

Mapping in the Cloud: Working with Google Maps and other Mapping Services

Michael P. Peterson

University of Nebraska at Omaha

Outline

1. Coding as Literacy
2. Cloud
3. Cloud Maps
4. Education
5. Future

Mapping in the Cloud

- Maps have always been dependent on tools
 - Tools for measuring world
 - Tools for making maps – printing, photography
- Map making tools are now in the cloud
 - Data, and the programs to process, display the data
- How should the modern map maker be trained with these new tools?

Programming Maps

- 1960s-1980s mapping by computer required knowledge of programming
- By 1990s, standard programs had developed that removed the programming requirement
- Using cloud resources for mapping again requires some programming knowledge

Coding as Literacy

- Formerly relegated to geeks, programming is a hot topic again
 - CodeAcademy and Khan Academy
 - Free online sites for learning how to code
 - new movement effectively argues that programming is a necessary skill that everyone should know, like reading and writing
 - Programming is now being viewed by many as a form of expression, as the “amplification of thinking,” and a necessary skill.

Internet has changed programming

- Application Programmer Interfaces (API)
 - Library of existing routines
- Possibility of using multiple APIs
- Programming has become the knowledge of integrating existing code
 - Languages like JavaScript make it easy to ‘examine’ source code
- Can this type of “Internet programming” be taught to make maps?

API

- Online libraries of functions, freely available
- Tools to acquire, manipulate and display information from a variety of sources
- Facilitate the fusion of data and mapping resources
- Multitude of APIs available for a variety of functions
- Most used: Google Maps API
 - Most used of any API

Teaching APIs

- Mashups taught since 2007
 - Austria, Germany, New Zealand & US
- Upper-division students with varied programming experience, some with none
- Experiences with
 - HTML, JavaScript
 - APIs: Points, lines, areas, layers, mobile browser
 - php, MySql

Cloud



On the ground

Power back-up

Initially: Lead-acid batteries

Then: 10 diesel generators

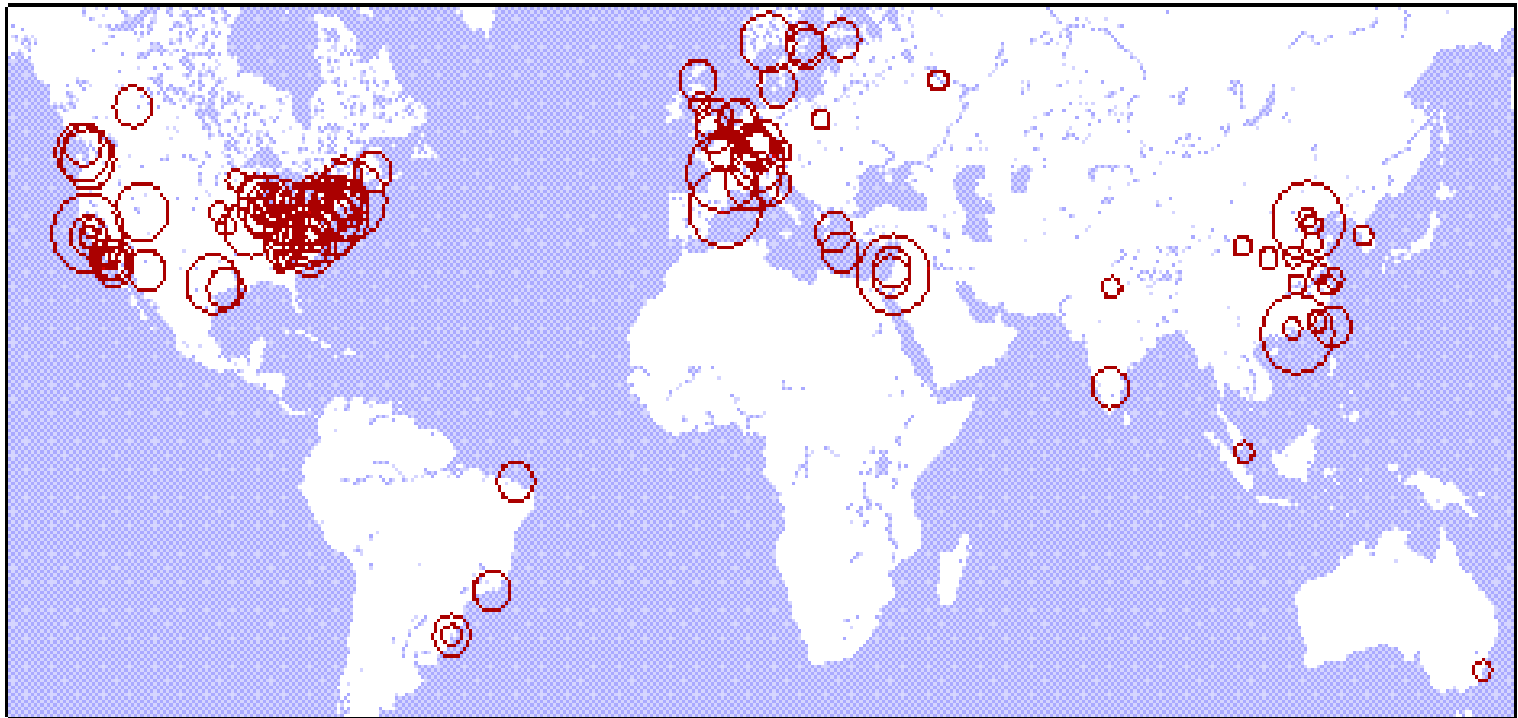




Content Delivery Networks (CDNs)

- Content Delivery Networks (CDN) provide file servers distributed so as to maximize bandwidth
- visitors to a website can be served content from a nearby server
- Big names in the CDN business are Google, Amazon, and Yahoo.
- Data centers are the brains, CDNs are the synapses

Location of Google's content delivery network (CDN)



Cartography

- Two most important technology-driven developments in the past decade:
 - MSPs and MOMM
- MSP – Multi-scale pan-able
 - Tile-based presentation of maps
- MOMM – Me on my Map
 - The reassuring blue dot
 - GPS, cell-phone tower triangulation, wi-fi signal triangulation
 - Most people couldn't find themselves on a paper map

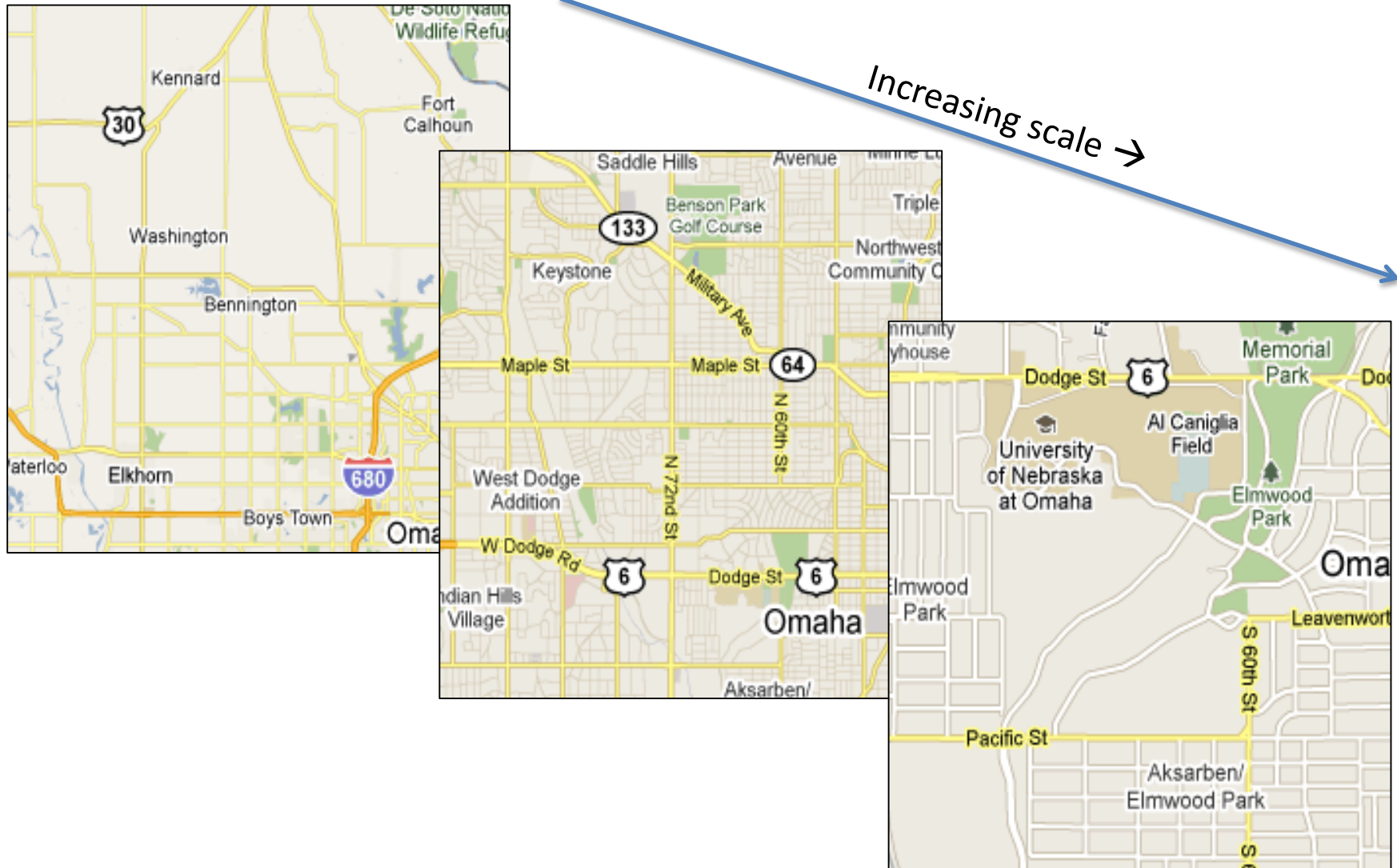
A Google Map Tile



Since 2005

Small
pictures:
256 x 256
pixel png
files

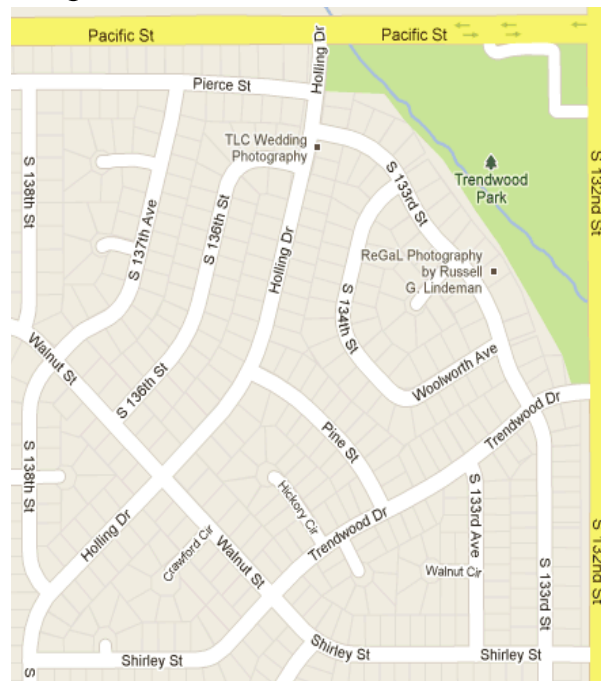
MSP Google Raster – 2005 to 2010



Renderings

- Different ways of symbolizing underlying vector data
- Many would try to limit cartography to this aspect

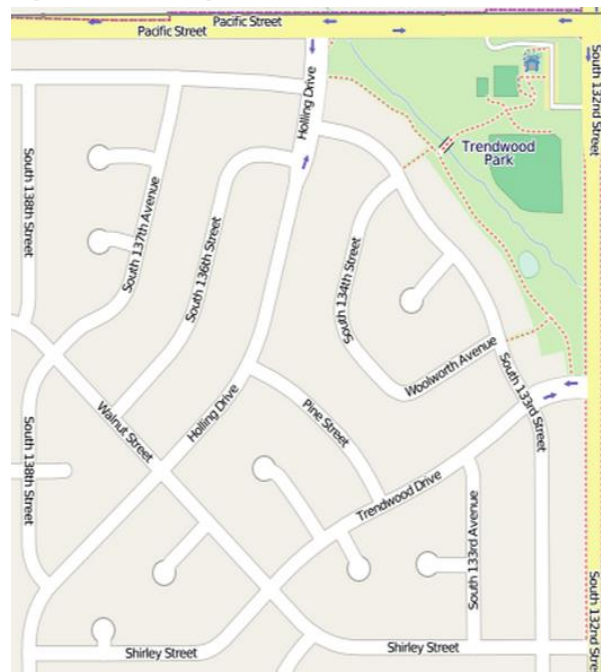
Google



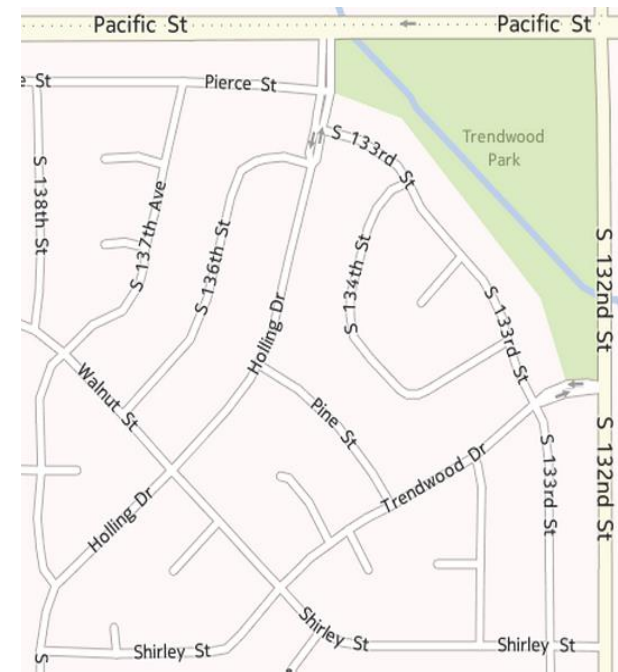
MapQuest



OpenStreetMap

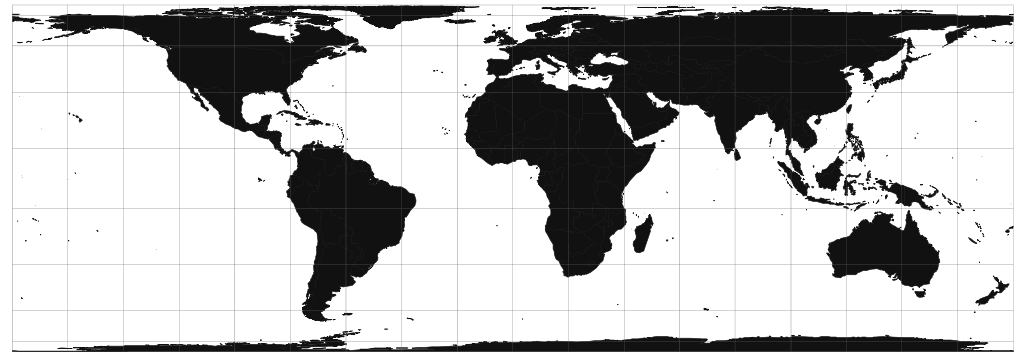
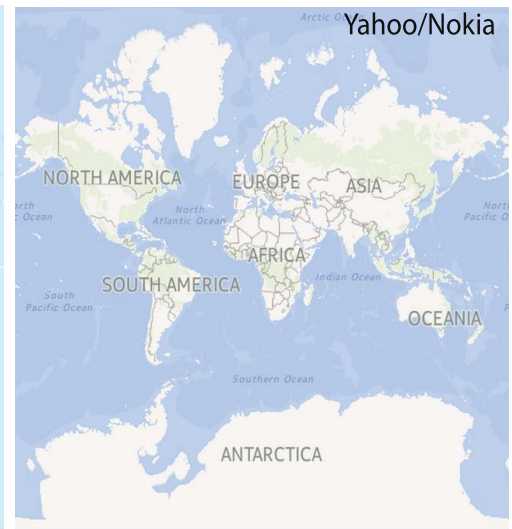
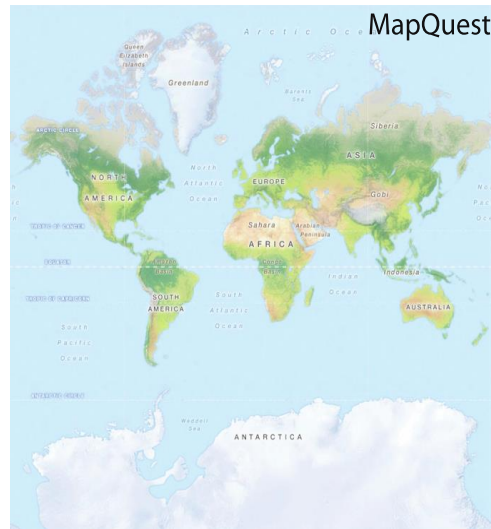


Yahoo/Nokia

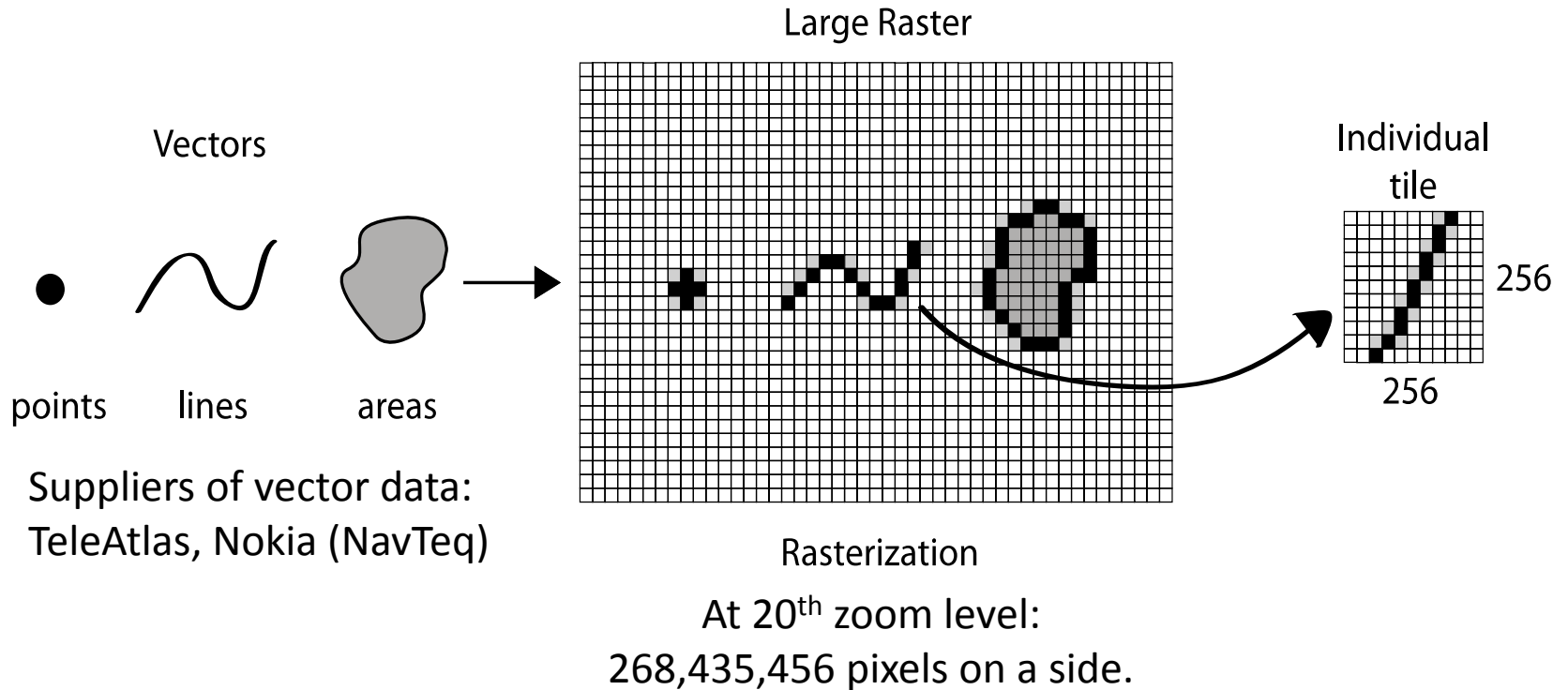


Web Mercator

- Mercator projection used by all major map providers
- Severely distorts area
- Greenland takes as much space on the map as Africa, when in reality Africa's area is 14 times greater



Tiling



- Raster tiles are still used by Bing, Yahoo (Nokia), MapQuest, OpenStreetMap, Baidu
- All tile their maps exactly the same way.
- Same projection. Same divisions. Tiles are interchangeable.

Tiles and cost

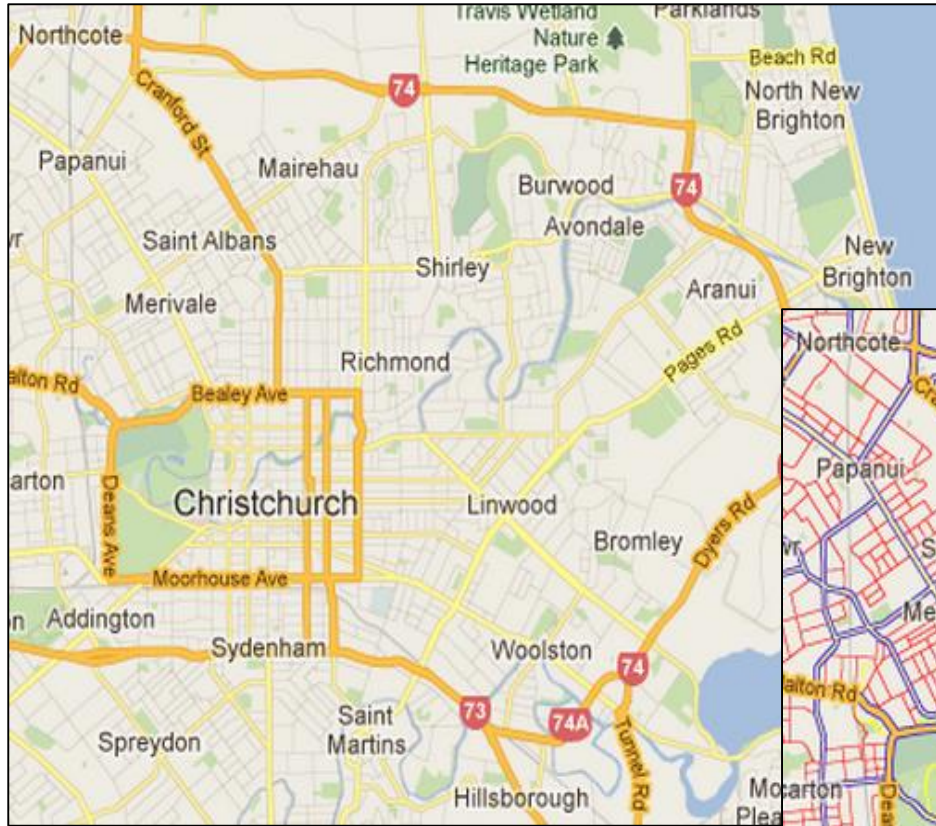
Levels of Detail (LOD)	Number of Tiles	Ground distance per pixel in meters	Storage requirements at 15 Kilobytes per tile	2010 Disk storage costs at \$100 per Terabyte	prices RAM memory storage costs at \$30 per Gigabyte
1	4	78,272	60 Kilobytes (KB)	\$0.000006	\$0.002
2	16	39,136	240 KB	\$0.00002	\$0.007
3	64	19,568	968 KB	\$0.0001	\$0.03
4	256	9,784	3.75 Megabytes (MB)	\$0.0004	\$0.11
5	1,024	4,892	15 MB	\$0.001	\$0.44
6	4,096	2,446	60 MB	\$0.006	\$1.76
7	16,384	1,223	240 MB	\$0.02	\$7.03
8	65,536	611.50	960 MB	\$0.09	\$28.13
9	262,144	305.75	3.75 Gigabytes (GB)	\$0.37	\$112.50
10	1,048,576	152.88	15 GB	\$1.46	\$450.00
11	4,194,304	76.44	60 GB	\$5.86	\$1,800.00
12	16,777,216	38.22	240 GB	\$23.44	\$7,200.00
13	67,108,864	19.11	968 GB	\$93.75	\$28,800.00
14	268,435,456	9.55	3.75 Terabytes (TB)	\$375	\$115,200.00
15	1,073,741,824	4.78	15 TB	\$1,500	\$460,800.00
16	4,294,967,296	2.39	60 TB	\$6,000	\$1,843,200.00
17	17,179,869,184	1.19	240 TB	\$24,000	\$7,372,800.00
18	68,719,476,736	0.60	960 TB	\$96,000	\$29,491,200.00
19	274,877,906,944	0.30	3.75 Petabytes (PB)	\$384,000	\$117,964,800.00
20	1,099,511,627,776	0.15	15 PB	\$1,536,000	\$471,859,200.00
Total	1,466,015,503,700		20,480 Terabytes or 20 Petabytes	\$2,048,000	\$629,145,600

Level s of Detail (LOD)	Number of Tiles	Storage requirements at 15 Kilobytes per PNG tile	2014 disk storage costs at \$50 per Terabyte	2014 RAM memory storage costs at \$15 per Gigabyte
1	4	60 Kilobytes (KB)	\$0.000003	\$0.0010
2	16	240 KB	\$0.000010	\$0.0035
3	64	968 KB	\$0.000050	\$0.02
4	256	3.75 Megabytes (MB)	\$0.000200	\$0.06
5	1,024	15 MB	\$0.000500	\$0.22
6	4,096	60 MB	\$0.003000	\$0.88
7	16,384	240 MB	\$0.01	\$3.52
8	65,536	960 MB	\$0.05	\$14.07
9	262,144	3.75 Gigabytes (GB)	\$0.19	\$56.25
10	1,048,576	15 GB	\$0.73	\$225.00
11	4,194,304	60 GB	\$2.93	\$900.00
12	16,777,216	240 GB	\$11.72	\$3,600.00
13	67,108,864	968 GB	\$46.88	\$14,400.00
14	268,435,456	3.75 Terabytes (TB)	\$187.50	\$57,600.00
15	1,073,741,824	15 TB	\$750.00	\$230,400.00
16	4,294,967,296	60 TB	\$3,000.00	\$921,600.00
17	17,179,869,184	240 TB	\$12,000.00	\$3,686,400.00
18	68,719,476,736	960 TB	\$48,000.00	\$14,745,600.00
19	274,877,906,944	3.75 Petabytes (PB)	\$192,000.00	\$58,982,400.00
20	1,099,511,627,776	15 PB	\$768,000.00	\$235,929,600.00
Total	1,466,015,503,700	20,480 TB or 20 PB	\$1,024,000.00	\$314,572,800.00

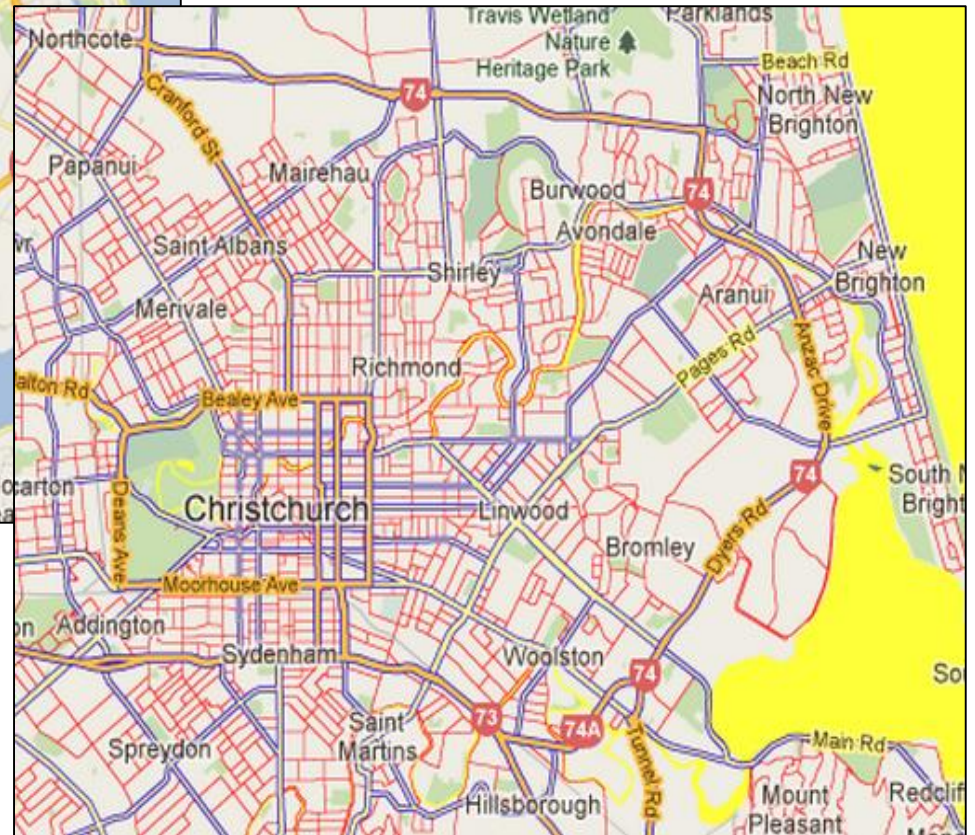
What does it mean?

- Supplying MSP (Multi-Scale Pan-able) maps is very expensive.
- Companies competing to provide the most up-to-date and fastest maps.
- Most important aspect of the online map is the speed of map display.
 - Governments and open-source solutions can't compete
 - Open source solutions too slow
 - Server resources, technology too expensive

2010 - Google Changes to Vector Tiles

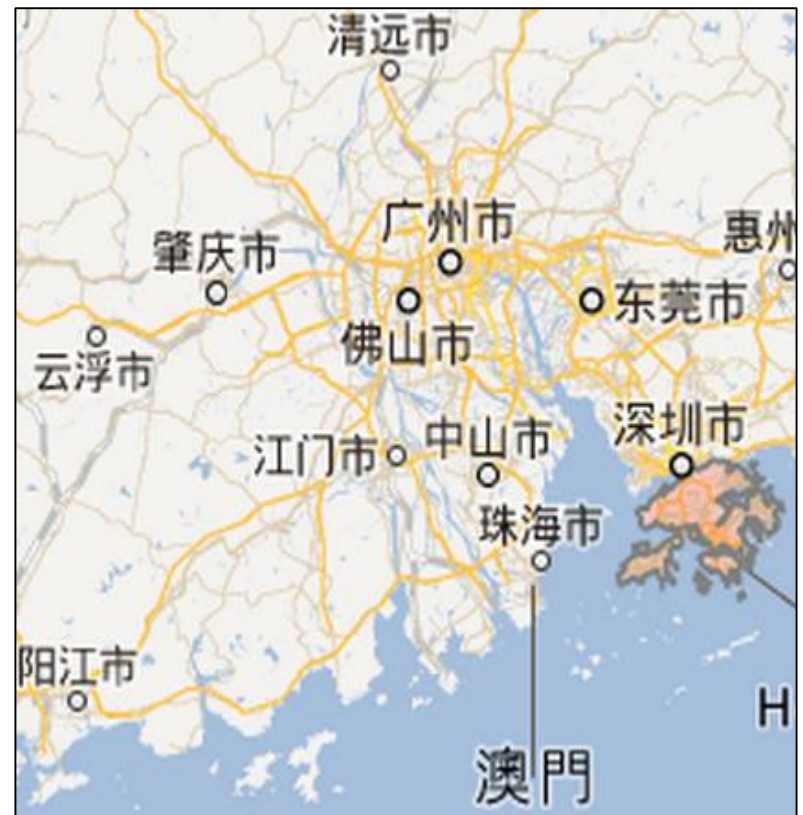
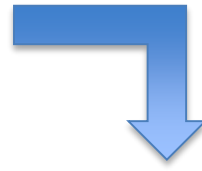


Vector tiles makes
it possible to style
the map



Rasterization done toward the
end of the map delivery
process.

and change text ...



Other map providers
“burn” text into each tile.

Evaluation of APIs

- Length-of-code criteria
 - How much code is needed to perform a specific operation
- Page-length code premise
 - Any code using an API should only be one page in length
 - Longer code is an indication that the API is not properly structured to support the needed operations
 - Extra JavaScript programming is needed to make-up for the shortcomings of the API

Resultant Map



Google Maps API

```
<html>
  <head>
    <script type="text/javascript" src="http://maps.google.com/maps/api/js?sensor=false">
    </script>
    <script type="text/javascript">
      function initialize(){
        var latlng = new google.maps.LatLng(46.810928, -90.817981);
        var myOptions = {
          zoom: 15,
          center: latlng,
          mapTypeId: google.maps.MapTypeId.ROADMAP
        };
        var map=new google.maps.Map(document.getElementById("map_canvas"),myOptions);
      }
    </script>
    <title></title>
  </head>
  <body onload="initialize()">
    <div id="map_canvas" style="width:600px; height:300px">
    </div>
  </body>
</html>
```

Link to Google Maps API

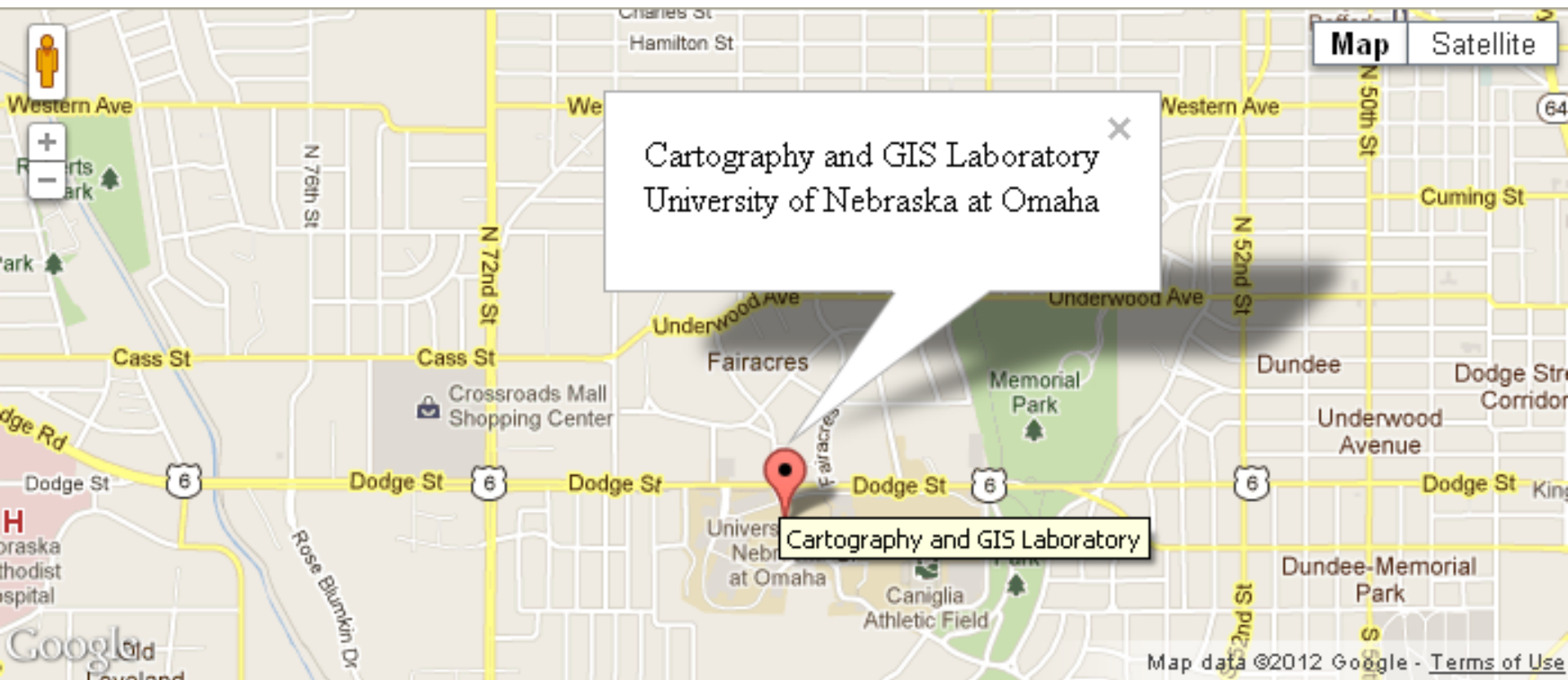
Latitude Longitude

Zoom level: 0-20

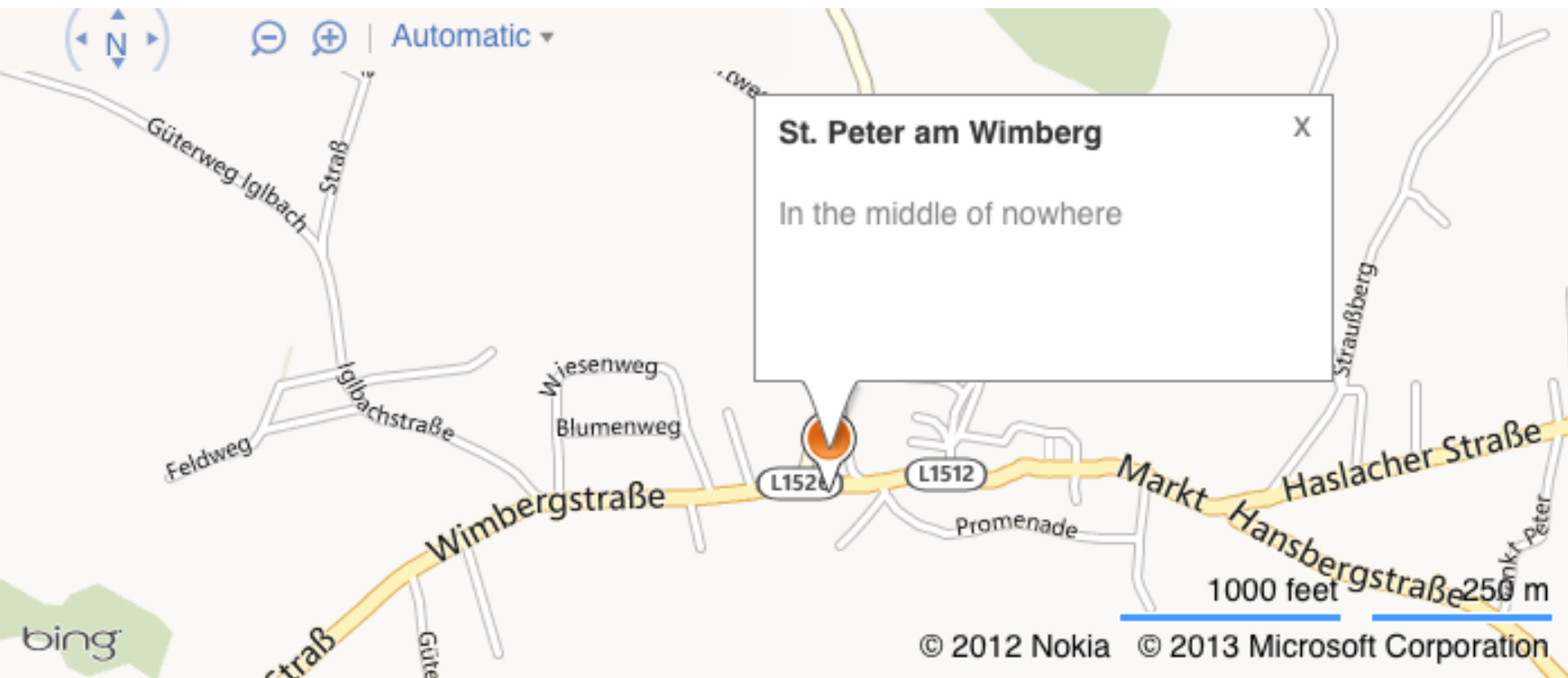
ROADMAP, HYBRID, SATELLITE or TERRAIN

Size of map in pixels

The map



Bing



Bing code

```
function GetMap(){
    map = new Microsoft.Maps.Map(document.getElementById("myMap"), {
        credentials: "AjPEInnalw4KLza4VYB3g06IbyZJ0r-J6P9W1qRjDShZYeMdPuSYEblFfr1ellgF"});
    var loc = new Microsoft.Maps.Location(48.501924, 14.078808);

    map.setView({
        center: loc,
        zoom: 15});

    var pin = new Microsoft.Maps.Pushpin(loc, {text: '1'});
    pinInfobox = new Microsoft.Maps.Infobox(pin.getLocation(), {
        title: 'St. Peter am Wimberg',
        description: 'In the middle of nowhere',
        visible: false,
        offset: new Microsoft.Maps.Point(0, 15)});

    Microsoft.Maps.Events.addHandler(pin, 'click', displayInfobox);
    Microsoft.Maps.Events.addHandler(map, 'viewchange', hideInfobox);

    map.entities.push(pin);
    map.entities.push(pinInfobox);
}
```

Infobox for Pushpin

Text for inside of Pushpin

Initially invisible

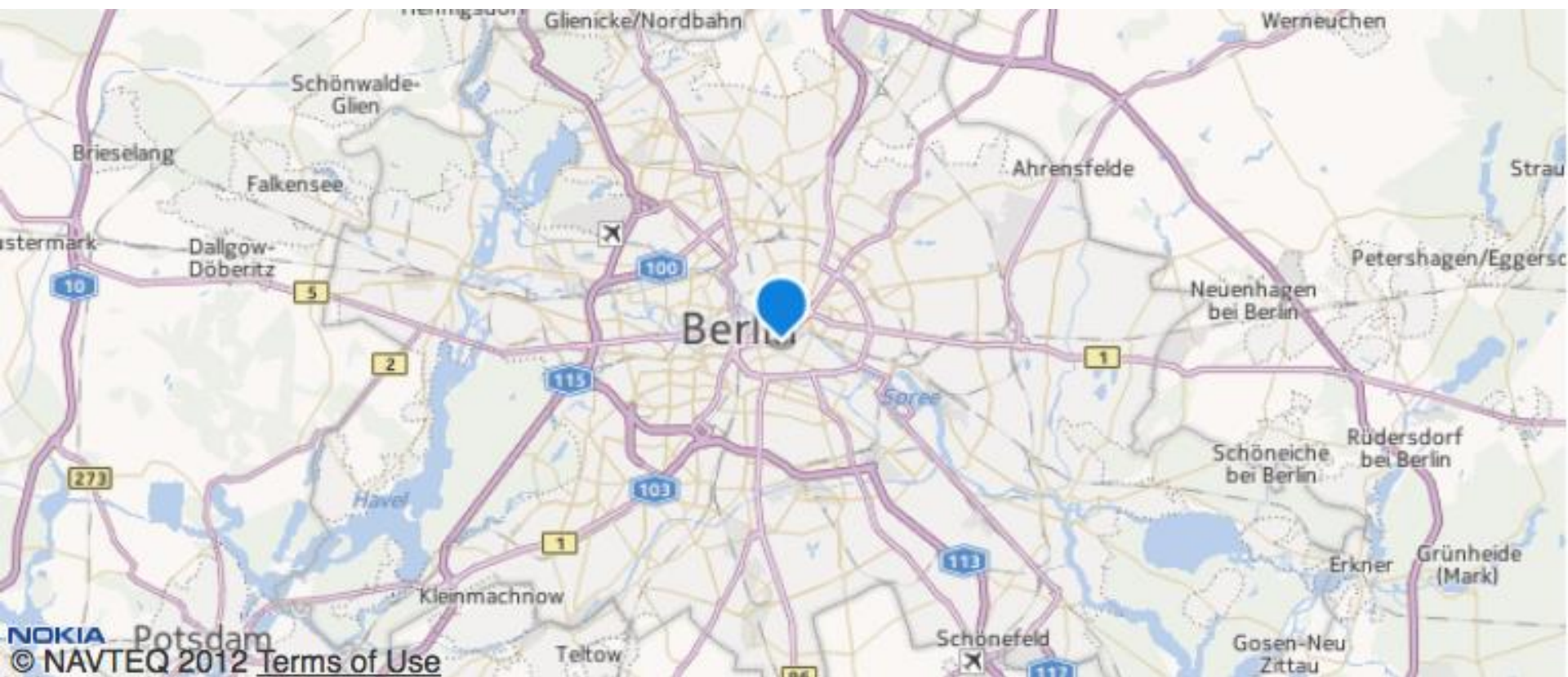
Handler for click event

Hide infobox if map is moved

Add pin and infobox to map

HTML code missing

Nokia



MapQuest



MapQuest Code

```
<html>
```

```
<head>
```

```
<script src="http://www.mapquestapi.com/sdk/js/v7.0.s/mqa.toolkit.js?key=Fmjtd%7Cluub29urnl%2Cb5596z2h4"></script>
```

```
<script type="text/javascript">
```

```
/*An example of using the MQA.EventUtil to hook into the window load event and execute defined func  
passed in as the last parameter. You could alternatively create a plain function here and have it  
executed whenever you like (e.g. <body onload="yourfunction">).*/
```

```
MQA.EventUtil.observe(window, 'load', function() {
```

```
/*Create an object for options*/
```

```
var options={
```

```
elt:document.getElementById('map'),    /*ID of element on the page where you want the map add
```

```
zoom:10,                               /*initial zoom level of map*/
```

```
latLng:{lat:39.743943, lng:-105.020089}, /*center of map in latitude/longitude*/
```

```
mtype:'map'                            /*map type (map)*/
```

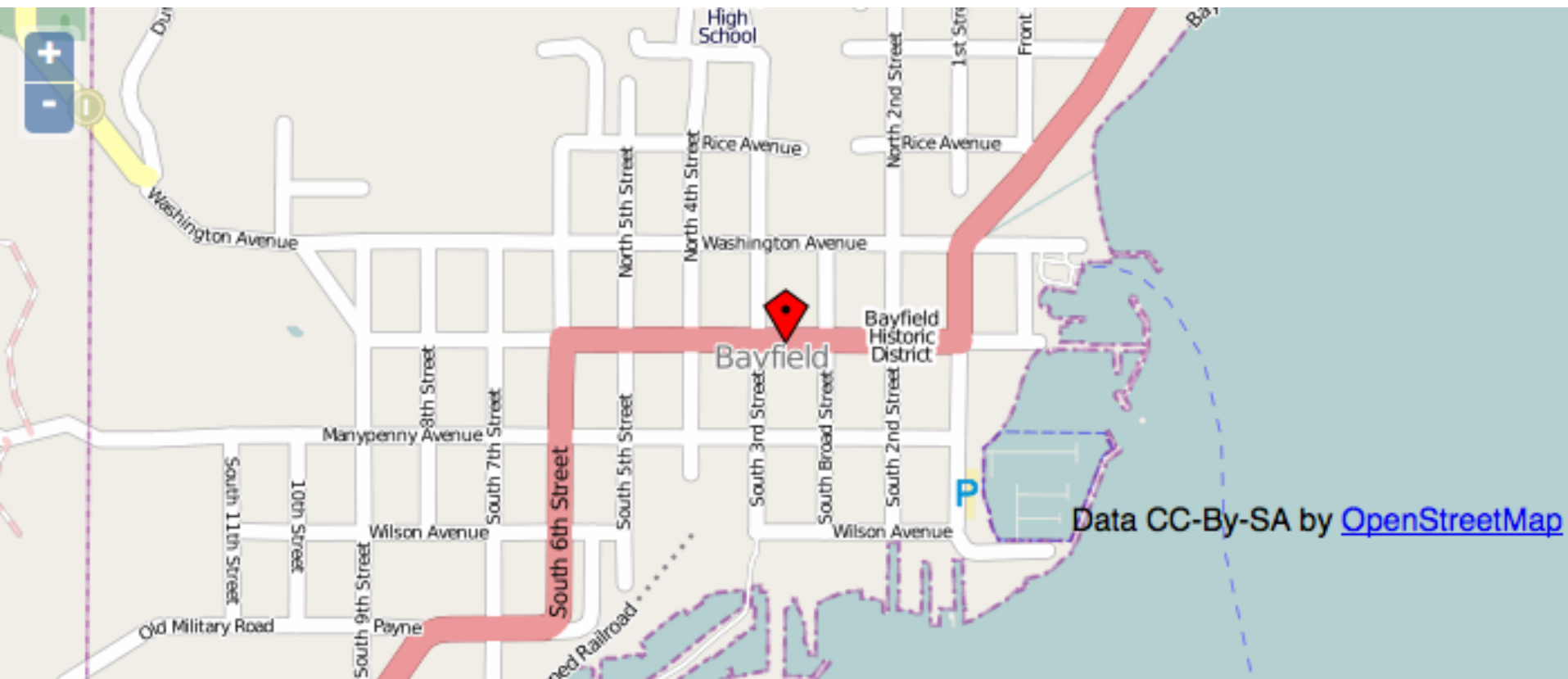
```
};
```

```
/*Construct an instance of MQA.TileMap with the options object*/
```

```
window.map = new MQA.TileMap(options);
```

```
/*An example using the MQA.Poi constructor. You will need to pass in an object containing  
the lat (Latitude) and lng (Longitude) property defining where to place the POI
```

OpenStreetMap



OpenStreetMap Code

```
<!DOCTYPE HTML>
<html>
<head>
<title>OpenLayers Simplest Example</title>
</head>
<body>
<div id="Map" style="height:300px; width:700px"></div>
<script src="http://www.openlayers.org/api/OpenLayers.js"></script>
<script>
    var lat      = 46.810928;
    var lon      = -90.817981;
    var zoom     = 15;

    var fromProjection = new OpenLayers.Projection("EPSG:4326"); // Transform from WGS 1984
    var toProjection   = new OpenLayers.Projection("EPSG:900913"); // to Spherical Mercator Projection
    var position       = new OpenLayers.LonLat(lon, lat).transform( fromProjection, toProjection);

    map = new OpenLayers.Map("Map");
    var mapnik      = new OpenLayers.Layer.OSM();
    map.addLayer(mapnik);

    var markers = new OpenLayers.Layer.Markers( "Markers" );
    map.addLayer(markers);
    markers.addMarker(new OpenLayers.Marker(position));

    map.setCenter(position, zoom);
```

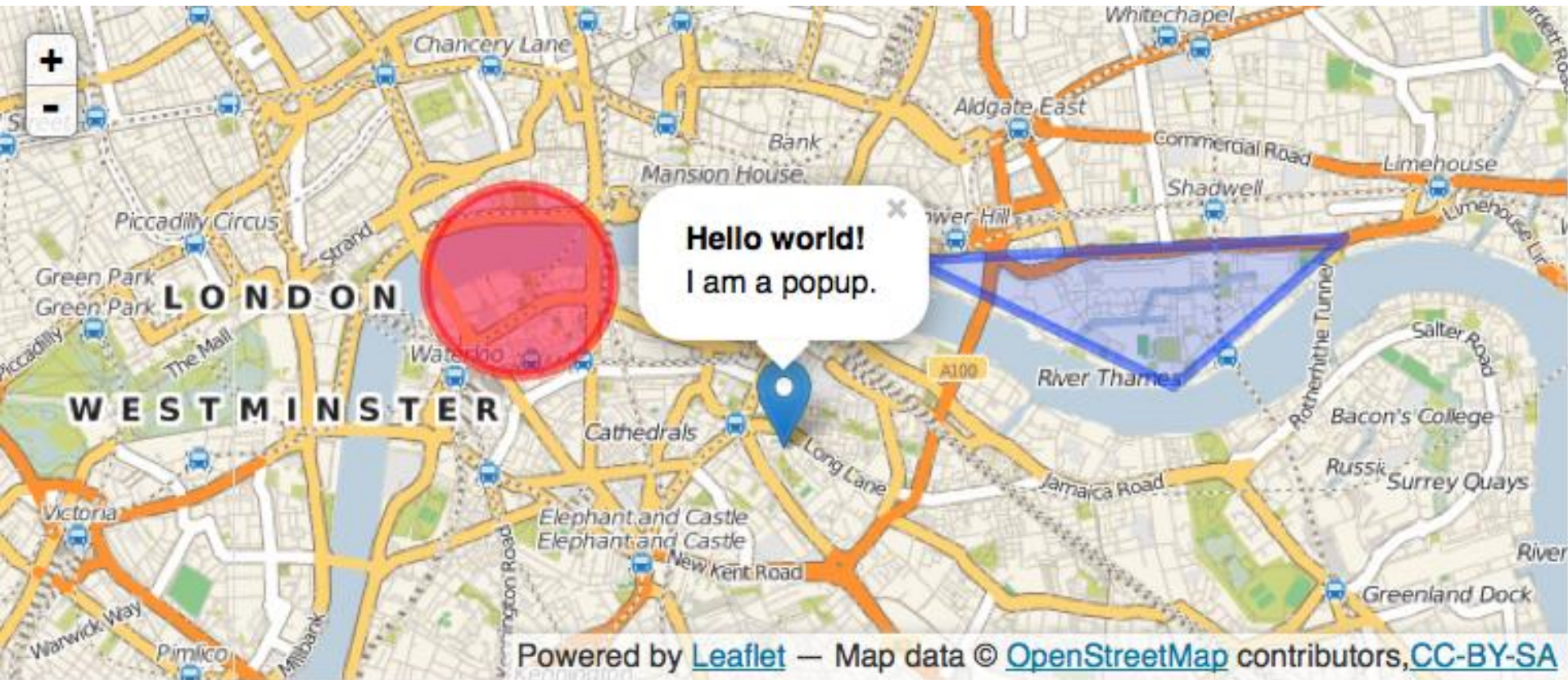
Baidu

Baidu Map API Key

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <script type="text/javascript"
    src="http://api.map.baidu.com/api?v=1.5&ak=927cb23887926d2b345b0c762045feb3"></script>
  <script type="text/javascript">
    function initialize() {
      var map = new BMap.Map("allmap");           // Create a map
      var point = new BMap.Point(116.404, 39.915); // Make a point
      map.centerAndZoom(point, 15);                // Set map center and zoom level
      var marker = new BMap.Marker(point);          // Create a marker
      map.addControl(new BMap.NavigationControl()); // Add Navigation Control for zoom and pan
      map.addControl(new BMap.ScaleControl());       // Show the scale of the map 添加比例尺控件
      map.addControl(new BMap.OverviewMapControl()); // Add an overview control
      map.enableScrollWheelZoom();                  // Enable wheel of mouse for zooming
      map.addControl(new BMap.MapTypeControl());     // Add a control for choosing map types
      map.addOverlay(marker);                        // Add the marker to the map
      marker.setAnimation(BMAP_ANIMATION_BOUNCE);   // set animation type of the marker

      var infoWindow = new BMap.InfoWindow("Marker Information"); // Create an info window
      marker.addEventListener("click", function(){this.openInfoWindow(infoWindow);}); //Add a listener
    }
  </script>
  <title>Animated Marker</title>
</head>
<body onLoad="initialize()">
  <div id="allmap" style="width:700px; height:300px"> </div>
</body>
</html>
```


Leaflet



Leaflet code

```
<body>
  <div id="map" style="width: 700px; height: 300px">
    </div>
    <script src="http://leafletjs.com/dist/leaflet.js">
    </script>
    <script>
      var map = L.map('map').setView([51.505, -0.09], 13);

      L.tileLayer('http://{s}.tile.cloudmade.com/BC9A493B41014CAABB98F0471D759707/997/256/{z}/{x}/{y}.png', {
        maxZoom: 18,
        attribution: 'Map data &copy; <a href="http://openstreetmap.org">OpenStreetMap</a> contributors, ..
      }).addTo(map);

      L.marker([51.5, -0.09]).addTo(map).bindPopup("<b>Hello world!</b><br />I am a popup.").openPopup();

      L.circle([51.508, -0.11], 500, {
        color: 'red',
        fillColor: '#f03',
        fillOpacity: 0.5
      }).addTo(map).bindPopup("I am a circle.");

      L.polygon([[51.509, -0.08], [51.503, -0.06], [51.51, -0.047]]).addTo(map).bindPopup("I am a polygon.");
    </script>
  </body>
```

Center & zoom

Get 28 KB Leaflet API

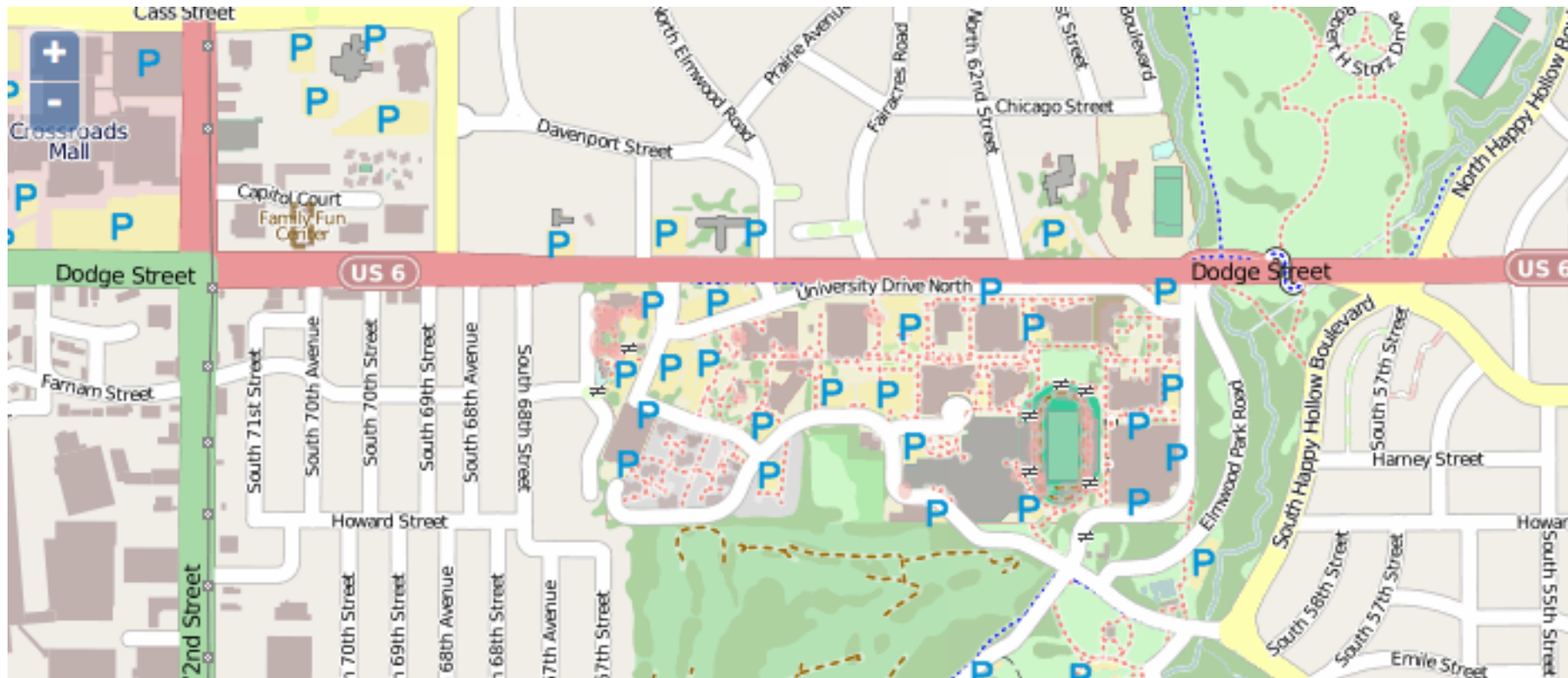
Get CloudMade OpenStreetMap Tiles

Marker and popup

Circle and popup

Polygon and popup

MapStraction



MapStraction Code

```
<head>
  <title>Basic Mapstraction Map</title>
  <script src="http://openlayers.org/api/OpenLayers.js"></script>
  <script src="https://raw.githubusercontent.com/mapstraction/mxn/master/source/mxn.js?(openlayers)"
    type="text/javascript"></script>
  <style type="text/css">
    #map {
      width: 700px; height: 300px;
    }
  </style>

  <script type="text/javascript">
    var mapstraction;
    function create_map() {
      mapstraction = new mxn.Mapstraction('mymap', 'openlayers');
      mapstraction.setCenterAndZoom(
        new mxn.LatLonPoint(41.258531, -96.012599), 15);
    }
  </script>
</head>
```

Get OpenLayers
API

Get MapStraction API

Map size

MapStraction map
with OpenLayers

Center & Zoom
Level

Shortest code?

- Difficult to evaluate
 - All APIs have verbose and short-hand (default) forms
 - Variation in the functionality per code line
 - Leads to variation in functionality per code page
- Leaflet is compact
- Nokia & OpenStreetMap are verbose
- But ...

What about long-term viability?

- Maintaining/developing the code
- Maintaining/developing the underlying map
- Who is updating their API?
- Map updates
 - Currency of map (How up-to-date is the map?)
- Speed of map delivery
 - How fast the map is delivered to the user.
 - Speed of map delivery is the single most important aspect of mapping in the cloud

Revenue Stream

- Updating maps and making them available is expensive
 - \$600,000 for hard-drives to store all tiles for 20 zoom levels
 - Continuous and major revenue stream is needed
- Making money from maps that are provided for free is not easy
 - Whichever company does this the best has the best API

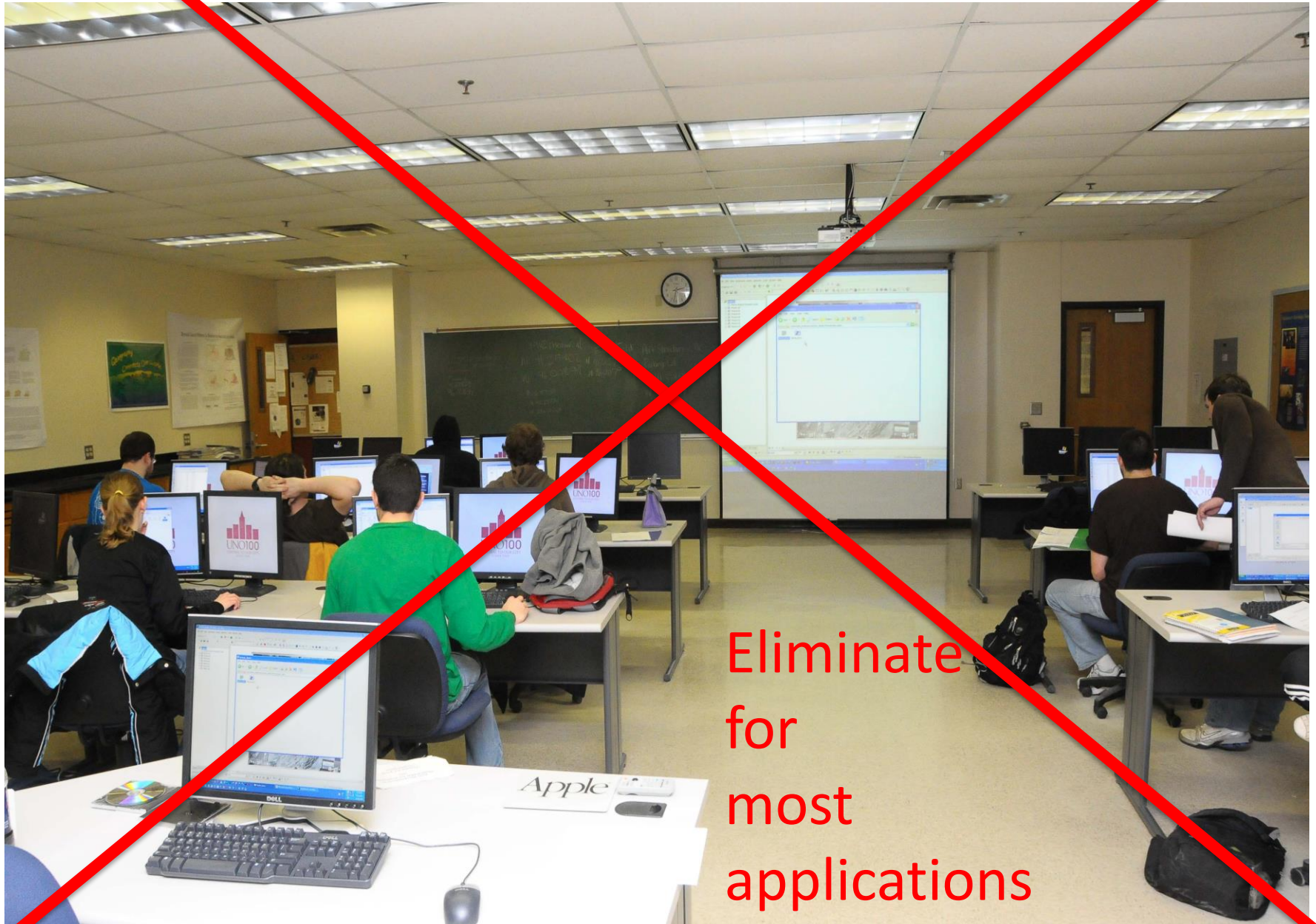
Comments about MSP maps

- A student of ESRI ArcGIS: “Why can’t this be more like Google Maps?”
 - The map interface in GIS software is old. The software produces maps that nobody wants to use.
- About Bing Maps: “I think Google Maps does a better job of representing the more important roads. I also prefer the colouring scheme used in Google Maps.”
 - Google Maps has become a standard map. Other representations are not viewed as maps – including Apple maps.

Ubiquitous Geospatial Education lab



Ubiquitous Geospatial Education lab

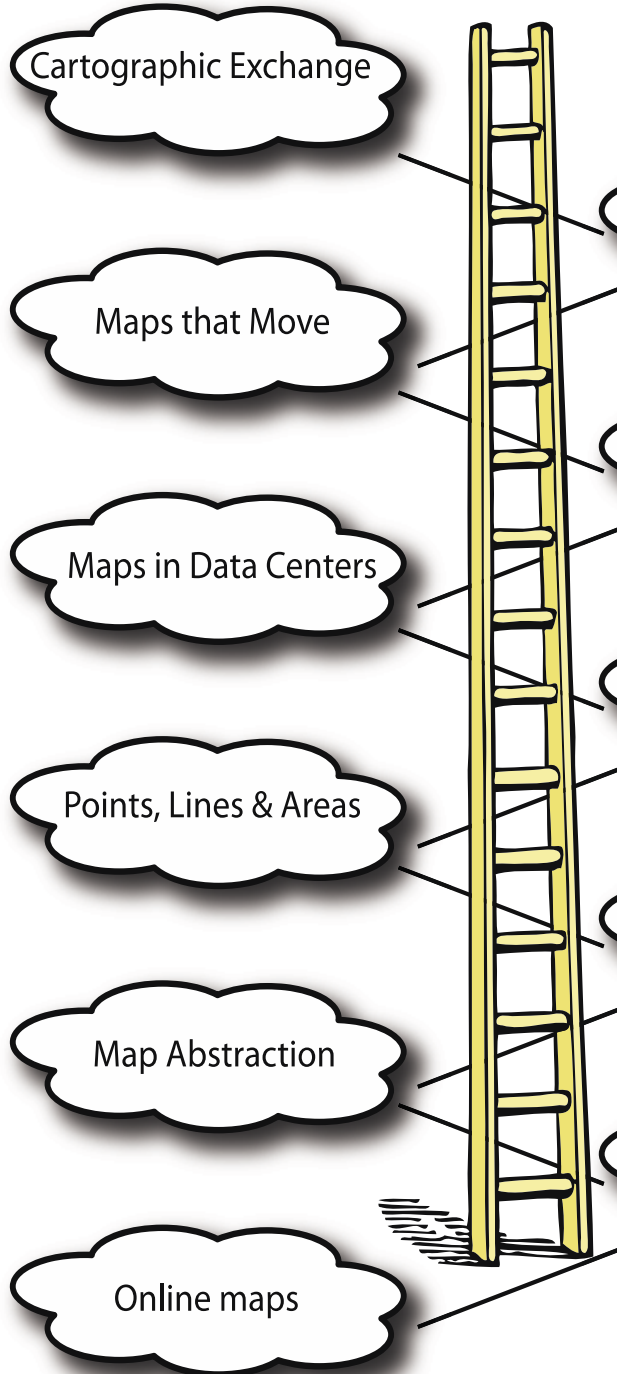


Eliminate
for
most
applications

Turning a Ship

- Lab environment extremely important aspect of geospatial education
- In the past...
 - Yearly software maintenance contract
 - Hardware replacement every 3-4 years
 - Software that students couldn't otherwise access
- In the future...
 - Creates an environment for human interaction to the cloud
 - Basic computers, large monitors, no software

THEORY



Cartographic Exchange

Maps that Move

Maps in Data Centers

Points, Lines & Areas

Map Abstraction

Online maps

Animation

Databases

Mashups

JavaScript

Web & HTML

PRACTICE

Future

- Next 20 years will see even greater changes than last 20 years
- Movement to the cloud
 - New methods of map delivery
 - New raster and vector tiles
 - New methods of map interaction
 - It's all about interaction
 - Developing new methods of instruction
 - Effectively combining theory and practice
 - Changing computer labs to more effectively use the cloud

Thank you