

Google™



Live Wave: Notes and Q&A

<http://bit.ly/ctP6P5>

Where You At?

Connecting Your Users With the Places Around Them

Chris Lambert, Marcelo Camelo & Dave Wang

May 20th, 2010

Agenda

- Status of Location APIs today
- Google Latitude
- New Location APIs
- What this enables
- Places

Live Wave: Notes and Q&A

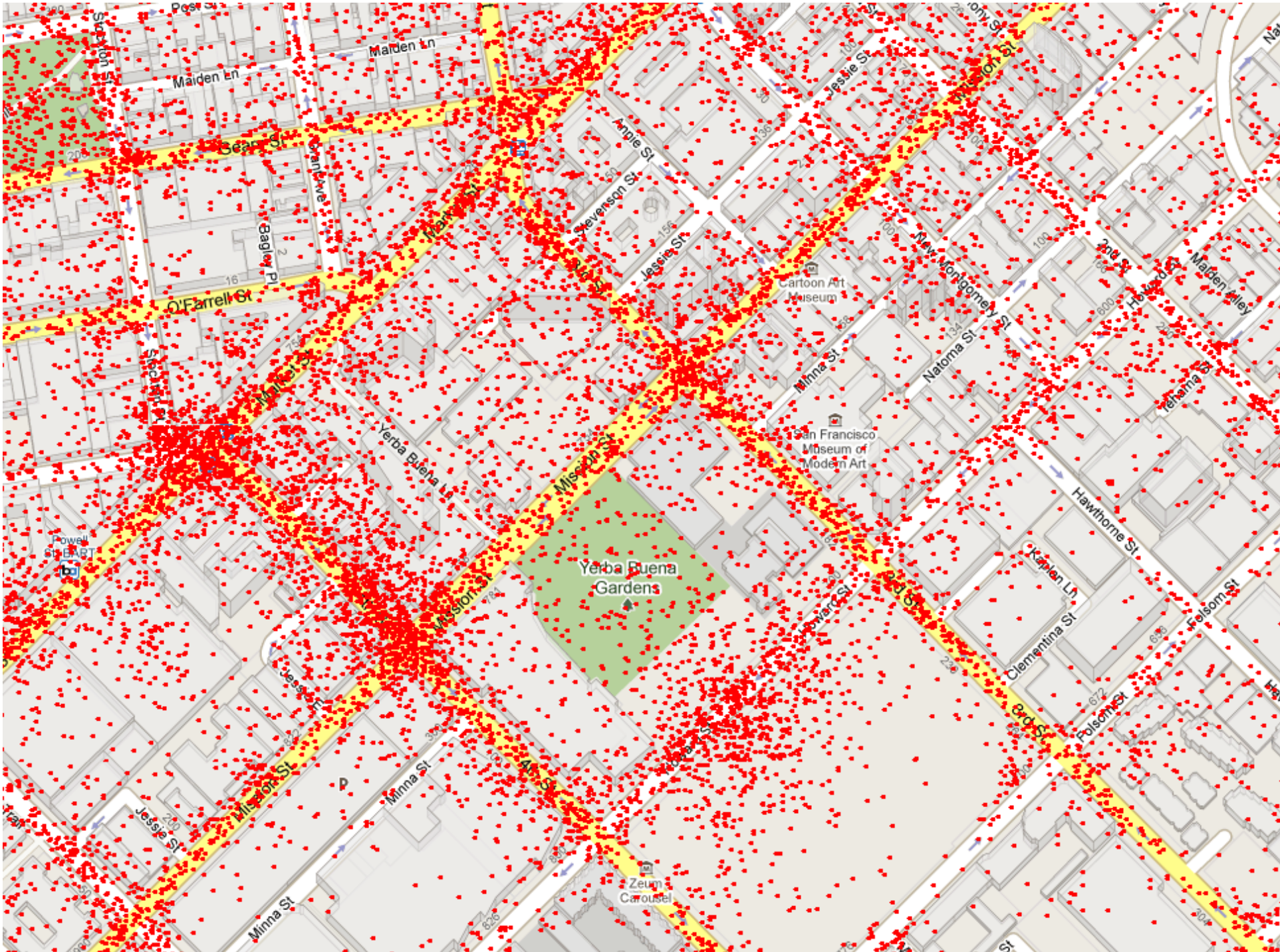
<http://bit.ly/ctP6P5>

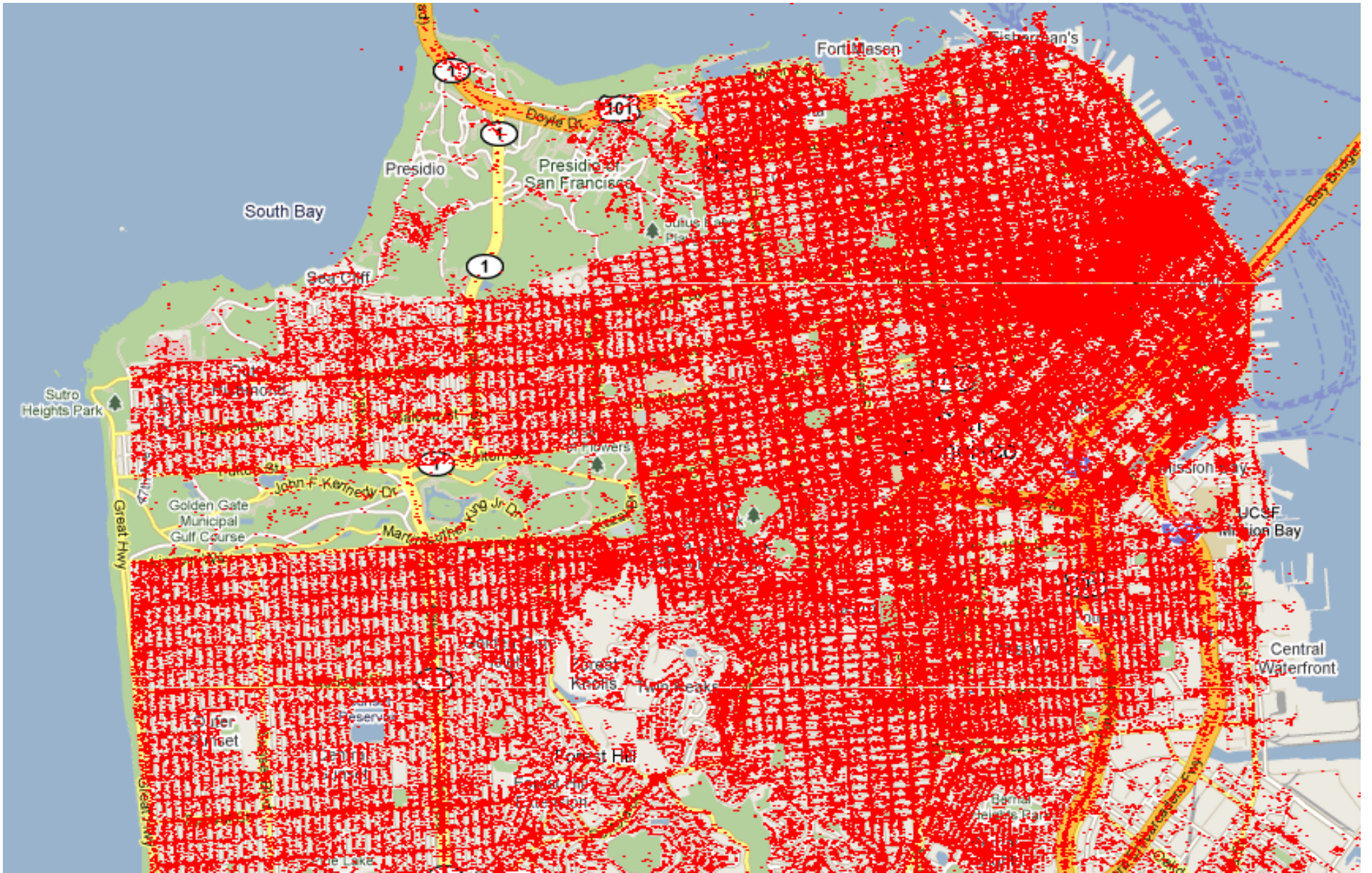
Google's Location APIs

- HTML5 Geolocation
 - Firefox
 - Chrome
 - Android Browser
- Gears Geolocation
 - Safari
 - Internet Explorer
- Google Toolbar on Internet Explorer
- Android's native Location Manager
- These APIs determine location through a mix of different location sources

Location API Sources

- IP Geolocation
 - City level accuracy
 - Works well on the desktop, no power cost
- Cell Tower Triangulation
 - Neighborhood or city level accuracy
 - Requires a connected mobile device
- Wi-Fi Mac Addresses
 - Street level accuracy
 - Requires a Wi-Fi enabled device
- GPS or assisted GPS
 - Best available accuracy
 - Power cost and unavailable indoors

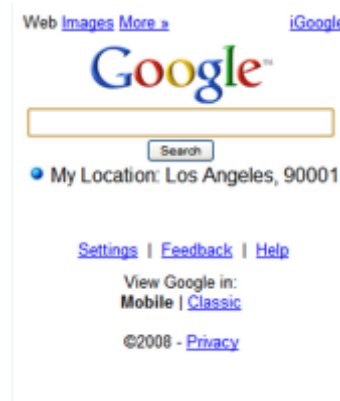
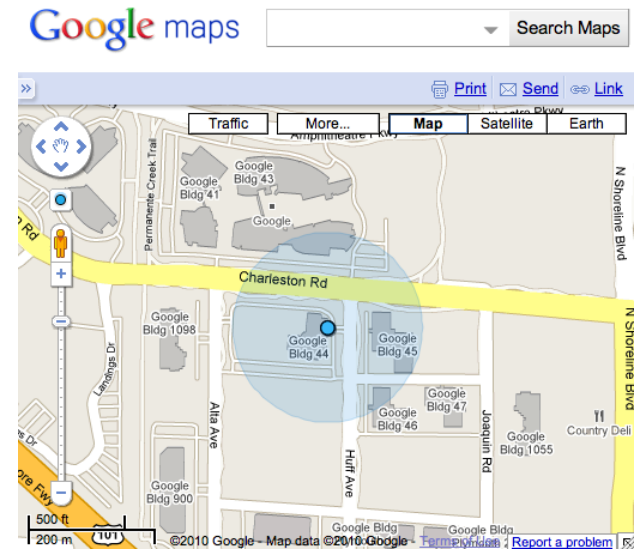
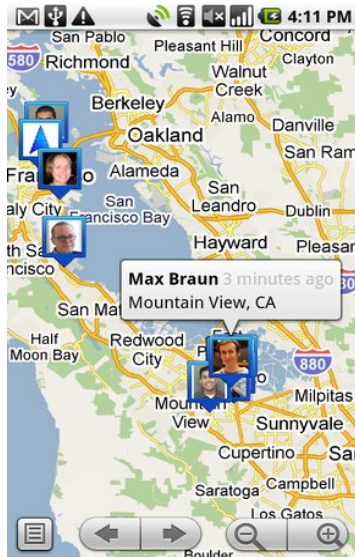




Easily accessible with HTML5

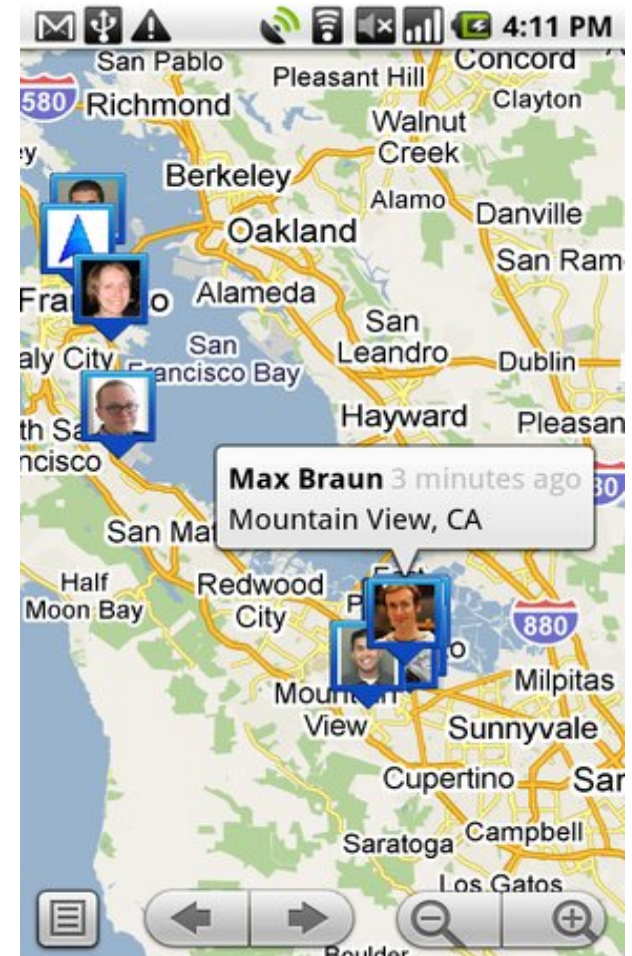
```
if (navigator.geolocation) {  
    navigator.geolocation.getCurrentPosition(showMap);  
}  
  
function showMap(location) {  
    map.setCenter(new google.maps.LatLng(  
        location.position.latitude,  
        location.position.longitude));  
}
```

Used extensively across Google properties



Google Latitude

- Latitude first launched as way to share your location with friends
- Continuous location reporting on Android, Blackberry, Windows Mobile, Nokia & iGoogle
 - Over 3 million active users
 - Approaching 100 million daily location reports
- Many applications would benefit from this location information, but it's not practical to have multiple location services running on your phone



Google Latitude API

- Launching today!

Google Latitude API

- Launching today!
- Latitude has always put users in charge of their privacy
 - Per-friend sharing and granularity control
- Latitude API follows the same privacy model
 - Users must explicitly authorize each application
 - Current location or location history
 - City level or best available granularity
 - Easy to revoke access at any time
 - No firehose of data
 - No friend locations

Latitude API Concepts

- The Latitude API is an authenticated RESTful API
- All API operations are based around the concept of user location
- User location is represented by a location resource
- The Latitude API provides two resource collections:
 - Current Location
 - Location History
- Location can be viewed at either the best available granularity, or city-level only

Location Resource Object

```
{  
  "kind": "latitude#location",  
  "timestampMs": "1274169376143",  
  "latitude": 37.783706,,  
  "longitude": -122.399755,  
  "accuracy": 12,  
  "speed": 15,  
  "heading": 45,  
  "altitude": 120,  
  "altitudeAccuracy": 20,  
  "placeid": "..."  
}
```

Current Location: Supported Operations

GET [https://www.googleapis.com/latitude/v1/
currentLocation](https://www.googleapis.com/latitude/v1/currentLocation)

GET [https://www.googleapis.com/latitude/v1/
currentLocation?granularity=best](https://www.googleapis.com/latitude/v1/currentLocation?granularity=best)

POST [https://www.googleapis.com/latitude/v1/
currentLocation](https://www.googleapis.com/latitude/v1/currentLocation)

DELETE [https://www.googleapis.com/latitude/v1/
currentLocation](https://www.googleapis.com/latitude/v1/currentLocation)

Location History: Supported Operations

GET <https://www.googleapis.com/latitude/v1/location?max-results=1000&max-time=1274169376143>

GET <https://www.googleapis.com/latitude/v1/location/1274169376143>

POST <https://www.googleapis.com/latitude/v1/location/1274169376143>

DELETE <https://www.googleapis.com/latitude/v1/location/1274169376143>

City Level Current Location

```
GET https://www.googleapis.com/latitude/v1/  
currentLocation
```

```
{  
  "data": {  
    "kind": "latitude#location",  
    "timestampMs": "1274164031860",  
    "latitude": 37.7749295,  
    "longitude": -122.4194155  
  }  
}
```


Best Available Current Location

```
GET http://www.googleapis.com/latitude/v1/  
currentLocation?granularity=best
```

```
{  
  "data": {  
    "kind": "latitude#location",  
    "timestampMs": "1274164031860",  
    "latitude": 37.7749295,  
    "longitude": -122.4194155,  
    "accuracy": 15,  
    "altitude": 212  
  }  
}
```

Best Available Historical Location

```
GET https://www.googleapis.com/latitude/v1/  
location?granularity=best&max-results=10  
&maxtime=1262332800000
```

```
{  
  "data": {  
    "kind": "latitude#locationFeed",  
    "items": [  
      {  
        "kind": "latitude#location",  
        "timestampMs": "1262326551143",  
        "latitude": 37.786485,  
        "longitude": -122.4141,  
        "accuracy": 107  
      },  
      ...  
    ]  
  }  
}
```



Requesting User Authorization



The site **joonix.net** is requesting access to your Google Latitude account. If you are granting joonix.net access for the first time, you'll need to grant access to Latitude below and then to your Google Account on a following page.

Do you want to grant joonix.net access to:

- Your **best-available** location
- View and update your **current location** and **location history**

Grant access

Deny access

Changing access? If you've previously granted access to joonix.net and are now changing Latitude access levels (e.g. best available location or location history), you'll only need to grant access to the changed levels above to complete the changes.

Stay safe!

Please grant access only to third parties you trust to access your Latitude location data in accordance with the Latitude API [Terms of Service](#).

By granting access, you grant access to all applications or services by joonix.net. Google can't control how third parties will use your data once you grant them access, so please refer to their privacy policy to understand how your data will be used.

- Registered OAuth required
- Specify desired granularity & history permissions parameters when authorizing your request token
- API requests for authorized users will succeed once you receive an access token

What could this enable?

- Automatically geotag your travel photos based on EXIF data joined with your location history
- Receive LinkedIn alerts when anyone in your network is at the same conference
- Improved anti-fraud detection for your credit cards
- iPhone location reporters with iPhone OS 4.0
- Home automation software to warm up the house
- What will you build?

More Information

<http://code.google.com/apis/latitude>

Getting Started, Reference Guide, Support Forum, Release Notes, etc.

<http://www.google.com/latitude/apps>

Live Wave: Notes and Q&A

<http://bit.ly/ctP6P5>

Summary

- Overview of Places
 - What?
 - Why?
- Google's vision and offerings
- Hands on

What Is a Place

- "Public locations that people care about"
 - Park, monument, beach
 - Restaurant, hospital, shop
 - Country, city, neighborhood, street



Why Care?

- More meaningful and compact
- LatLng requires a map
- Better reflects how people think and communicate
- Subjective qualities: preferences, status, coolness

Why Care?

- Increased interest in location based services
- Check-in based social games
- Location based media and advertising
- 53% willing to disclose location

Places at Google

- Google Maps
- Google Place Pages
 - Goal: one page for every place in the world
 - Aggregates the internet around a place
 - Photos and videos
 - Reviews and rating
 - Related pages and maps

Place Pages






Google maps Search Maps Show search options
Find businesses, addresses and places of interest.

[Get Directions](#) [My Maps](#) [Print](#) [Email](#) [Link](#)

Sydney NSW


Australia
3:54pm Friday (GMT+10:00)
[Directions](#) [Search nearby](#) [Save to...](#) [more](#)


Photos & Videos


 Panoramic
 Panoramic
 Panoramic
 Panoramic
 YouTube


[More photos >](#) [More videos >](#)



Popular places

[Hyde Park, Sydney](#) ★★★★★ 137 ratings
 Hyde Park is a large park in Sydney, New South Wales, Australia. Hyde Park is on the eastern side of the Sydney central business district. It is the southernmost of a chain of parkland that extends north to the shore of Port Jackson (Sydney ... - [Wikipedia](#))

[Sydney Tower](#) ★★★☆☆ 160 ratings
 Sydney Tower (also known as the AMP Tower, AMP Centrepoint Tower, Centrepoint Tower or just Centrepoint) is Sydney's tallest free-standing structure, and the second tallest in Australia (with the Q1 building on the Gold Coast being the tallest ... - [Wikipedia](#))


[Sydney Opera House](#)
 The Sydney Opera House is a multi-venue performing arts centre on Bennelong Point in Sydney, New South Wales, Australia. It was conceived and largely built by Danish architect Jørn Utzon, who in 2003 received the Pritzker Prize, architecture's ... - [Wikipedia](#)

[Queen Victoria Building](#) ★★★★★ 19 ratings
 The Queen Victoria Building, or QVB, is a late nineteenth century building by the architect George McRae in the central business district of Sydney, Australia. The Romanesque Revival building is 190 metres long by 30 wide, and fills a city block ... - [Wikipedia](#)



Sponsored Links

Related Maps

 [Sydney's Happy Hours](#)
By [andre_dantas](#)
List of Happy Hours in Sydney by day of the week
[maps.google.com](#)

What About Developers?

- Lack of support for business search in the Maps API
- Requires separate API (AJAX LocalSearch API)
 - Different types, formats and conventions
 - No unified concept of place
 - Different terms of service

Nearby Places

- Goals
 - Help developers connect users to places
 - Allow developers to harness Google's places DB
- Use Cases / Capabilities
 - Users searching for a place
 - Users looking to learn about a place
 - Users looking to have conversation about a place

Sneak Preview



Sample Nearby Places Application

Setting Up

- Bootstrap the V3 API

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

***Don't worry,
it's modular.***

Creating the Widget

- Declare the div
- Create the widget instance

```
<head>
  <script type="text/javascript">
    var buttonDiv = document.getElementById("button");
    widget = new google.maps.NearbyPlaces(buttonDiv);
  </script>
</head>

<body>
  <div id="button"></div>
</body>
```


Reading User Input

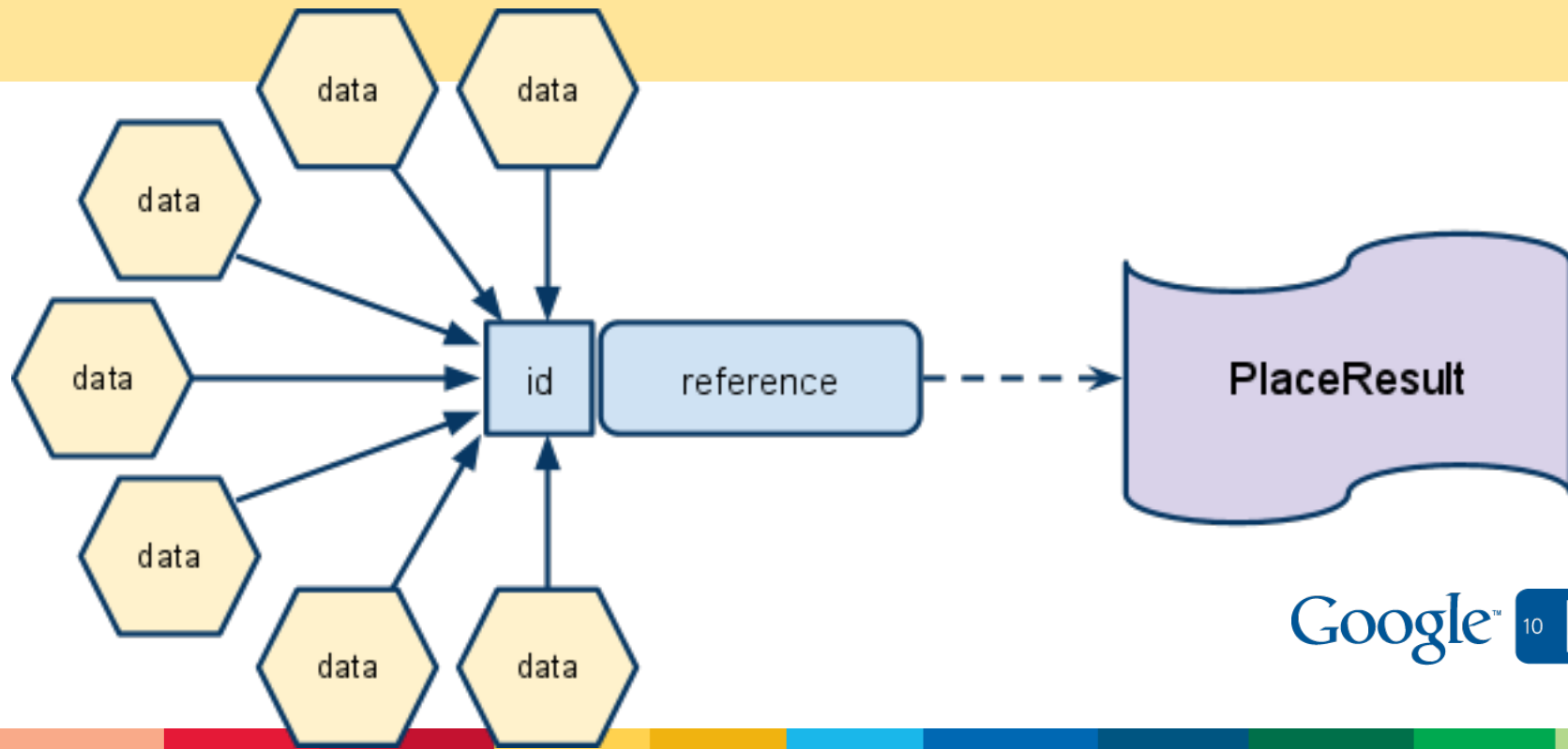
- Listen to `place_changed`
- Use `getPlace()`

```
google.maps.event.addListener(  
  widget, 'place_changed', function() {  
    var place = widget.getPlace();  
    if (place.name) {  
      ShowPlaceDialog(place);  
    }  
  });
```

Clustering and Retrieving Details

- Use **id** to aggregate content about a place
- Use **reference** to retrieve details of a place

```
var place = PlacesService.getPlaceByReference(  
  data.reference, function(place) {  
    // user place object here  
  });
```



Places Web Service

- HTTP (RESTful) Interface
- Non JS Applications
- Functionality analogous to NearbyPlace
- Outputs JSON and XML

Places Web Service

```
http://maps.google.com/maps/api/place/search/json?  
location=40.71,-73.95&radius=1500&sensor=true
```

```
{  
  "status": "OK",  
  "results": [ {  
    "name": "Williamsburg",  
    "types": [ "locality", "political" ],  
    "icon": "http://maps.gst.../geocode-71.png",  
    "reference": "CIRBAAAAXP...lHAPyHom2aG"  
  },  
  ...  
}
```

Places Web Service

```
http://maps.google.com/maps/api/place/details/xml?  
reference=REFERENCE_ID
```

```
<PlaceDetailsResponse>  
<status>OK</status>  
<result>  
  <name>Peter Luger Steakhouse</name>  
  <vicinity>Broadway, Brooklyn</vicinity>  
  <type>restaurant</type> <type>establishment</type>  
  <formatted_phone_number>...</formatted_phone_number>  
  <formatted_address>...</formatted_address>  
  ...  
</result>  
<PlaceDetailsResponse>
```

Recapping

- "Places" is a friendly way of talking location
- Location based services are mainstream
- Google is committed to location based tools
- Use Nearby Places to find, learn and talk about places
- Watch this space

David Wang

**Vice President of Business Development
Booyah Inc.**



myTOWN

Google Places
Web Service Integration

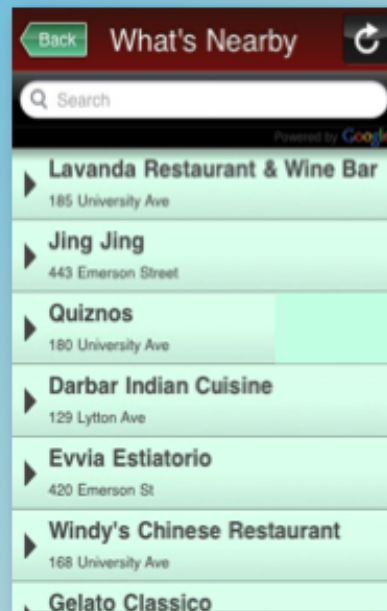


What is MyTown?

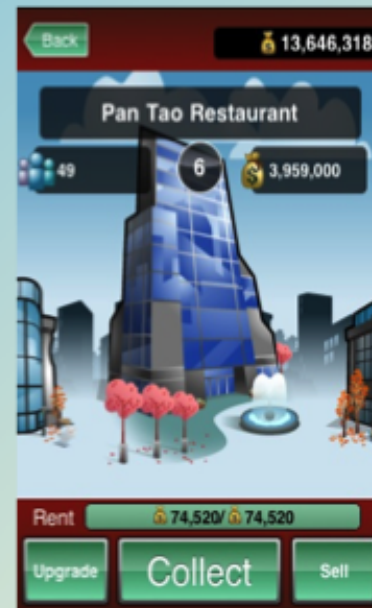
Real-World Game:
Monopoly with **Real** Locations.



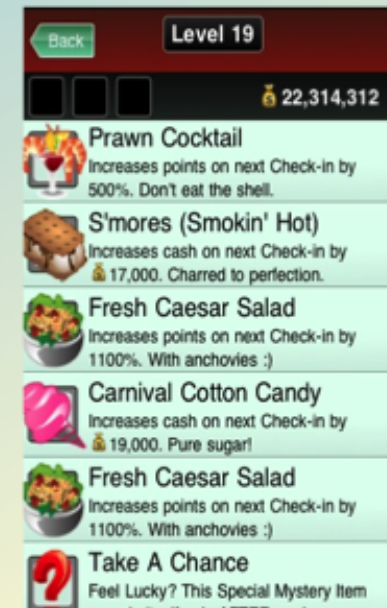
My Town



GPS Check-in

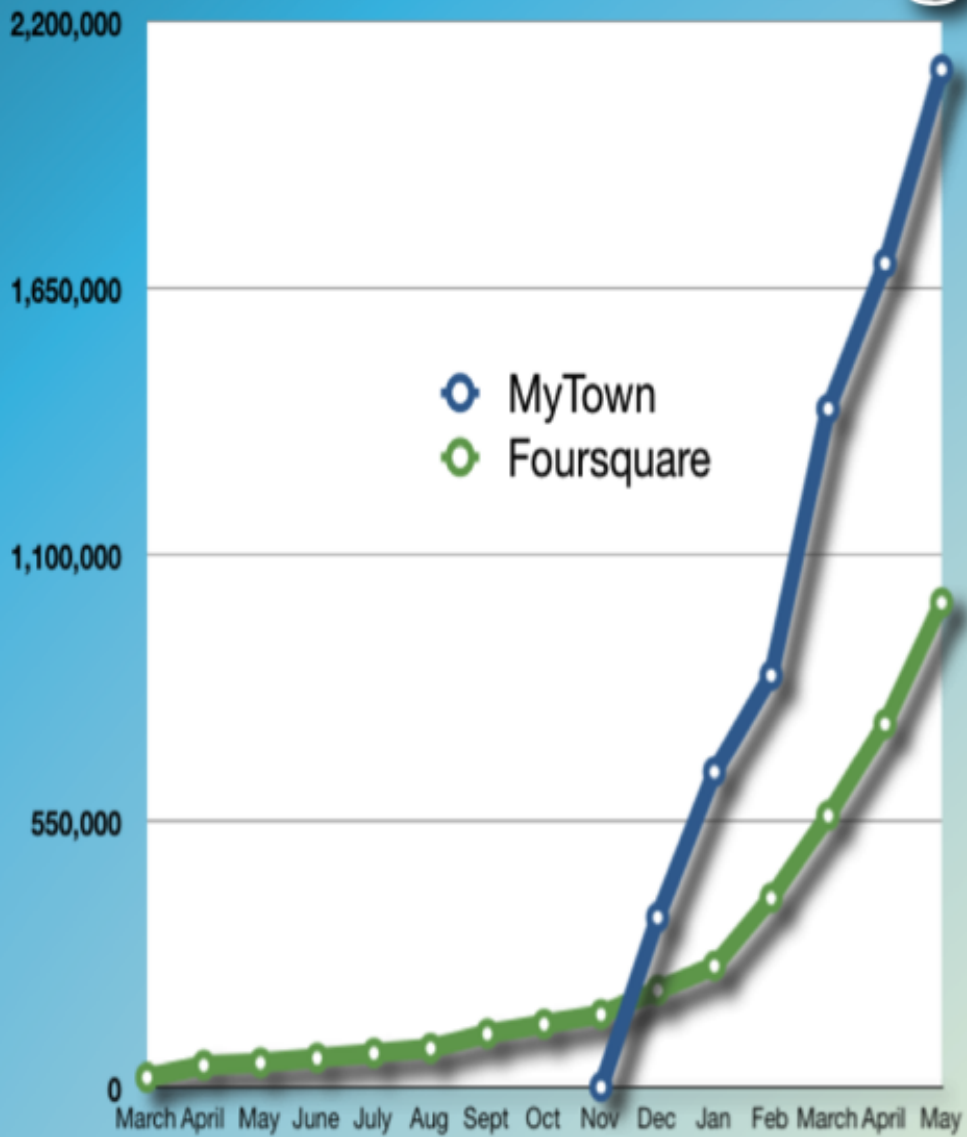


Collect Rent!



Virtual Items

myTOWN



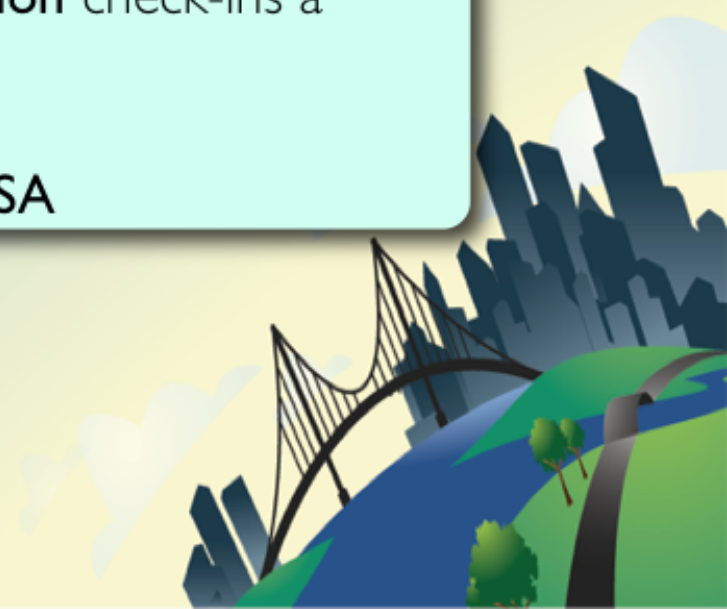
MyTown Stats

2.1 million Users in 5 months

1 hour+ daily playtime

100 million check-ins a month

100% USA



Booyah's Context

Leadership in Real-World Gaming

- Intersection of Entertainment, Social and Location Graphs

Strategic Decision to Focus on the Application Layer

- Don't re-create mapping/directory infrastructure

Seek Best-in-Class Partner

- International Expansion
- Strategic Roadmap



Implementation

Persistent Location Identification

- Enables MyTown GameData to map on Locations

Categorization

- Deeper Game Experience
- More interesting Signals

In-App HTML5 Place Page





myTOWN

Thank you



Google Developer Qualification



Chrome
Extensions



Gadgets



Search



App Engine



JS Maps API



KML



3D

Live Wave: Notes and Q&A

<http://bit.ly/ctP6P5>

Customize the UI

- Instantiate the widget

```
widget = new google.maps.NearbyPlaces();
```

- Show/hide the widget

```
function openPopup() {  
  widget.setPopupVisible(true);  
}
```


Customize the UI

- Listening to locality changes

```
google.maps.event.addListener(  
  widget, 'locality_changed', function() {  
    var locality = widget.getLocality();  
    updateUserLocality(locality);  
  });
```