User Interaction: Google Maps API

Asst. Professor Donald J. Patterson
INF 133 Fall 2011
Why maps?

- Central to much mobile interaction work.
- It can be generalized to interactions with enormous images very easily.
- Mapping data is a natural visualization that helps to reveal patterns.
Why maps?

http://gigapan.org/viewGigapanFullscreen.php?auth=03ef14483ee899496648c2b4b06233c
Credit: John Snow, On the Mode of Communication of Cholera, 1855
Google Maps API

- Google Maps v3 is optimized for mobile
- It used to be just for mobile but is being adapted as the standard Maps API for all platforms
Basics

- Google hosts most of the code in a javascript library
- A developer includes the javascript library
  - Just like jQuery
- A developer makes calls to the library to:
  - display the map
  - display controls
  - display points
  - display other U/I elements
Google Maps API v3

Basics

• You can register event handlers
• So that when a user does something on the map, you can do something on the webpage also (through JavaScript)
Basic Map Objects

• Create a placeholder in your webpage to hold the map object

```html
<html>
  <head>
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.4/jquery.min.js"></script>
    <script src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.8/jquery-ui.min.js"></script>
    <script src="Step01.js"></script>
  </head>

  <body>
    <div id="map_canvas" style="float:right;width:500px;height:353px;border:1px black">
      <p>
        <img src="http://farm5.static.flickr.com/4093/4770854356_143415fb82.jpg" width="500" height="353" alt="Globe centred on the Americas - Satellite image - PlanetObserver"></a>
      </p>
    </div>
  </body>
</html>
```

Basic Map Objects

- The size is up to you
- Android/iPhone work best at full screen
- Javascript can detect this for you

```javascript
function detectBrowser() {
  var userAgent = navigator.userAgent,
  mapdiv = document.getElementById("map_canvas");

  if (userAgent.indexOf('iPhone') !== -1 || userAgent.indexOf('Android') !== -1) {
    mapdiv.style.width = '100%';
    mapdiv.style.height = '100%';
  }
}
```
Google Maps API v3

Basic Map Loading

• Loading the Google Maps Library

<script type="text/javascript" src="http://maps.google.com/maps/api/js?sensor=set_to_true_or_false">

• Parameters for the library
  • "sensor": does your location come from a sensor?
    • mandatory parameter
  • allows Google to respect business agreements
  • "language": what language are the labels in?
  • "region": bias behavior toward a region
  • "Paris" in TX or "Paris" in France?
Putting the map on the web page

```javascript
function initializeMap(){
    var latlng = new google.maps.LatLng(33.643298, -117.841983);
    var myOptions = {
        zoom: 8,
        center: latlng,
        mapTypeId: google.maps.MapTypeId.ROADMAP
    };
    var map = new google.maps.Map(document.getElementById("map_canvas"), myOptions);
}
```
Google Maps API v3

Putting the things on the web page

- Markers

```javascript
var marker = new google.maps.Marker({
  position: latlng,
  map: map,
  title: "Hello World!"
});
```
Putting the things on the web page

- Icons

```javascript
var image = new google.maps.MarkerImage('images/icon.png',
    // This marker is 32 pixels wide by 32 pixels tall.
    new google.maps.Size(32, 32),
    // The origin for this image is 0,0.
    new google.maps.Point(0,0),
    // The anchor for this image is at 0,16.
    new google.maps.Point(0, 16));

var shadow = new google.maps.MarkerImage('images/shadow.png',
    // The shadow image is larger in the horizontal dimension
    // while the position and offset are the same as for the main image.
    new google.maps.Size(41, 18),
    new google.maps.Point(0,0),
    new google.maps.Point(0,0));

var marker = new google.maps.Marker({
    position: latlng,
    map: map,
    shadow:shadow,
    icon: image,
    title:"Hello World!"
});
```
Google Maps API v3

Putting the things on the web page

- Polylines
Google Maps API v3

Putting the things on the web page

• Polygons
Google Maps API v3

Putting the things on the web page

- Info Windows

Uluru, also referred to as Ayers Rock, is a large sandstone rock formation in the southern part of the Northern Territory, central Australia. It lies 335 km (208 mi) south west of the nearest large town, Alice Springs; 450 km (280 mi) by road. Kata Tjuta and Uluru are the two major features of the Uluru - Kata Tjuta National Park. Uluru is sacred to the Pitjanjatjara and Yankunytjatjara, the Aboriginal people of the area. It has many springs, waterholes, rock caves and ancient paintings. Uluru is listed as a World Heritage Site.

Putting the things on the web page

```javascript
var contentString = '<div id="content">' +
    '<div id="siteNotice">' +
    '</div>' +
    '<h1 id="firstHeading" class="firstHeading">Uluru</h1>' +
    '<div id="bodyContent">' +
    '<p>Uluru, also referred to as <b>Ayers Rock</b>, is a large sandstone rock formation in the southern part of the Northern Territory, central Australia. It lies 335 km (208 mi) south west of the nearest large town, Alice Springs; 450 km (280 mi) by road. Kata Tjuta and Uluru are the two major features of the Uluru – Kata Tjuta National Park. Uluru is sacred to the Pitjantjatjara and Yankunytjatjara, the Aboriginal people of the area. It has many springs, waterholes, rock caves and ancient paintings. Uluru is listed as a World Heritage Site.</p>' +
    '</div>' +
    '</div>' +

var infowindow = new google.maps.InfoWindow({
    content: contentString
});

var marker = new google.maps.Marker({
    position: myLatLng,
    map: map,
    title:"Uluru (Ayers Rock)"
});

google.maps.event.addListener(marker, 'click', function() {
    infowindow.open(map,marker);
});
```
Google Maps API v3

Geocoding Addresses

• Icons

```
function codeAddressHelper(results, status) {
    if (status == google.maps.GeocoderStatus.OK) {
        map.setCenter(results[0].geometry.location);
        var marker = new google.maps.Marker({
            map: map,
            position: results[0].geometry.location
        });
    } else {
        alert("Geocode was not successful for the following reason: " + status);
    }
}

function codeAddress() {
    var geocoder = new google.maps.Geocoder();
    var address = document.getElementById("address").value;
    geocoder.geocode({ 'address': address, codeAddressHelper });
}
```
Google Maps API v3

More info

- Google Maps API v3
  - http://code.google.com/apis/maps/documentation/javascript/
How do you debug?

- Web Developer Toolbar
- Firefox Error Console
- Internet Explorer Script Debugger
- Firebug
- Google’s GLog library
  - For adding your own debugging statements
- screencast tutorials here:
Assignment #3

- Add a map to Assignment #2
- AJAX call
- Make your table entries link to the map (center on click)
- Plot the table entries on the map
- Enhance the map in one interesting way
  - For example, multiple map marker icons
  - More than a custom marker or an info window