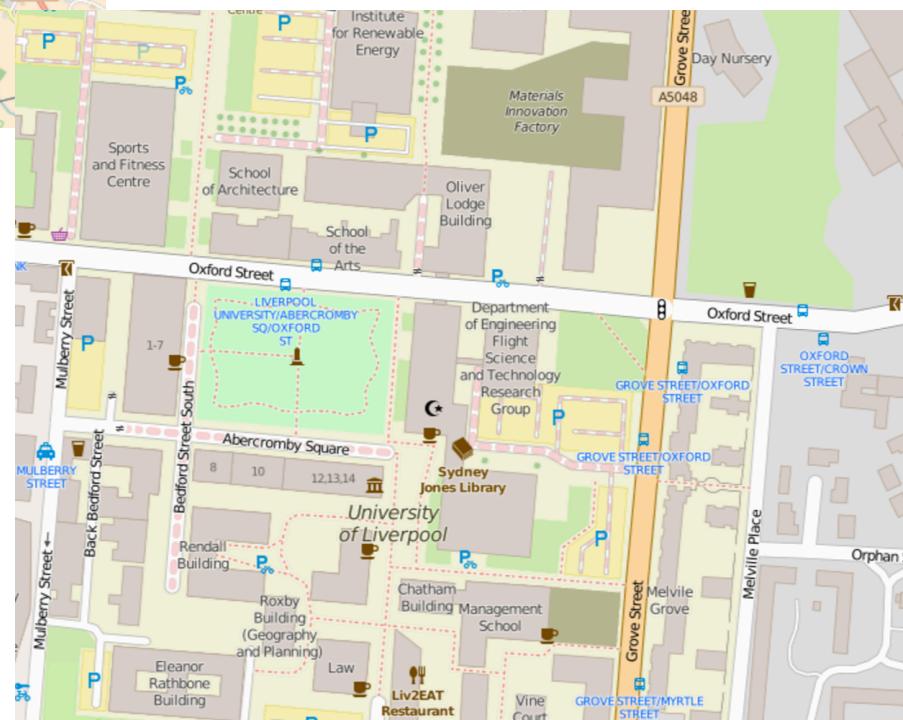
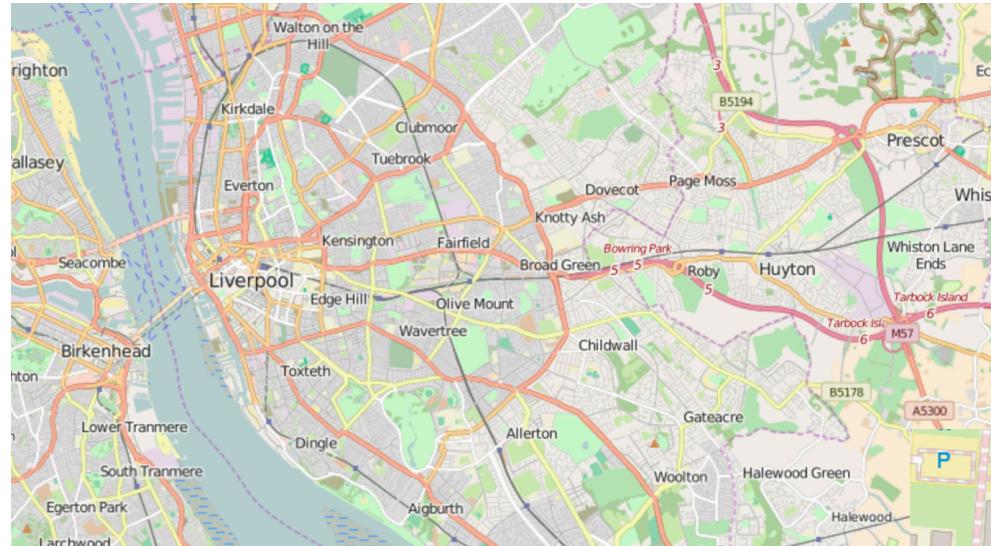
Why can't we use just a web-server?

- Web servers :
 - Serving static, existing images/files
 - Dynamic files: usually text-based contents
- Memory problem:
 - Cannot predict where users might view
 - Multiple zooming levels

Same map, different places (panning)



Same map, different zoom levels



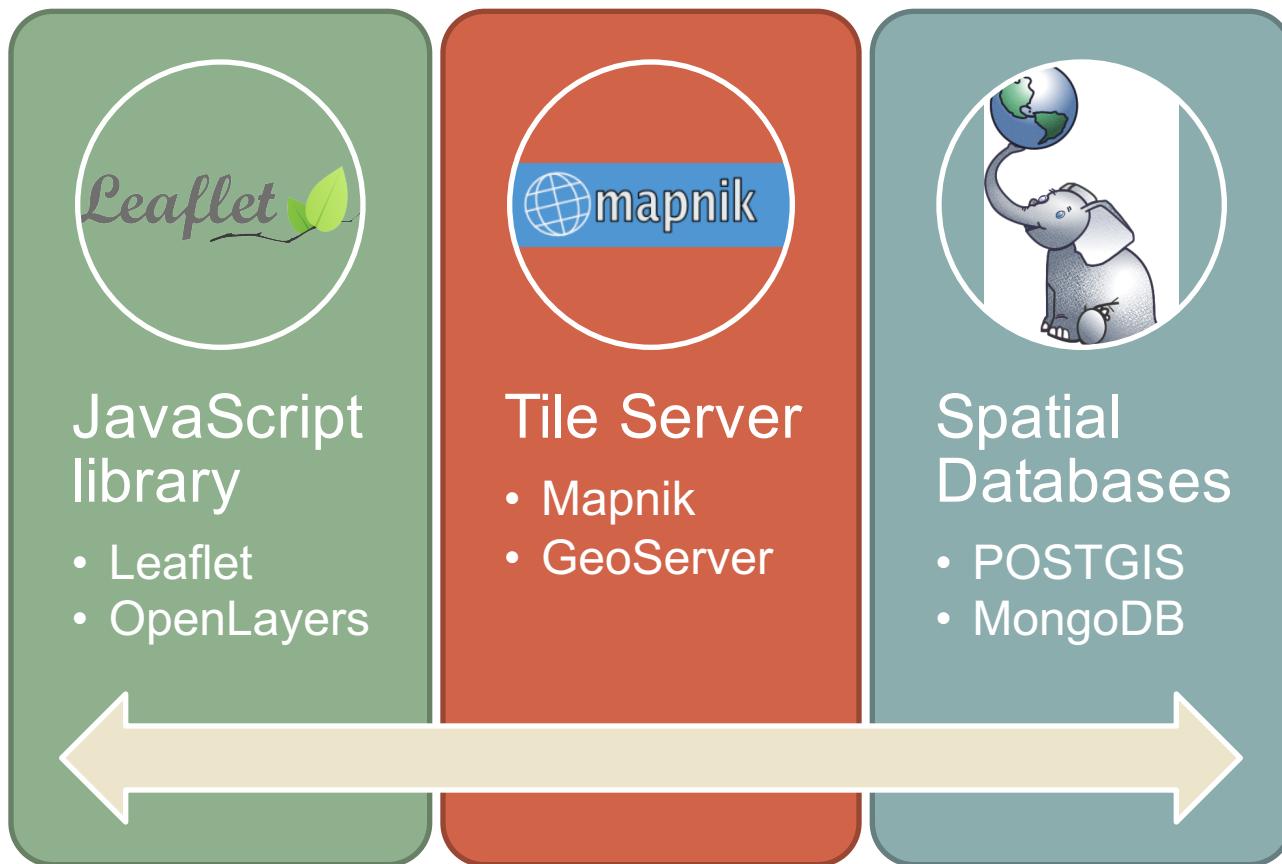
Slippy Maps

- Definition: “Modern web maps which let you zoom and pan around(http://wiki.openstreetmap.org/wiki/Slippy_Map)
- **Slippy Map:**
 - A join of multiple smaller “tiles”
 - Each tile is an image
 - These images are prepared at **a tile server**

Map viewing - behind the scenes:

1. A user interacts to a web map (with panning and zooming)
2. A **JavaScript library** sends corresponding requests to a map server to request “**tiles**”
3. **The map (tile) server** requests the spatial database server required map data.
4. **The Spatial database server** search and returns required map data to the map server.
5. The map server generates and returns requested tiles as **images** (and possibly keep them for later use-caching)
6. The **JavaScript library** renders these tiles and update the map seamlessly

A Web Mapping Stack



A Web-Mapping Stack

- **A geospatial database server:** store (geospatial) data ([Week 7](#))
- **A web-server:** serve HTML documents (webpages)
- **A tile-server:** serve web tiles ([Week 8](#))
- **Server-side technologies/languages:** PHP, Python, Node.js (and extra packages)
- **Client-side technologies/languages:** request and display map tiles on the browser ([this week](#))

Web Map Services

- All server-side components are done for users (data storage, tiles generation, etc.)
- Users only need to request maps (using client-side libraries)
- Examples:
 - Google Maps, Open Street Map, Ordnance Survey Maps: owned/open map data sources/libraries
 - Mapbox, CartoDB: flexible, support multiple 3rd-party client-side libraries

Client-side Web Mapping Libraries

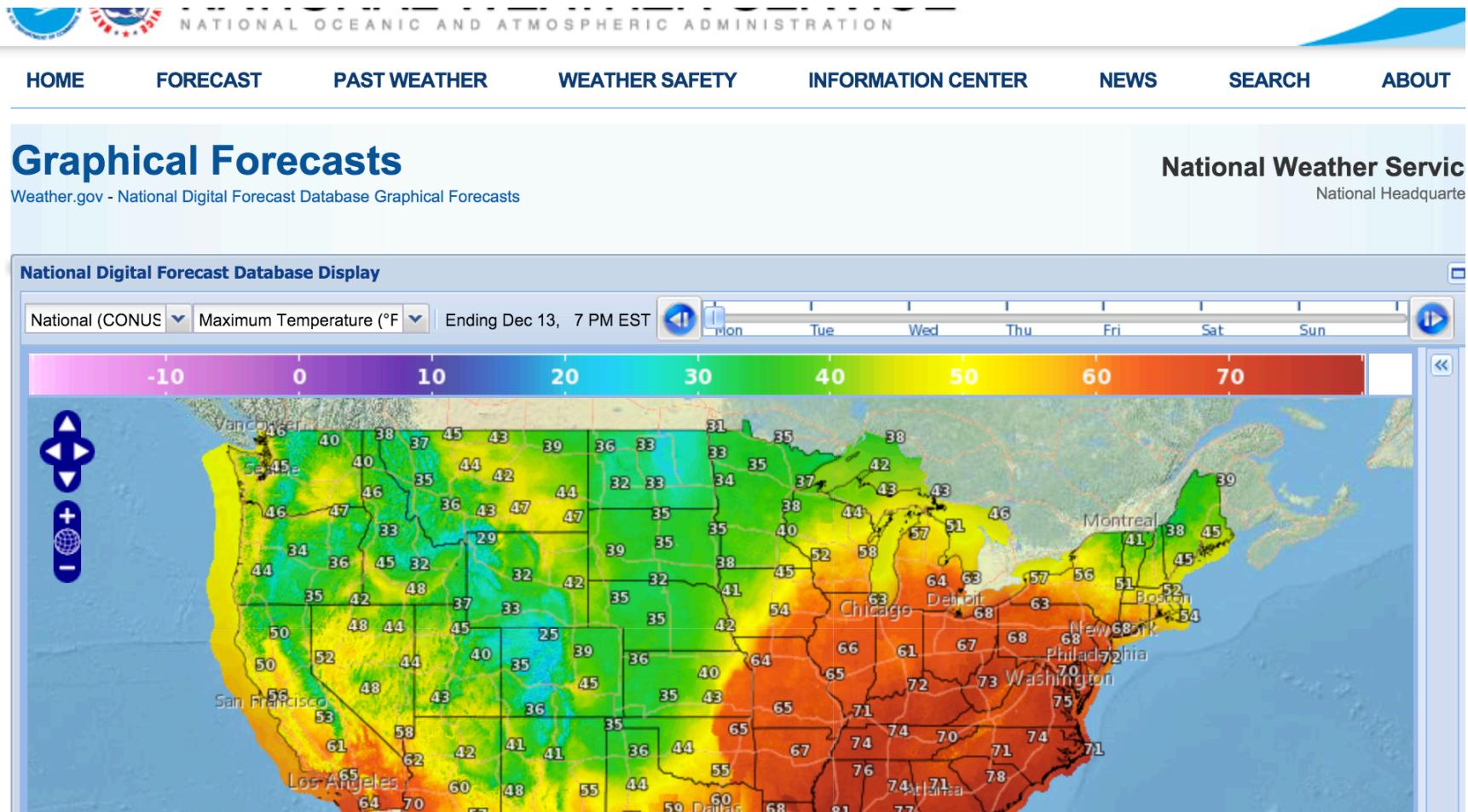
- Operating in client-side (on user's computer/browser)
- Most popular programming language: JavaScript (last week lecture)
- Examples:
 - OpenLayers
 - Leaflet
 - Google Maps API
 - Ordnance Survey Openspace API

OpenLayers

- Open-source JavaScript libraries for displaying map data on browsers
- The most popular mapping library
- Rich features
- Support Raster and Vector maps.
- “Bulky” size: (OL 3 is over 500KB)



digital.weather.gov



Leaflet

- Open-source JavaScript libraries for displaying map data on browsers
- Younger, increasing popularity
- Light-weight (33KB, 10 times lighter than OL3)
- Used by Foursquare, Flickr, etc.

Leaflet 

Foursquare.com Leaflet

ARE I'm looking for... ▾ Current Map View

Coffee near Liverpool

Haven't Been Following Price Open Now Saved Liked

places that your friends and experts love

Sign up with Facebook or Sign up with email

reet Coffee

1 St
Shop • £ £ £ £ • View Menu

Find bacon butty to go and off to Lime St and head home." (4 tips)

Cake was brilliant & the Flat White was excellent." (6 tips)

Search this area

1 26 2 22 28 12 19 17 4 14 10 3 5 7 15 25 23 3 20 16 18 21 24 27 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

D3.js

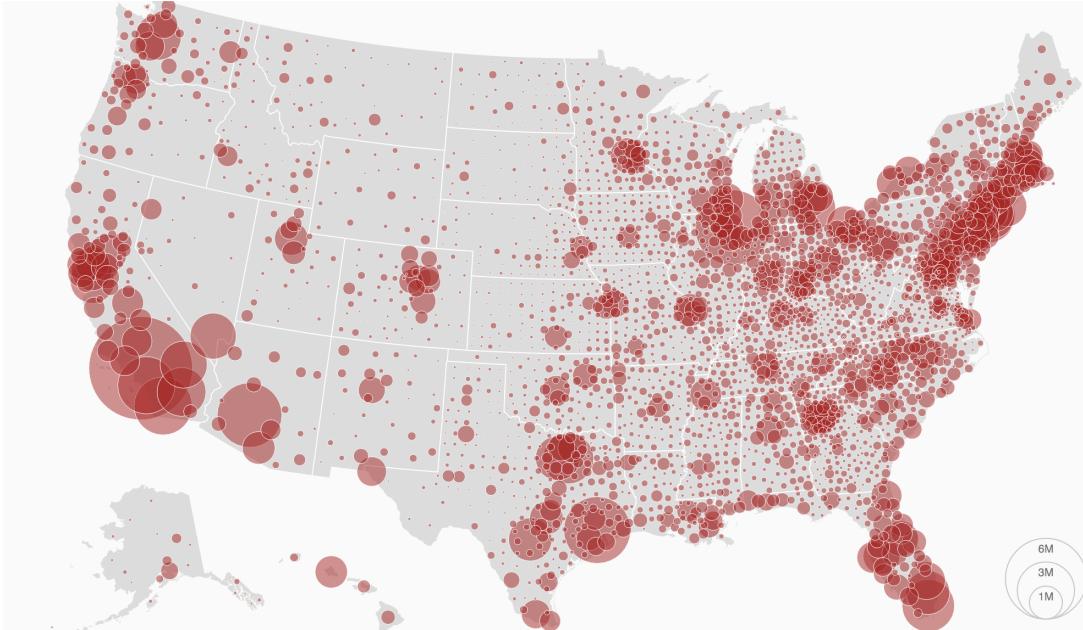
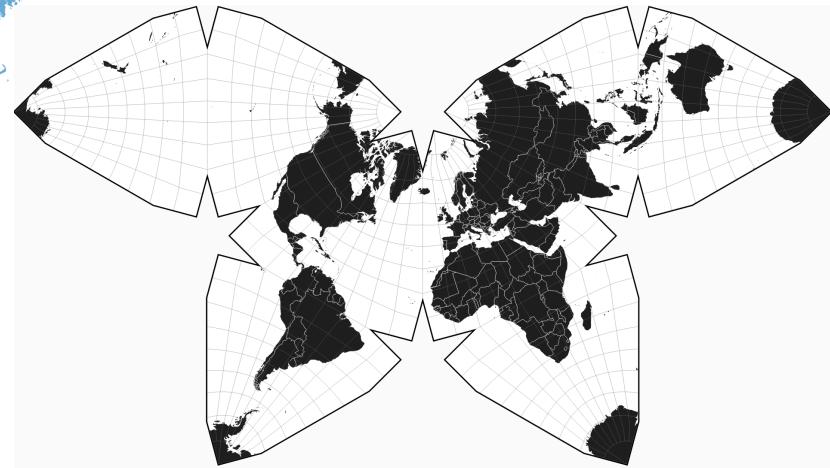
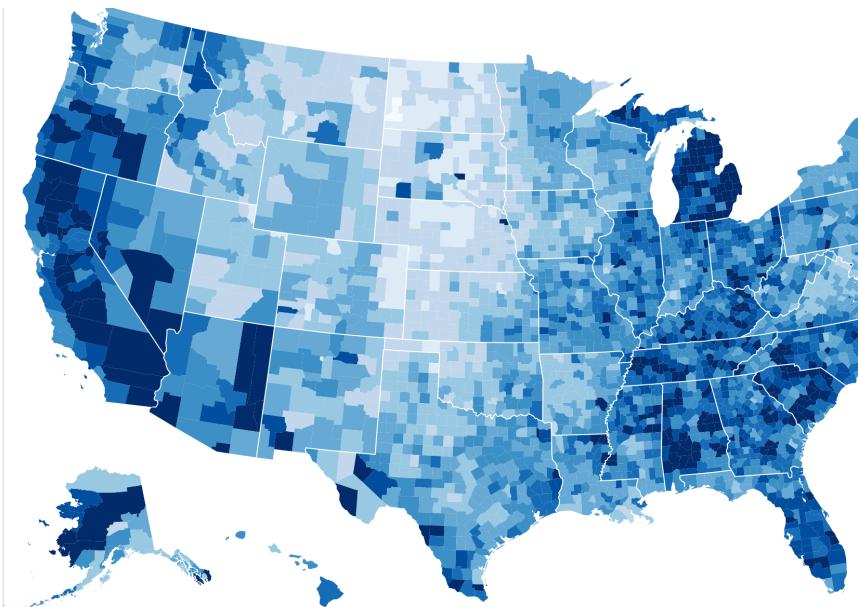
- A JavaScript library for data visualisation
- Visualise data on charts, graph, maps, etc.
- Support GeoJSON format
- Support different interactive maps:
 - Choropleth
 - Heatmap
- Steep learning curve



D3.js



Data-Driven Documents



Top Web Mapping Tech

Top in Mapping · Week beginning Dec 14th 2015

Name	10k	▲ 100k	Million	Entire Web
Google Maps	↑1,142	↑8,912	↑184,979	↑3,648,145
Google Maps API	↑788	↑5,724	↑67,916	↑1,508,938
Bing Maps	↑83	↓326	↓4,079	↓35,437
Map Point	↑82	↓318	↓3,538	↓33,266
MapBox	↑55	↑172	↑1,130	↓16,042
Leaflet	↑43	↑183	↑1,842	↓37,411
ArcGIS	↑32	↑183	↑546	↑2,014
Google Maps Engine	↑18	↑87	↓501	↓19,640
OpenLayers	↑14	↑67	-869	↓6,021
MaxMind	↑12	↑101	↓1,194	↓8,376
MapQuest	↑2	↓16	-1,618	↓3,272
Yandex Maps	↑2	↓17	↓4,909	↑118,816
Google Ditu	↑1	↑1	↓30	↑421
CartoDB	↓1	-6	↑35	↑304
Baidu Maps	-1	↑10	↓1,164	↓30,659

Example: Mapping with Leaflet using JavaScript

1. Prepare your HTML page
2. Write your JavaScript code
 - a. Initialise the map
 - b. Add a tile layer and attach it to the map
 - c. Add extra features
(marker/popup/interactivity)
3. Refresh your page

Step 1: Preparing your Leaflet app

- Include Leaflet CSS file in the head section of your document:

```
<link rel="stylesheet" href="http://cdn.leafletjs.com/leaflet/v0.7.7/leaflet.css" />
```

- Include Leaflet JavaScript file:

```
<script src="http://cdn.leafletjs.com/leaflet/v0.7.7/leaflet.js"></script>
```

- Put a `div` element with a certain `id` where you want your map to be:

```
<div id="map"></div>
```

- Make sure the map container has a defined height, for example by setting it in CSS:

```
#map { height: 180px; }
```

Step 2: Write your JavaScript code

1. Initialise a map
2. Add a tile layer
3. Add markers/popup/interactivity etc.

```
<script>
// initialise a map and set a view (centre coordinates and zoom level)
var map = L.map('mymap').setView([53.405936,-2.9655722], 15);
// add a tilelayer
L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png', {
    attribution: '&copy; <a href="http://osm.org/copyright">OpenStreetMap</a>
                  contributors'
}).addTo(map);
// add a marker and show a popup
L.marker([53.405936,-2.9655722]).addTo(map)
  .bindPopup('University of <b>Liverpool</b>')
  .openPopup();
</script>
```

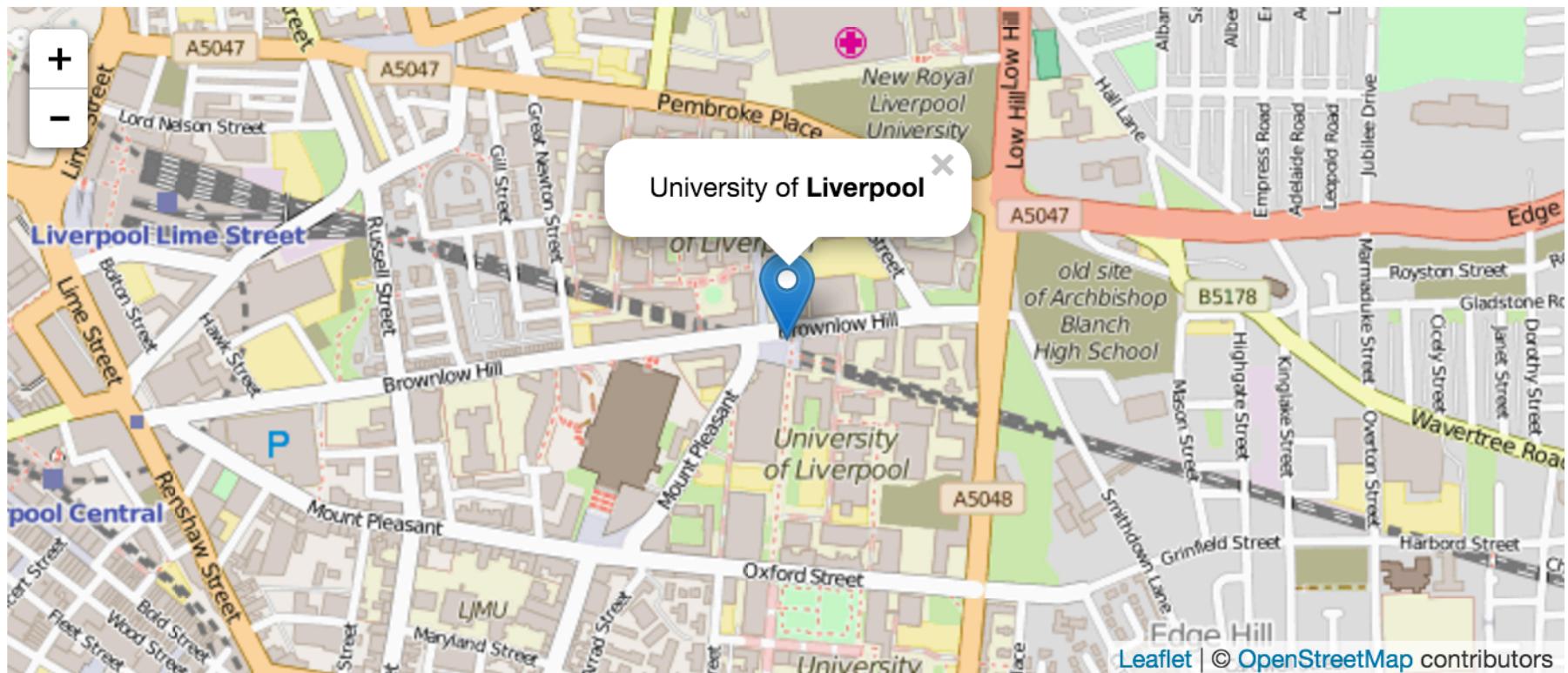
Full HTML source code

```
1 <head>
2
3     <!--STEP 1: loading leaflet css and js files from the CDN--&gt;
4     &lt;link rel="stylesheet" href="http://cdn.leafletjs.com/leaflet/v0.7.7/leaflet.css"
5         /&gt;
6     &lt;script src="http://cdn.leafletjs.com/leaflet/v0.7.7/leaflet.js"&gt;&lt;/script&gt;
7 &lt;/head&gt;
8 &lt;body&gt;
9     &lt;h2&gt;Leaflet Example - Web Mapping and Analysis&lt;/h2&gt;
10    &lt;!-- your map element --&gt;
11    &lt;div id="mymap" style="height:300px;"&gt;&lt;/div&gt;
12 &lt;/body&gt;
13 &lt;script&gt;
14     // STEP 2: Writing JS code
15     // initialise a map and set a view (centre coordinates and zoom level)
16     var map = L.map('mymap').setView([53.405936,-2.9655722], 15);
17     // add a tilelayer
18     L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/{y}.png', {
19         attribution: '&amp;copy; &lt;a href="http://osm.org/copyright"&gt;OpenStreetMap&lt;/a&gt;
20             contributors'
21     }).addTo(map);
22     // add a marker and show a popup
23     L.marker([53.405936,-2.9655722]).addTo(map)
24         .bindPopup('University of &lt;b&gt;Liverpool&lt;/b&gt;')
25         .openPopup();
26 &lt;/script&gt;</pre>
```

Result on a browser



Leaflet Example - Web Mapping and Analysis

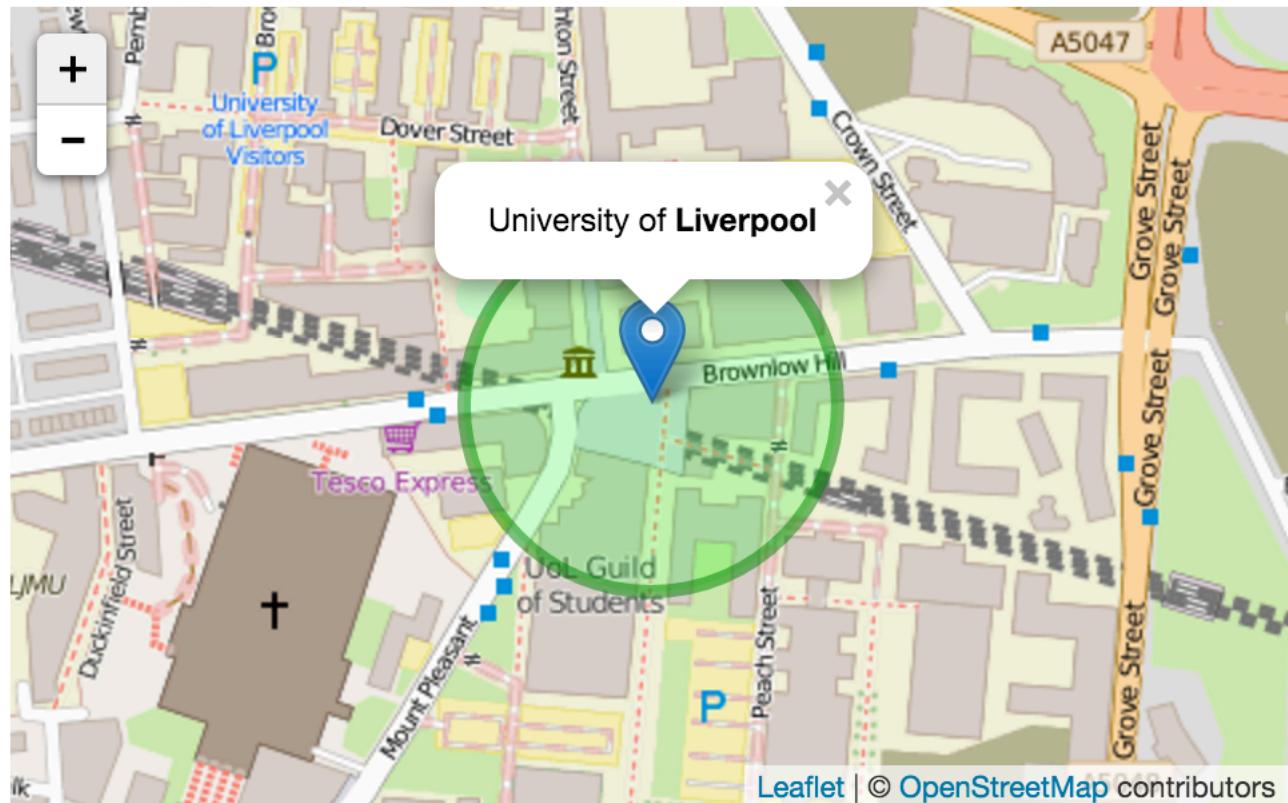


Other things to do with Leaflet

Draw shapes on the map

(<http://leafletjs.com/examples/quick-start.html>)

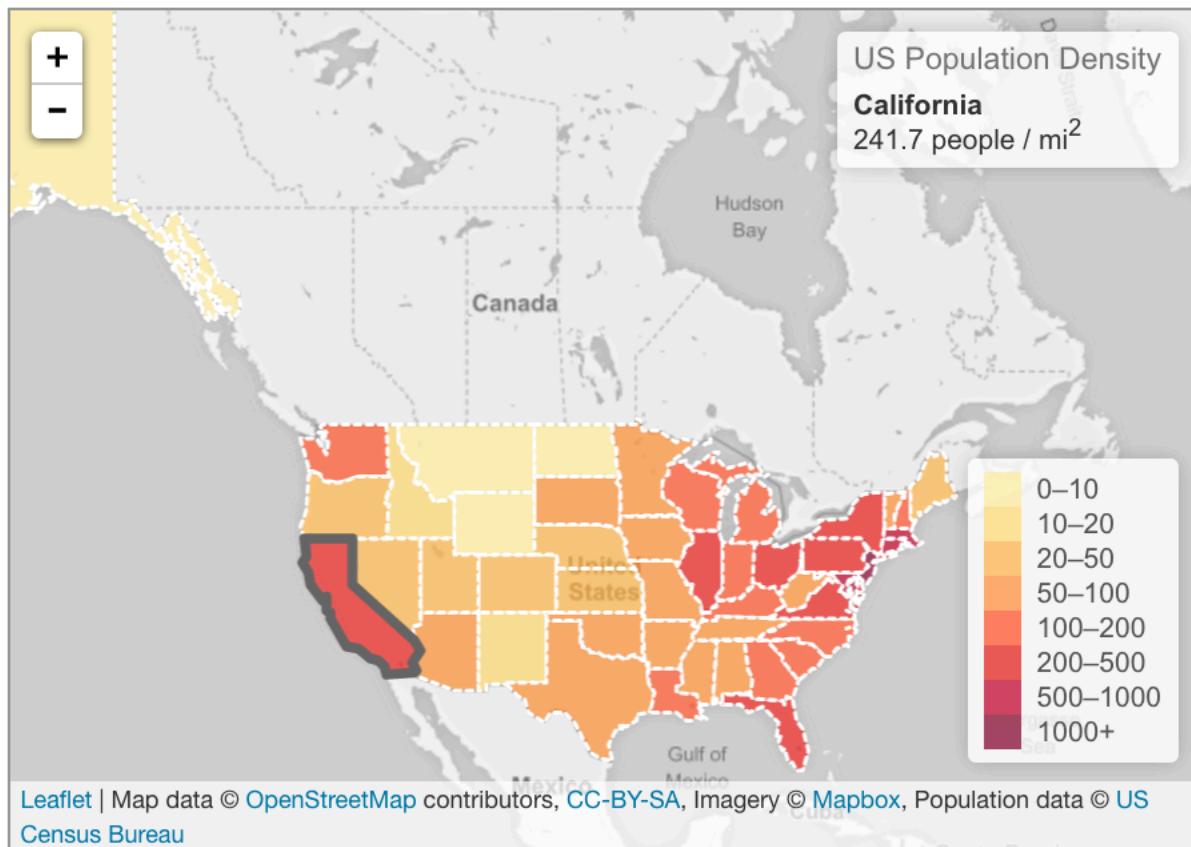
Leaflet Example - Web Mapping and Analysis



Other things to do with Leaflet

Add Interactivity

(<http://leafletjs.com/examples/choropleth.html>)



Can we create Leaflet map with R?

Leaflet for R

Introduction

The Map Widget

Basemaps

Markers

Popups

Lines and Shapes

JSON

Raster Images

Shiny Integration

Colors

Introduction

Leaflet is one of the most popular open-source JavaScript libraries for interactive maps. It's used by websites ranging from [The New York Times](#) and [The Washington Post](#) to [GitHub](#) and [Flickr](#), as well as GIS specialists like [OpenStreetMap](#), [Mapbox](#), and [CartoDB](#).

This R package makes it easy to integrate and control Leaflet maps in R.

Features

- Interactive panning/zooming
- Compose maps using arbitrary combinations of:
 - Map tiles
 - Markers
 - Polygons
 - Lines
 - Popups
 - GeoJSON
- Create maps right from the R console or RStudio
- Embed maps in [knitr/R Markdown](#) documents and [Shiny](#) apps

<https://rstudio.github.io/leaflet>

Practical

- Using Leaflet library with the R programming language
- Explore how to write web-like apps using leaflet package

Many thanks...

