Mapping Your Data

Tools to Map Your Data

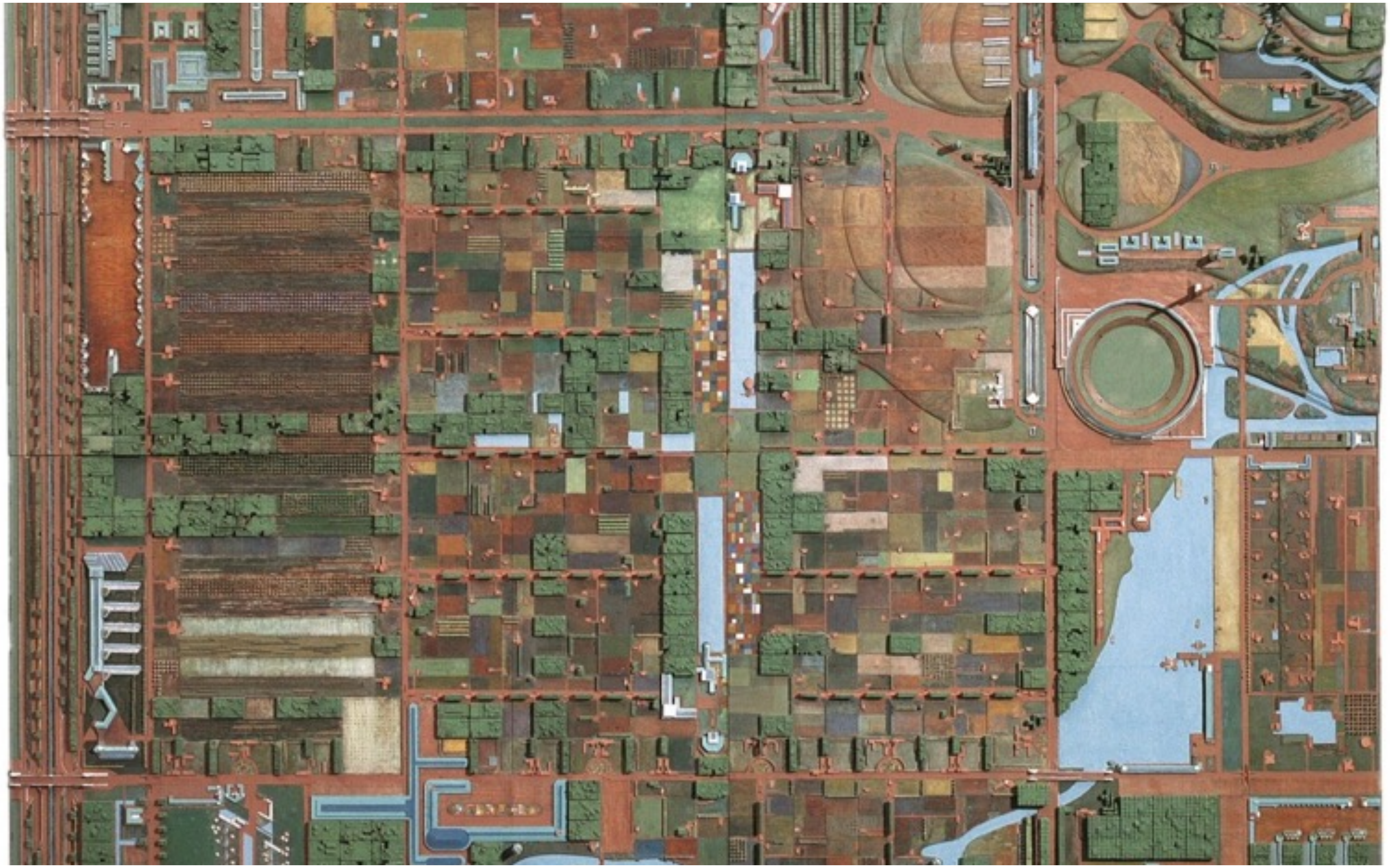
Maps!

- Showing data visually can help tell the story of data exploration

- Every map is a diagram, in the sense that a map is an abstracted representation of some but not all facets of a place (Taken from a very solid mapping article, http://www.spur.org/publications/urbanist-article/2012-11-09/grand-reductions-10-diagrams-changed-city-planning)

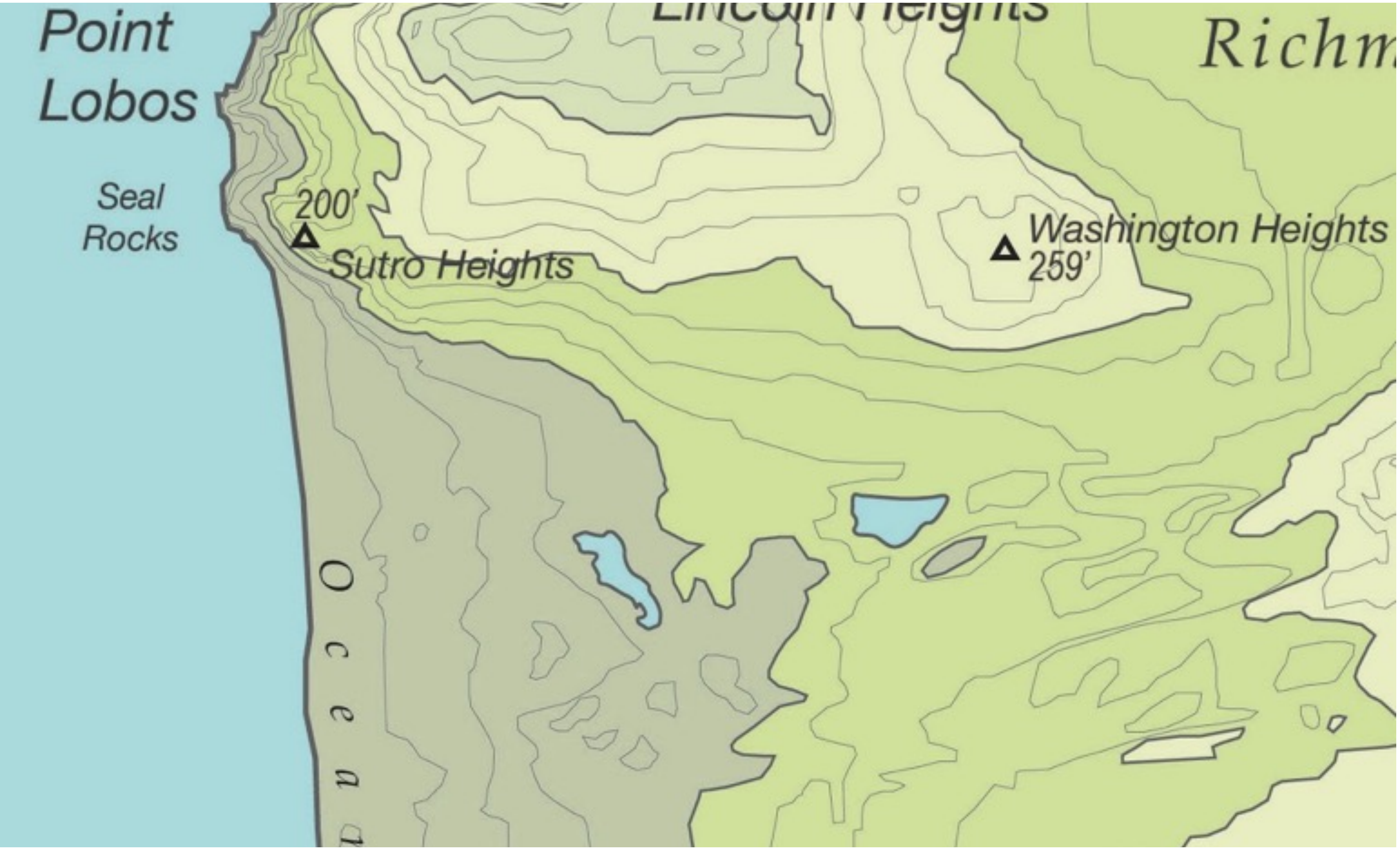
- Mapping can provide unseen insights into data

- Mapping can provide context to a data hypothesis

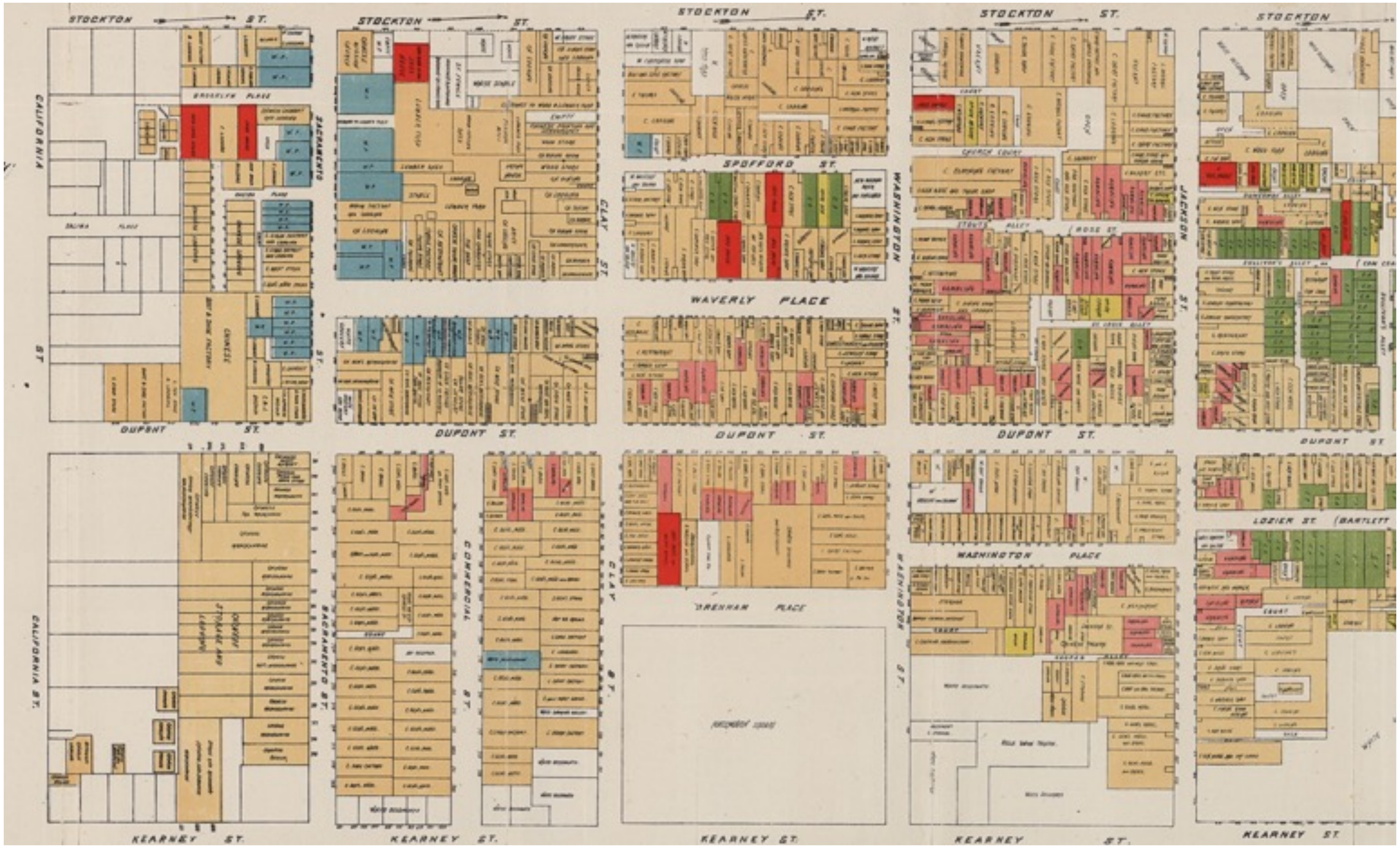


Mapping Styles

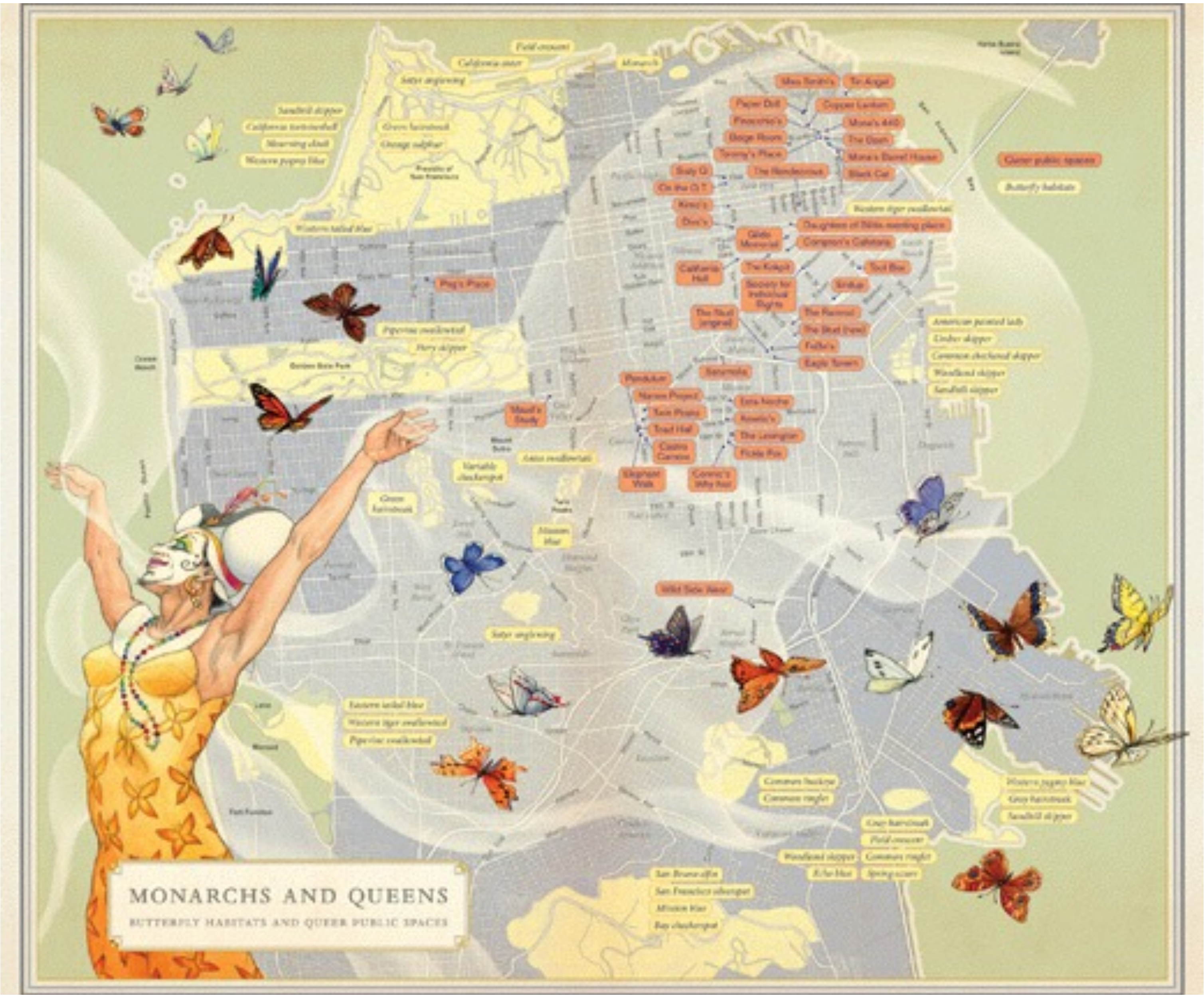
Various purposes
(Broadacre city imaginary map, 1932)



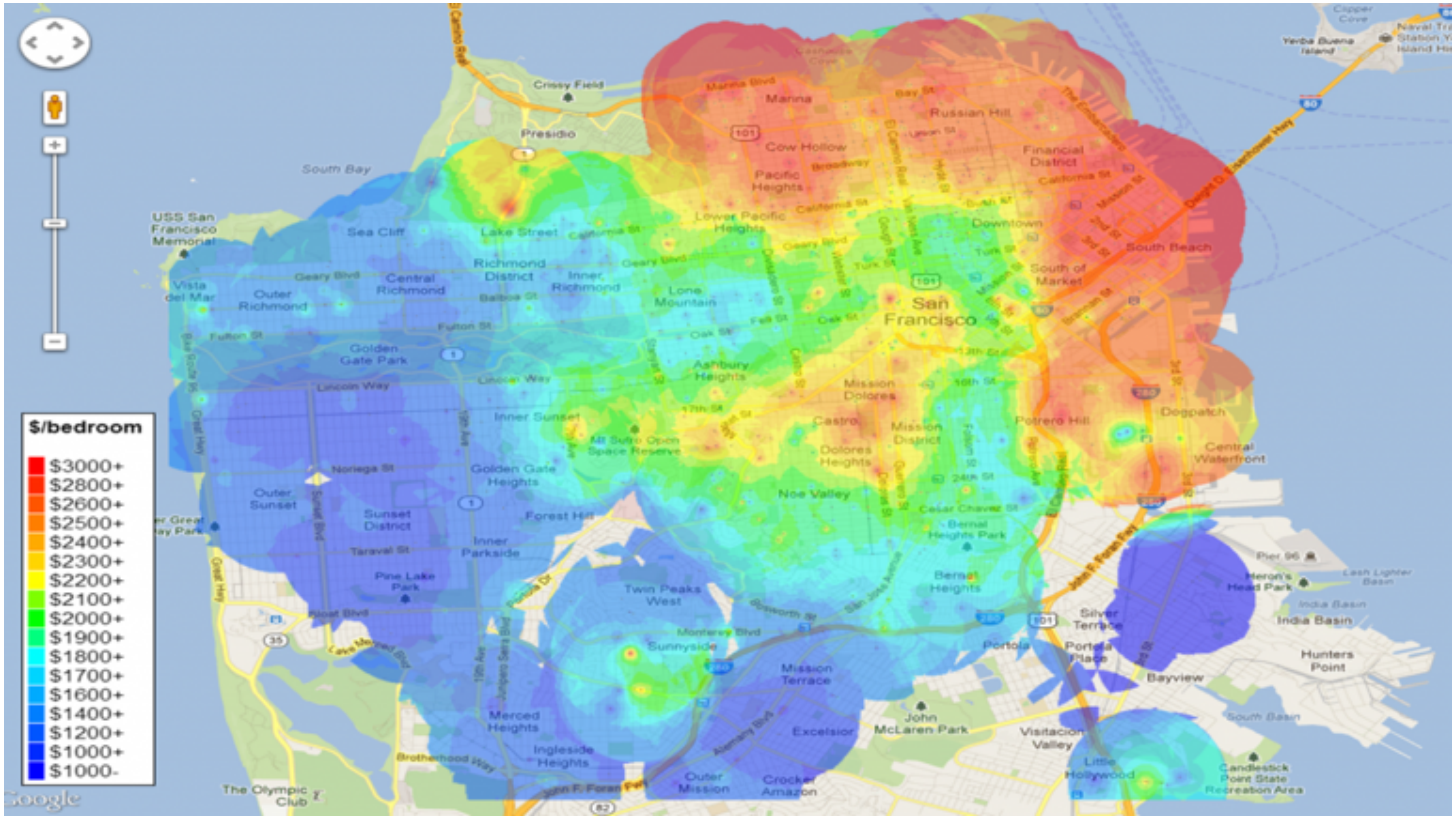
Topological Map



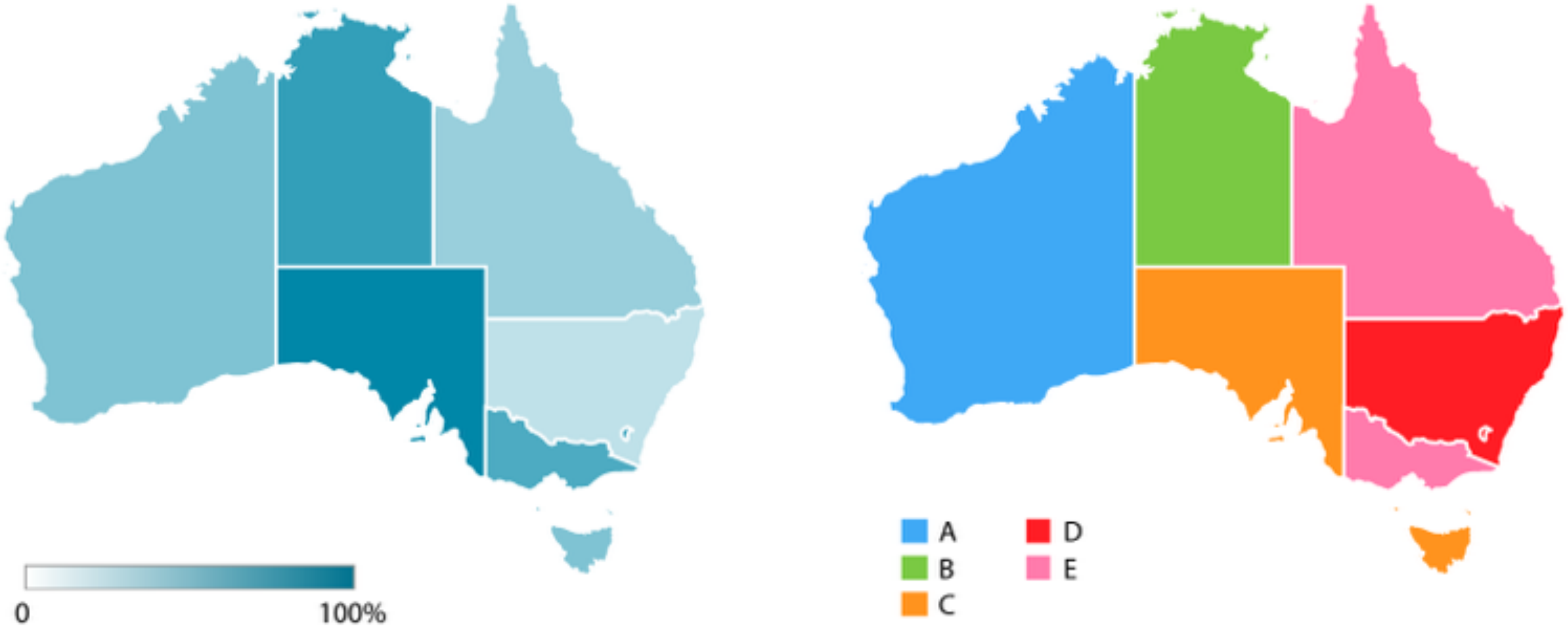
Zoning/Architectural Map



Thematic Map


$/bedroom
$3000+
$2800+
$2600+
$2500+
$2400+
$2300+
$2200+
$2100+
$2000+
$1900+
$1800+
$1700+
$1600+
$1400+
$1200+
$1000+
$1000-
Crissy Field
Presidio
South Bay
USS San Francisco Memorial
Marina
Russian Hill
Cow Hollow
Pacific Heights
Financial District
Lower Pacific Heights
Downtown
South Beach
Sea Cliff
Lake Street
Richmond District
Inner Richmond
Central Richmond
Outer Richmond
Vista del Mar
Lone Mountain
San Francisco
South of Market
Golden Gate Park
Ashbury Heights
Mission Dolores
Inner Sunset
Castro
Mission District
Potrero Hill
Dogpatch
Dolores Heights
Central Waterfront
Outer Sunset
Sunset District
Golden Gate Heights
Noe Valley
Forest Hill
Inner Parkside
Bernal Heights Park
Bernal Heights
Pine Lake Park
Twin Peaks West
Sunnyside
Silver Terrace
India Basin
Hunters Point
Portola
Portola Place
Bayview
Mission Terrace
Merced Heights
Ingleside Heights
Excelsior
John McLaren Park
Visitacion Valley
Little Hollywood
Outer Mission
Crocker Amazon
The Olympic Club
Candlestick Point State Recreation Area
South Basin
Yerba Buena Island
Google

Heat Map



Choropleth Map

Leaflet.js

is an open-source JavaScript library for mobile-friendly interactive maps

About Leaflet.js

- Easy to use

- Works efficiently across all major desktop and mobile platforms, can be extended with lots of plugins, and has a simple API

- Developed by Vladimir Agafonkin of Mapbox with a team of dedicated contributors (236 people so far!)

- Access the project's github here. https://github.com/Leaflet/Leaflet

- View Examples here: http://leafletjs.com/examples/quick-start.html
- View choropleth map example here: http://leafletjs.com/examples/choropleth.html

Leaflet (cont.)

- Leaflet is a great tool, but requires some javascript knowledge to use it

- Interacting with Pandas requires you to output panda data frames to .json and then have the leaflet library consume this information

- To have multiple map’s you might have to spin up a web server or use a series of plinkr’s (more concise)

Folium

YES. A leaflet library for python and specifically ipython notebooks!

Folium Features

- Allows us to combine our panda data frames while using most of the features of leaflet in python

- Builds on the data wrangling strengths of the Python ecosystem and the mapping strengths of the Leaflet.js library. Manipulate your data in Python, then visualize it in on a Leaflet map via Folium (From github)

- Supports geojson format and topojson format

Geojson sources

- GeoJSON is an open format for encoding information about geographic features using JSON format
- Easier to use and view than older shape files, .gis files, or .kml files
- One can download geojson files from a various sources to load into folium, to visualize countiy boundaries, city boundaries, any data that you would like to map.
- TopoJSON is an extension to GeoJSON that encodes topology in the geojson file
- http://eric.clst.org/Stuff/USGeoJSON
- https://catalog.data.gov/dataset?res_format=geojson
- https://data.sfgov.org/

```
{“type":"Feature",
 “properties”: {
      "name":"Nob Hill”,
      “cartodb_id”:25,
      “created_at":"2013-02-10T05:44:04.653Z",
     “updated_at”:"2013-02-10T05:44:04.923Z"},
 “geometry": {
    "type":"MultiPolygon","coordinates":[[[[-122.418609,37.78891],
    [-122.421954,37.788488],[-122.422333,37.79035],[-122.422697,37.792133],
    [-122.423435,37.795722],[-122.410257,37.7974],[-122.409062,37.791429],
    [-122.408805,37.790184],[-122.408797,37.790161],[-122.408789,37.790145],
    [-122.408912,37.790132],[-122.409094,37.790105],[-122.411745,37.789802],
    [-122.418609,37.78891]]]]}}
```

```
"type": "FeatureCollection",
    "features": [ {“type”:”Feature",
                       "geometry":{"type":"Point","coordinates":[-122.414495,37.769535]},
                         “properties":{"cartodb_id":12980,
                           “supervisor_district":9,
                           “caseid":3744148,
                           "point":"(37.769535428, -122.414494983)”,
                           “neighborhood”:”Mission”,"source":"Open311",
                           “opened":"2014-06-16T08:41:07Z",
                           "lon":-122.414494983,"lat":37.769535428}}
```

GeoJson Examples

Let’s give this library a shot