

#WWDC18

# Introducing MapKit JS

Session 212

Vicki Murley, MapKit JS Engineering Manager

Julien Quint, MapKit JS Engineer

Melody Kelly, MapKit JS Engineer



9:41



Mail



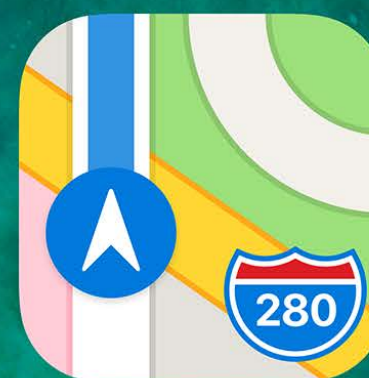
Calendar



Photos



Camera



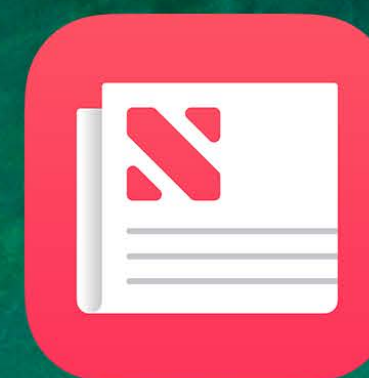
Maps



Clock



Weather



News



Home



Notes



Stocks



Reminders



TV



App Store



iTunes Store



iBooks



Health



Wallet



Settings



WWDC18

Overview Scholarships Attending More



Apple Worldwide Developers Conference  
San Jose, CA, June 4-8

# WWDC18

Overview Scholarships Attending More

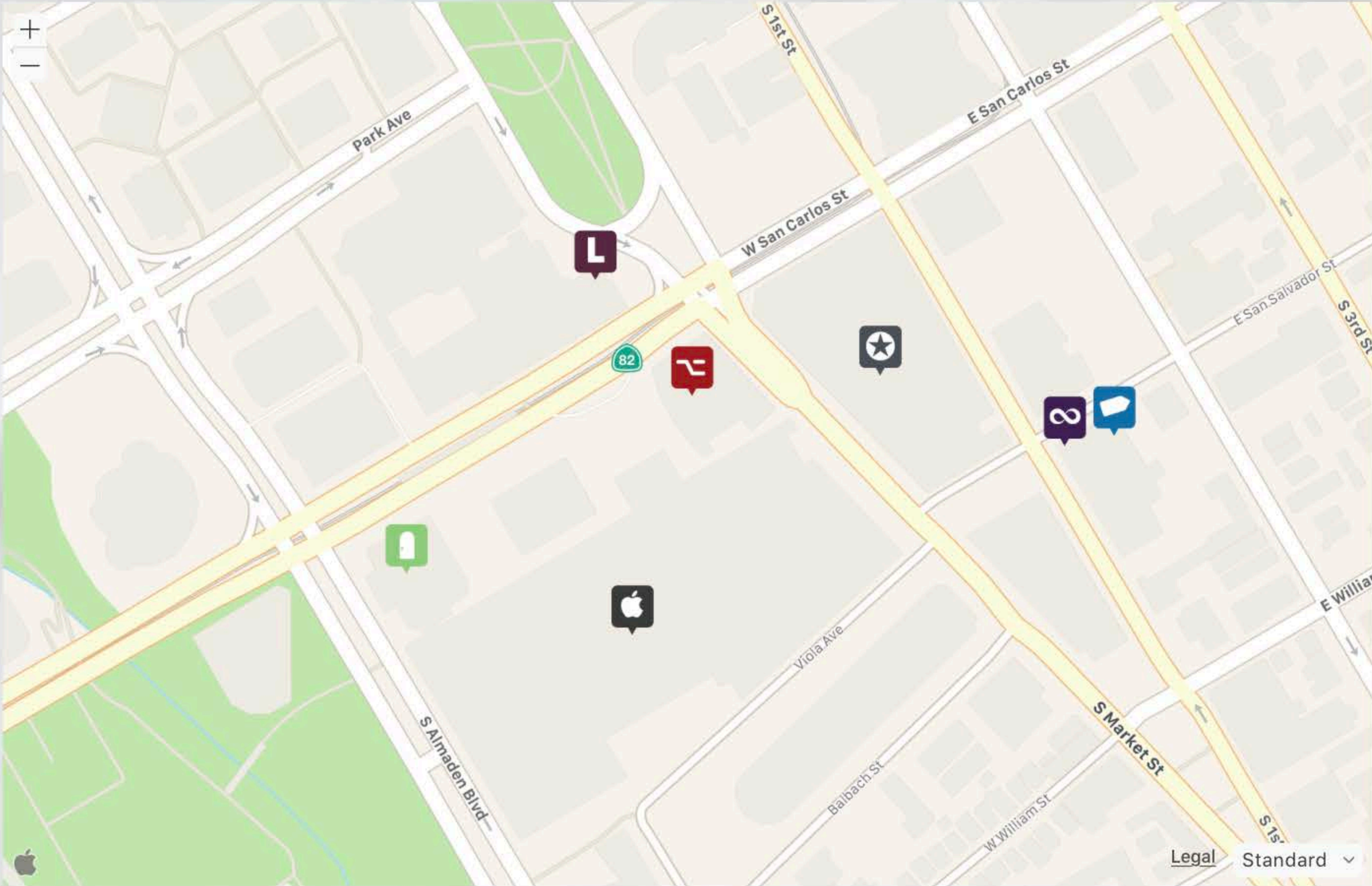


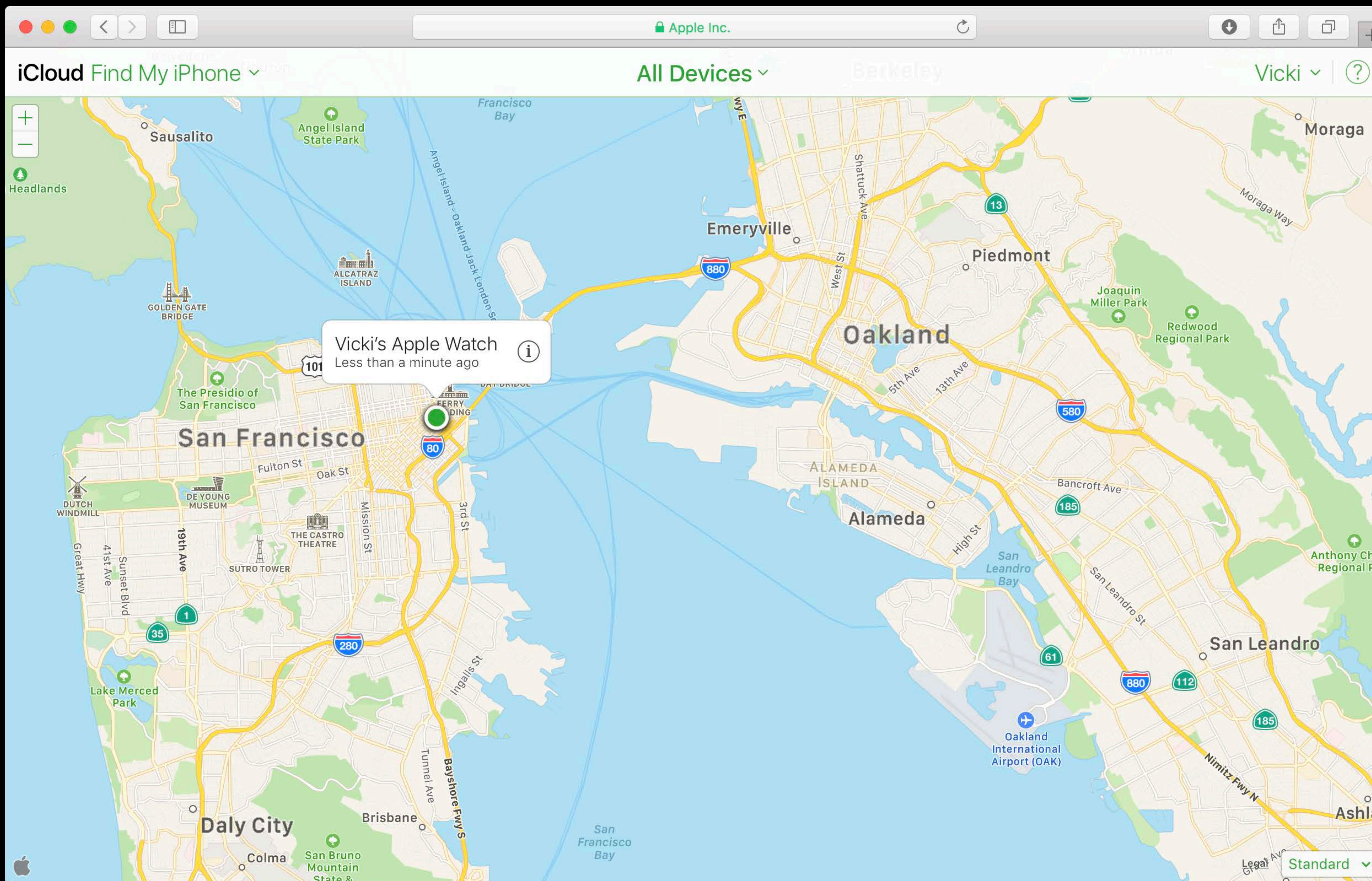
Apple Worldwide Developers Conference  
San Jose, CA, June 4-8



# Beyond WWDC.

In addition to Apple's Worldwide Developers Conference, a variety of other exciting developer events will take place throughout the week in San Jose. We'll be adding more events, so check back soon.



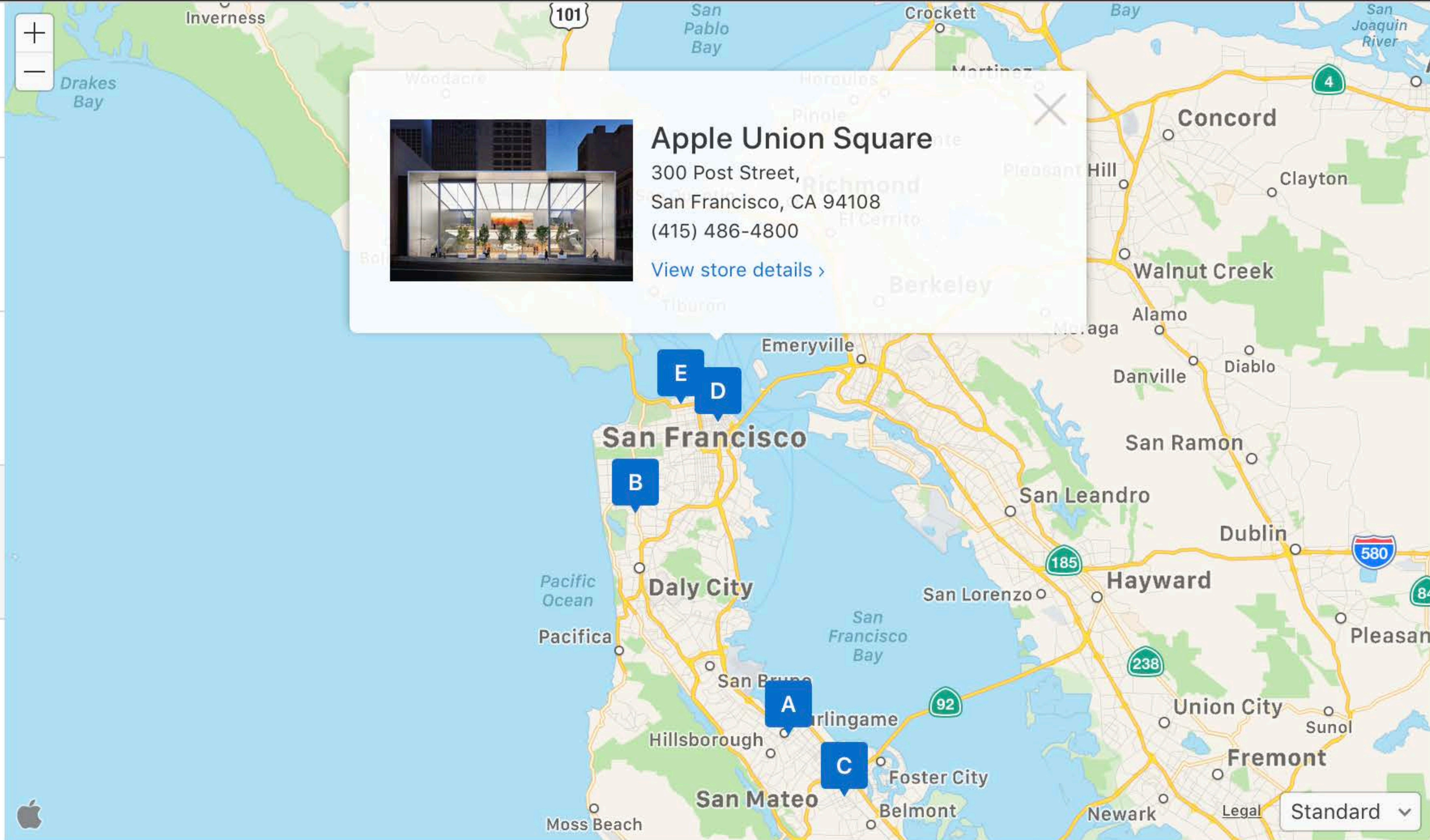


# Find a store

94044

[Complete store list >](#)

- A** **Apple Burlingame** 7.8 miles  
1301 Burlingame Avenue  
Burlingame, CA 94010  
[View store details >](#)
  - B** **Apple Stonestown** 8.7 miles  
3251 20th Avenue  
San Francisco, CA 94132  
[View store details >](#)
  - C** **Apple Hillsdale** 11.2 miles  
340 Hillsdale Shopping Center  
San Mateo, CA 94403  
[View store details >](#)
  - D** **Apple Union Square** 13.6 miles  
300 Post Street  
San Francisco, CA 94108  
[View store details >](#)
  - E** **Apple Chestnut Street** 14 miles  
2125 Chestnut Street  
San Francisco, CA 94123  
[View store details >](#)
- [Next >](#)



MapKit



NEW

# MapKit JS

NEW

# MapKit JS

BETA

250,000

Map initializations  
a.k.a. map views

25,000

Service requests

Geocoding, Search, Search autocomplete, and Directions

**Per Day**

# Per Day

Yes, really

Need more? Contact us.

<https://developer.apple.com/contact/request/mapkitjs>

# Get a MapKit JS Key

<https://developer.apple.com/account>



# Get a MapKit JS Key

<https://developer.apple.com/account>

Like, right now

# Why You'll ❤️ MapKit JS

# Why You'll ❤️ MapKit JS

Lets you unify on a single map provider for all platforms

# Why You'll ❤️ MapKit JS

Lets you unify on a single map provider for all platforms

MapKit JS APIs are inspired by native MapKit, and familiar to web developers

# Why You'll ❤️ MapKit JS

Lets you unify on a single map provider for all platforms

MapKit JS APIs are inspired by native MapKit, and familiar to web developers

Provides the native Apple Maps experience in web pages

- Localized and accessible
- Beautiful cartography
- Support for native gestures

# Why You'll ❤️ MapKit JS

Lets you unify on a single map provider for all platforms

MapKit JS APIs are inspired by native MapKit, and familiar to web developers

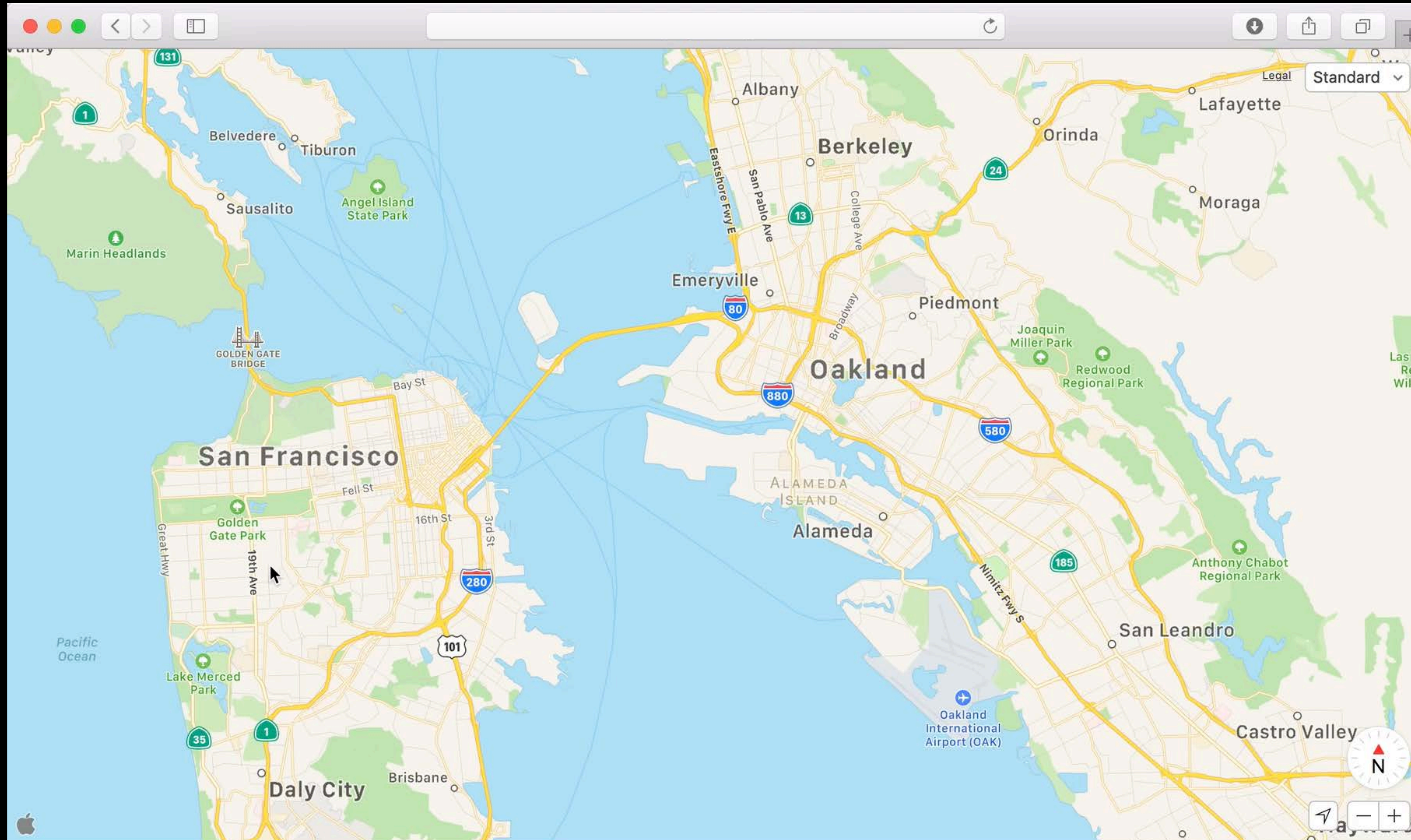
Provides the native Apple Maps experience in web pages

- Localized and accessible
- Beautiful cartography
- Support for native gestures

Uses adaptive rendering modes

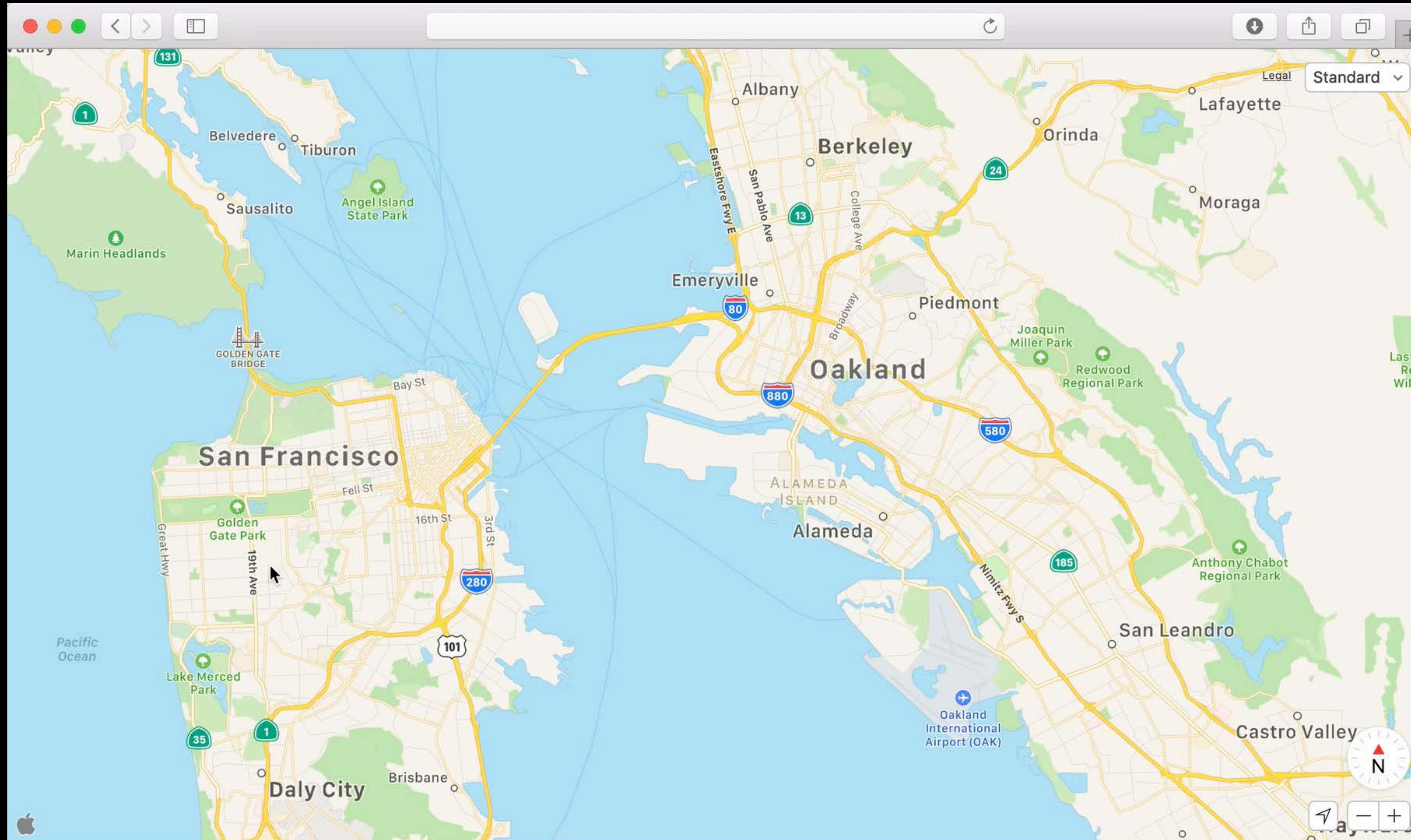
# Client-Side Rendering (CSR)

Full WebGL map rendering on the client



# Client-Side Rendering (CSR)

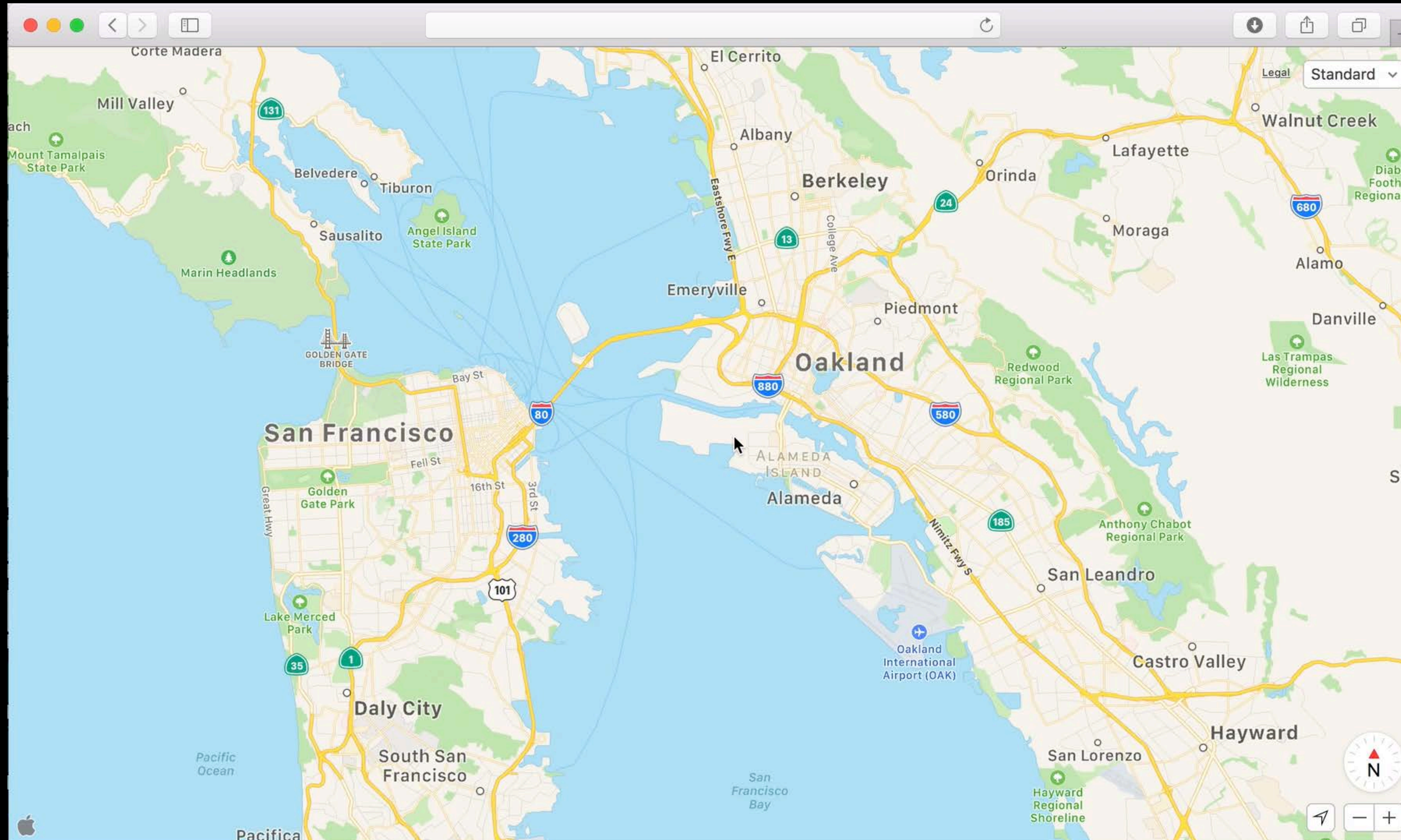
Full WebGL map rendering on the client





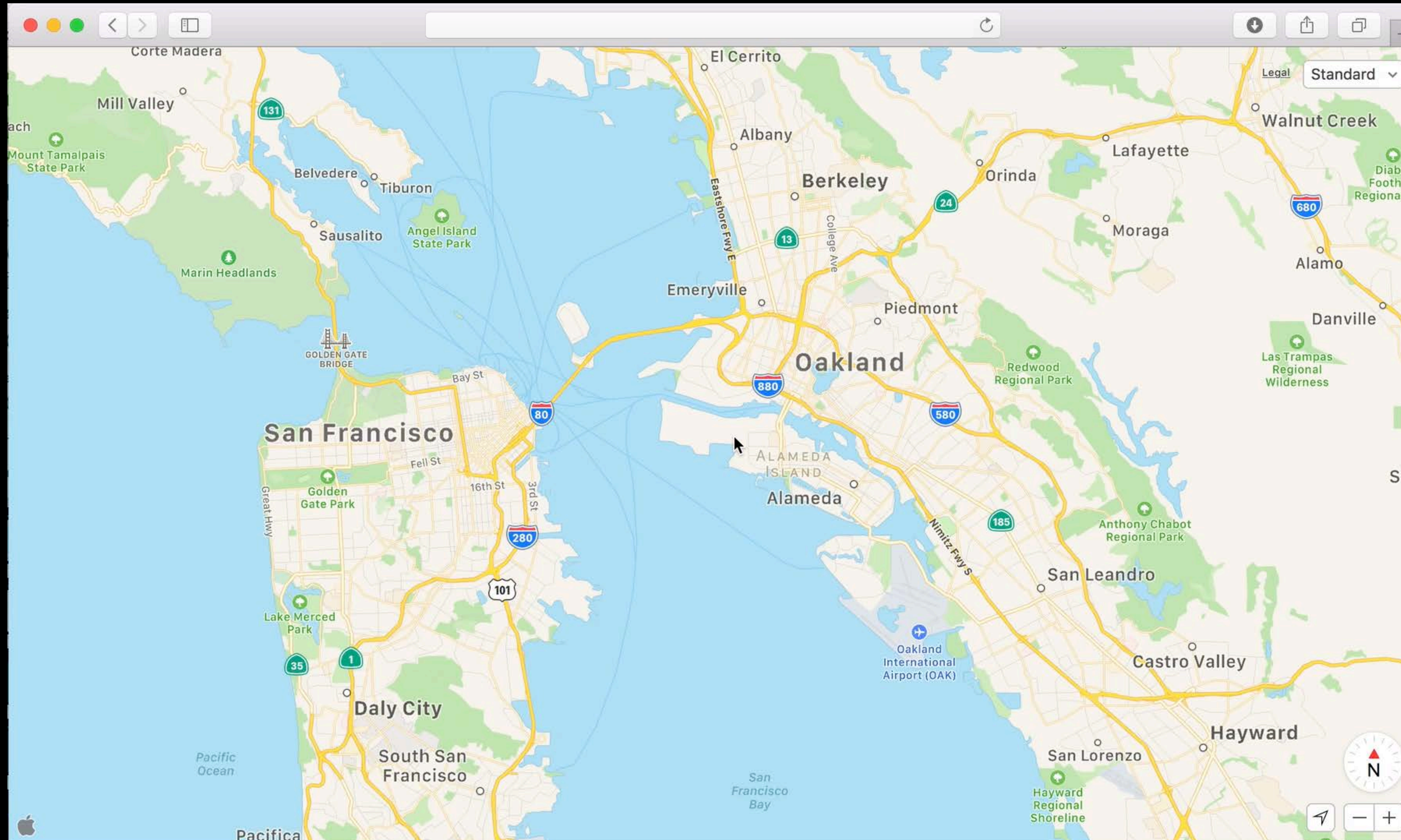
# Client-Side Rendering (CSR)

Full control over map labels for rotation



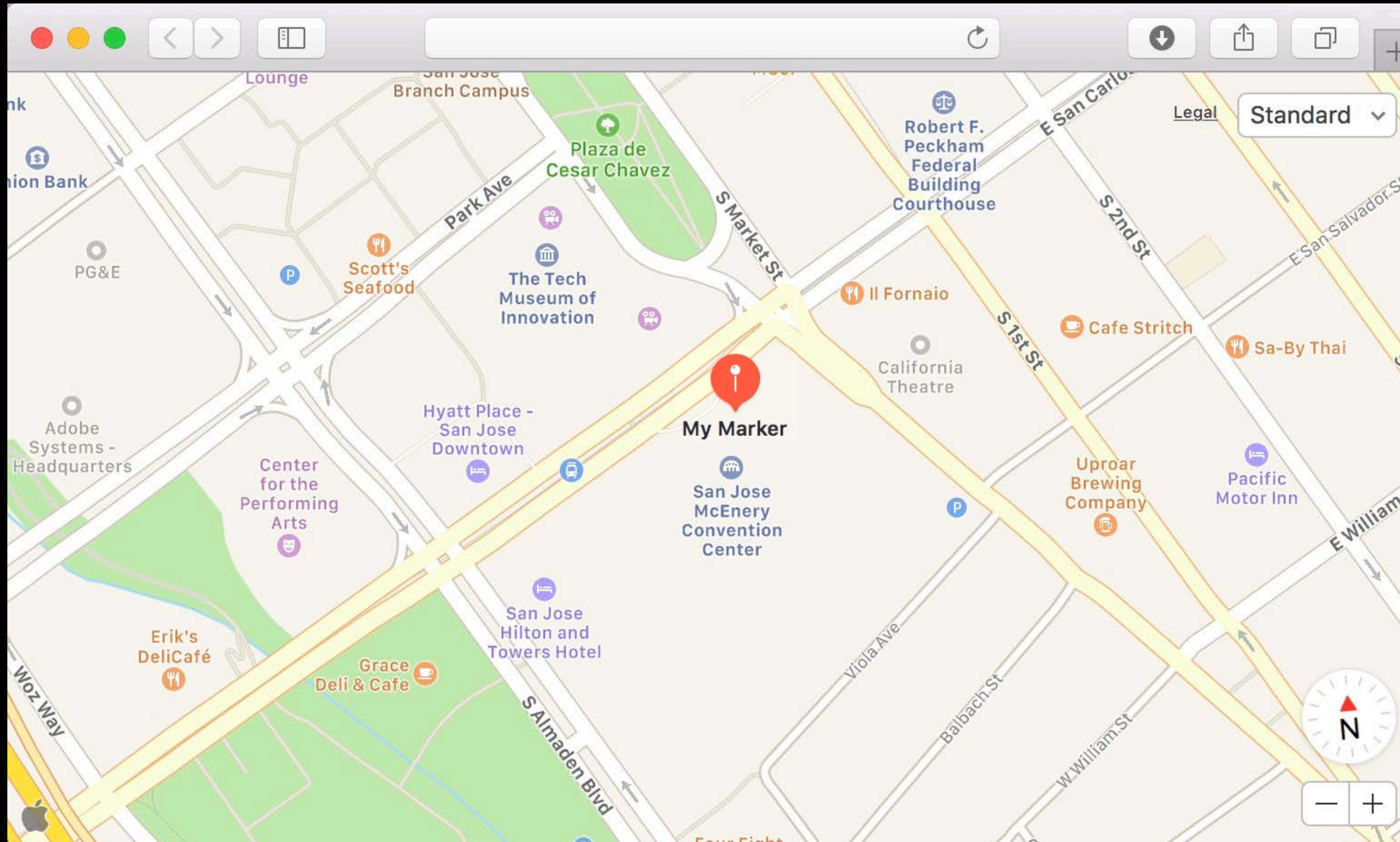
# Client-Side Rendering (CSR)

Full control over map labels for rotation



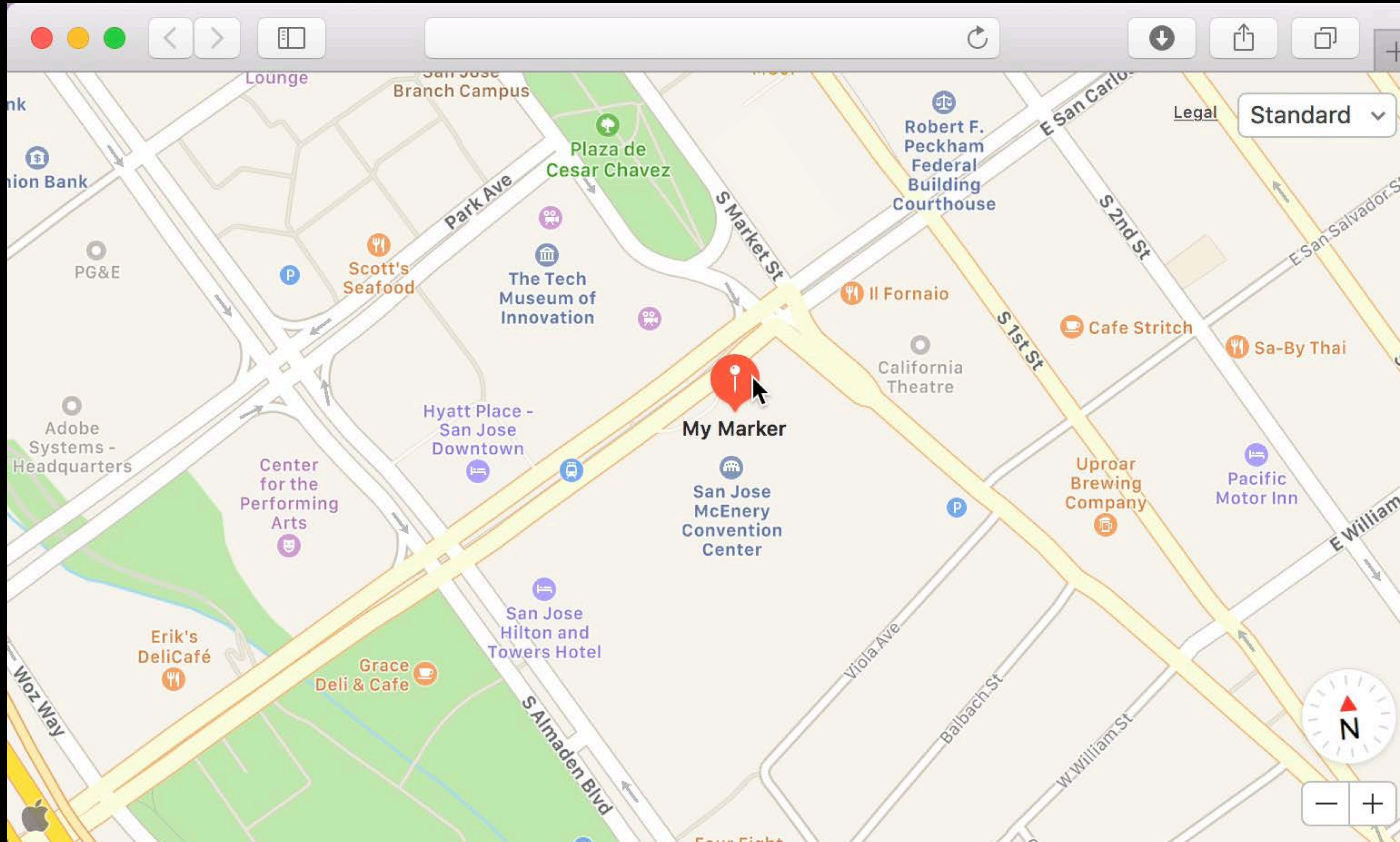
# Client-Side Rendering (CSR)

Full control over map labels for annotation label collisions



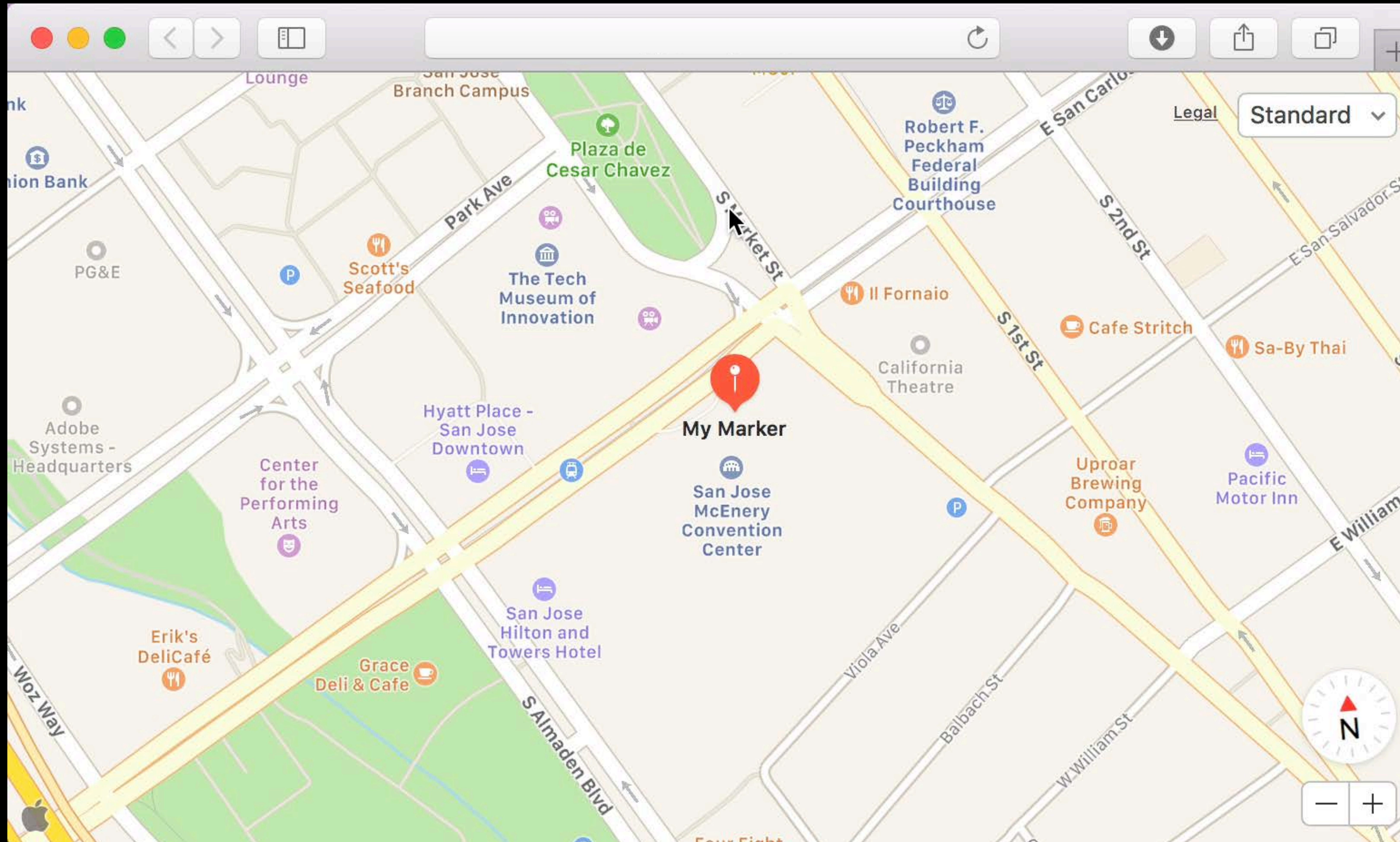
# Client-Side Rendering (CSR)

Full control over map labels for annotation label collisions



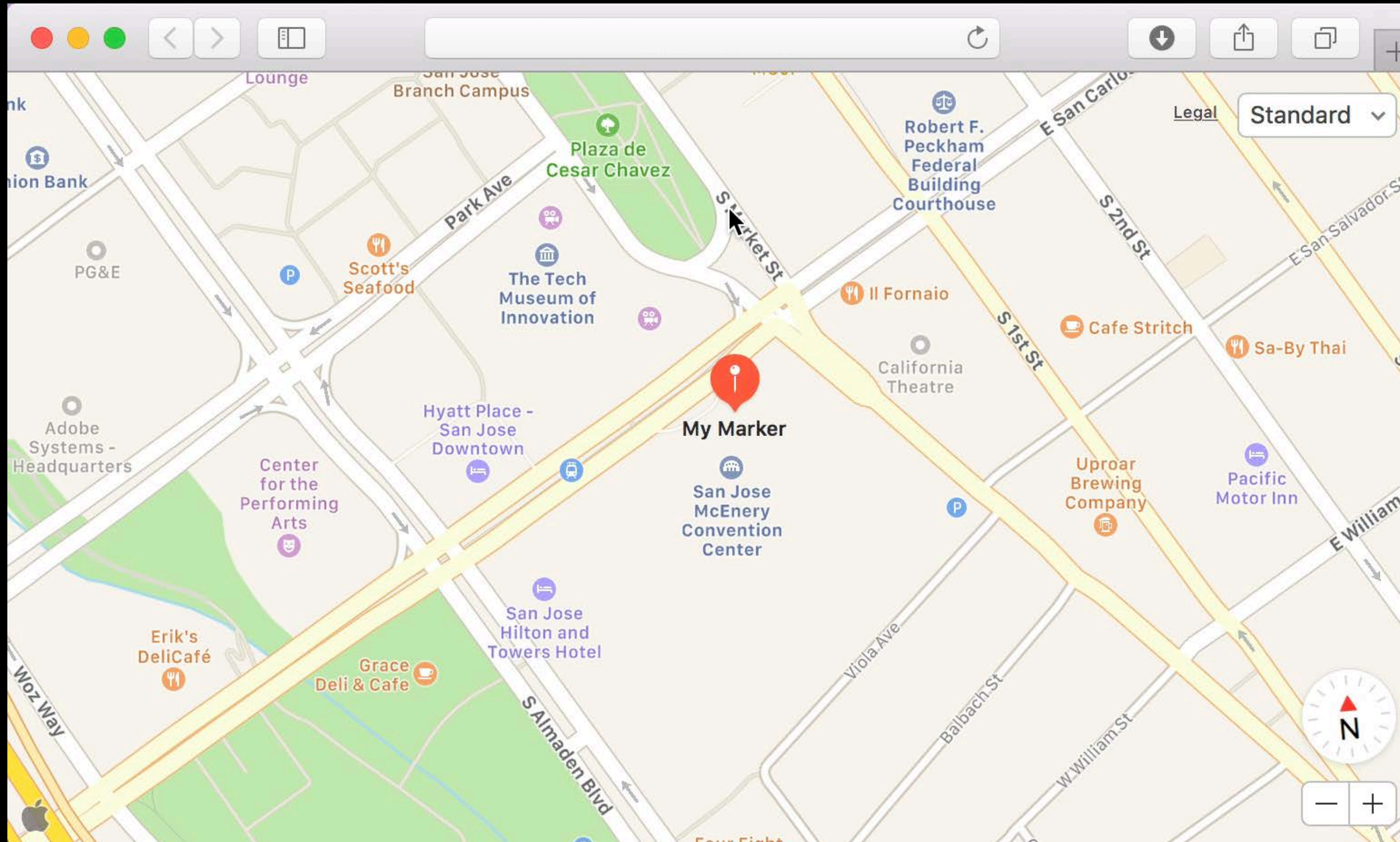
# Client-Side Rendering (CSR)

Full control over map labels for annotation label collisions



# Client-Side Rendering (CSR)

Full control over map labels for annotation label collisions



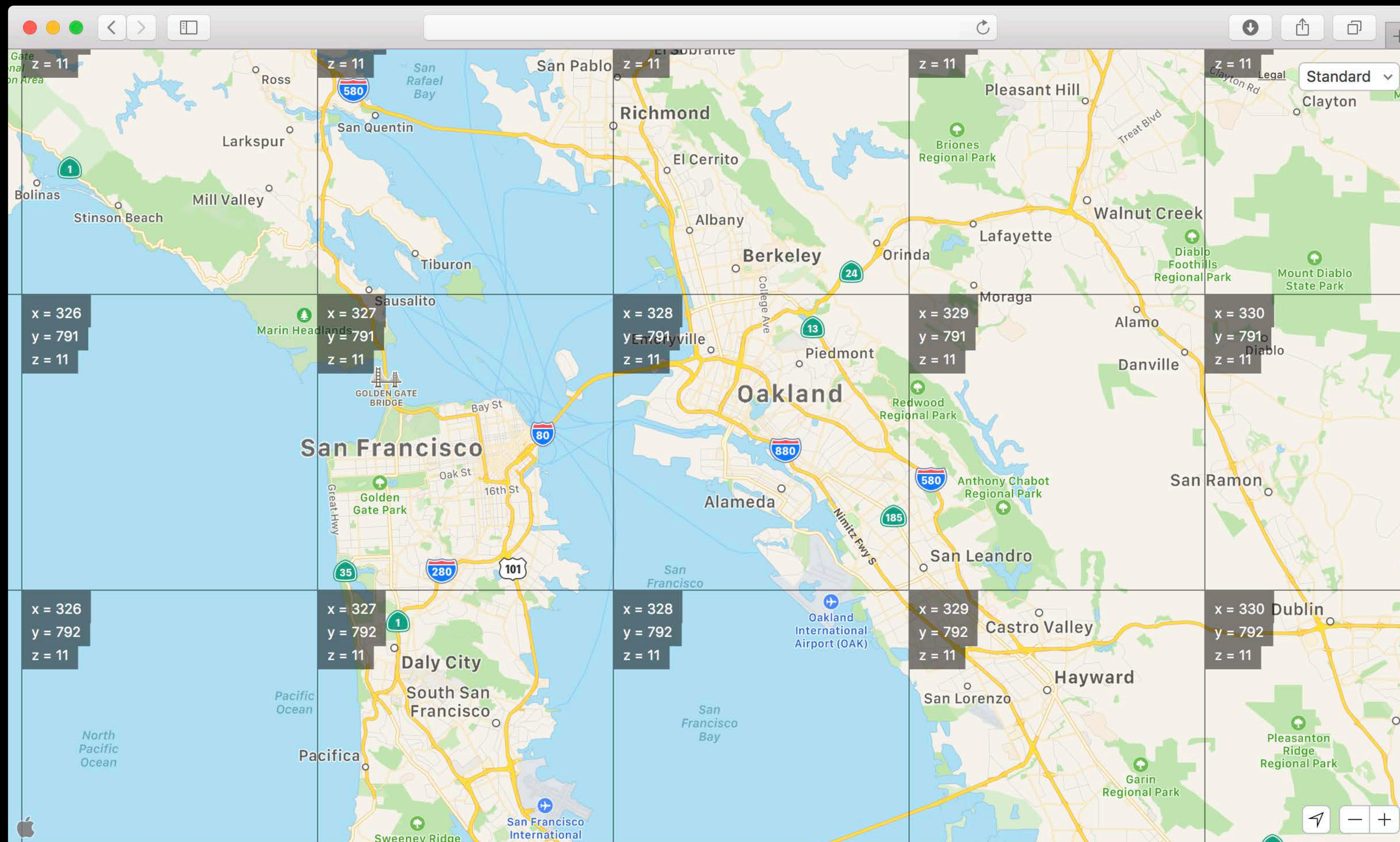
Web code can run anywhere

Not all devices are created equal



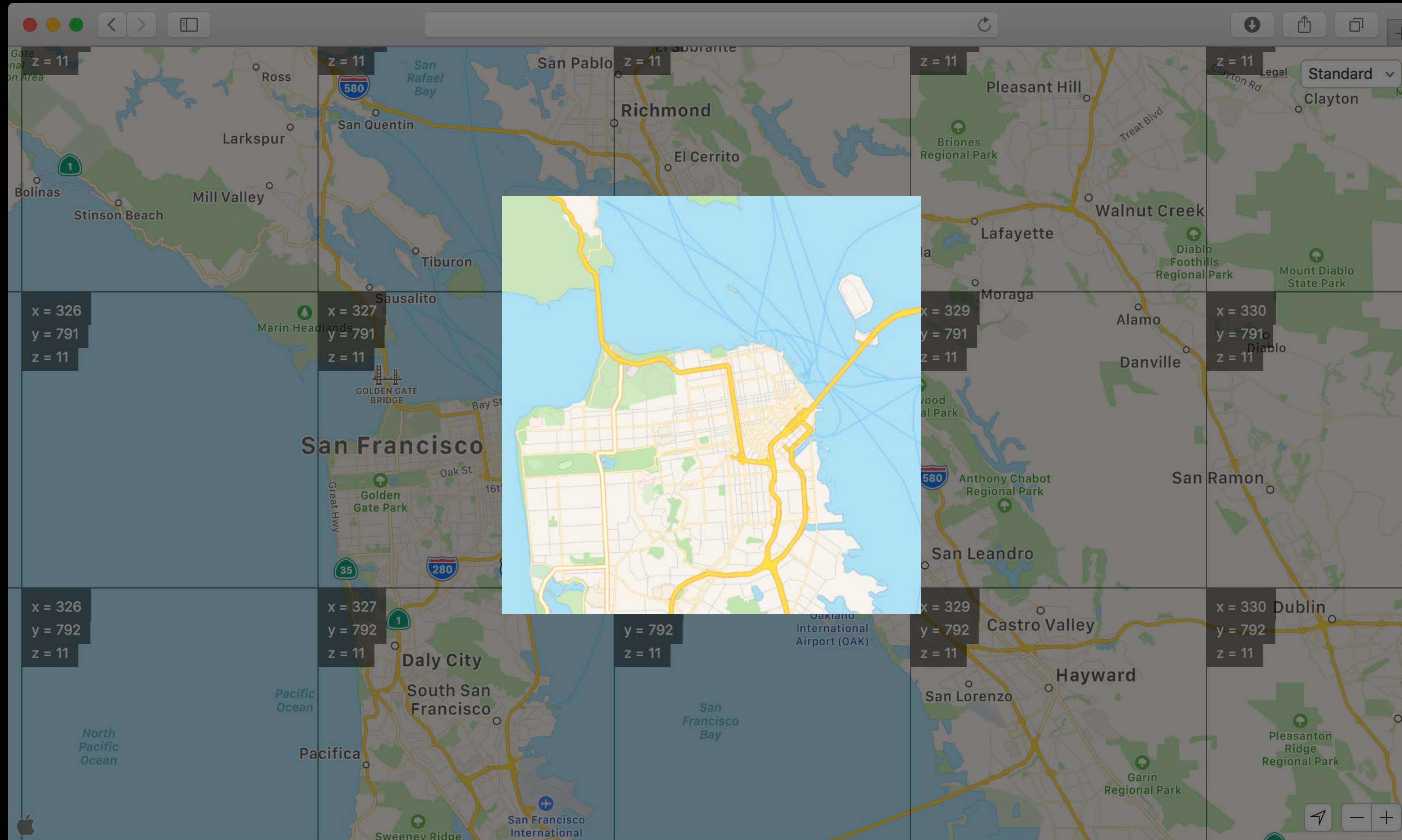
# Labels-Only Client-Side Rendering (LO-CSR)

Optimized for low-performance configurations



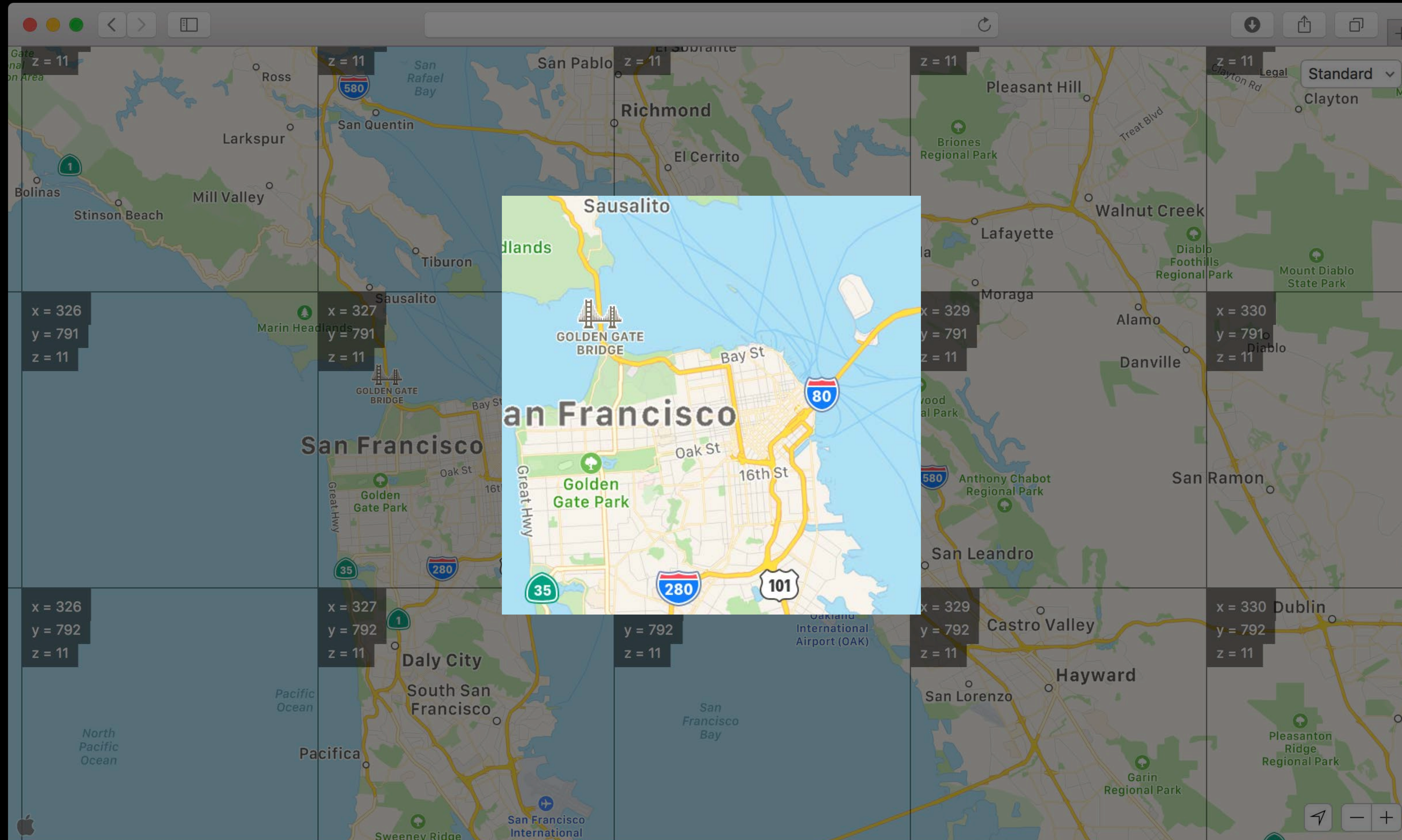
# Labels-Only Client-Side Rendering (LO-CSR)

Optimized for low-performance configurations



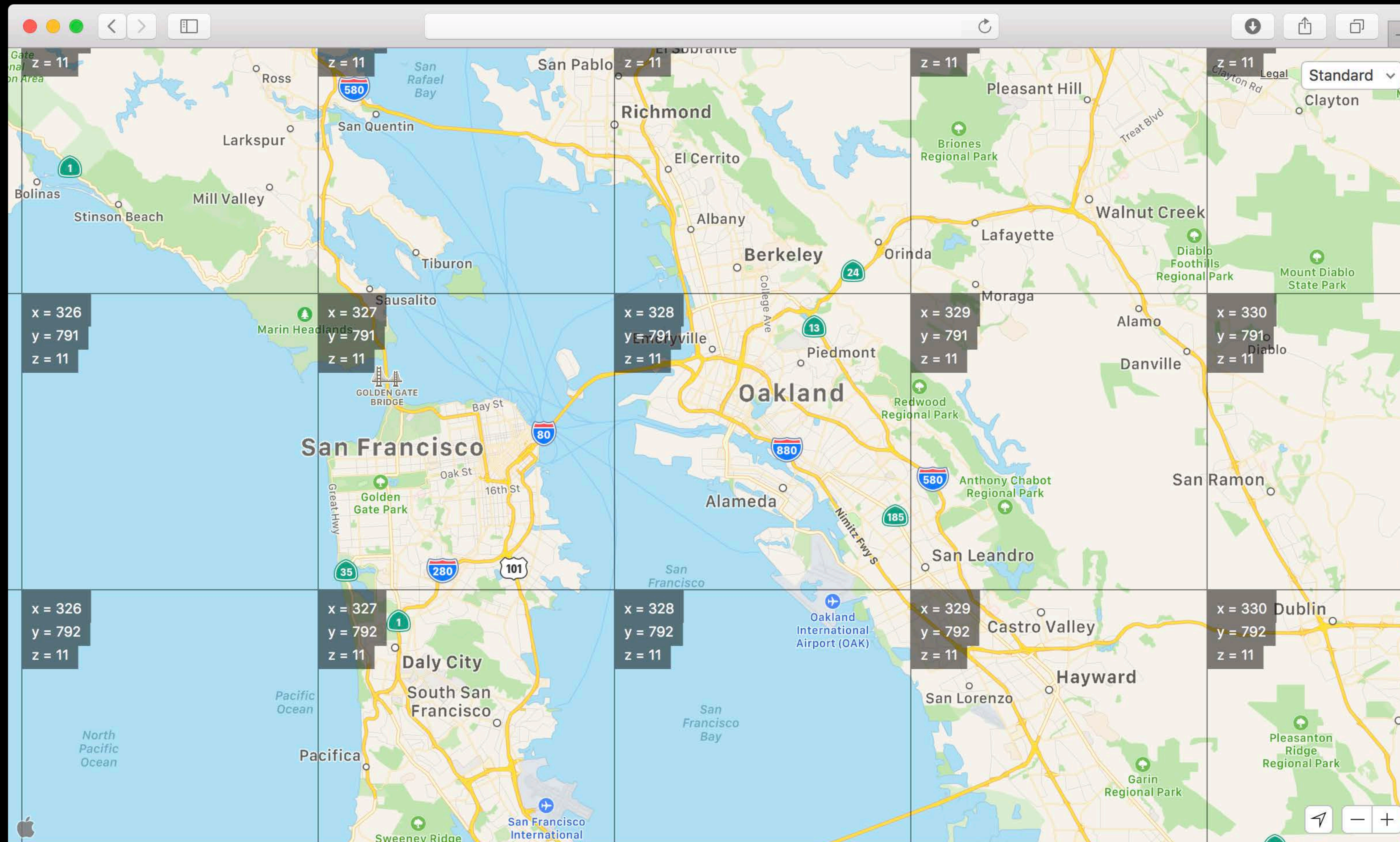
# Labels-Only Client-Side Rendering (LO-CSR)

Optimized for low-performance configurations



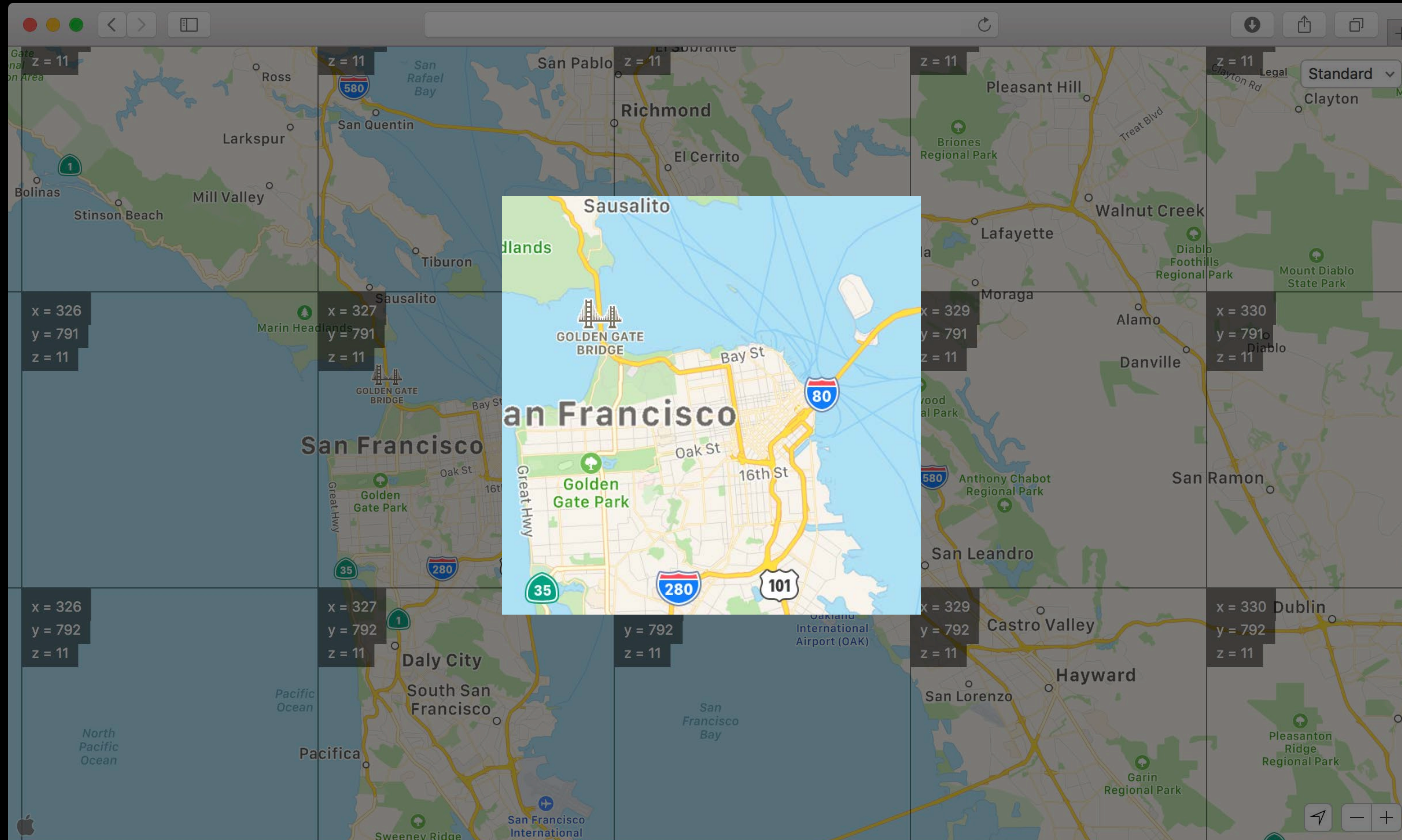
# Server-Side Rendering (SSR)

No WebGL? No problem.



# Server-Side Rendering (SSR)

No WebGL? No problem.



# Adaptive Rendering Modes

# Adaptive Rendering Modes

An ideal mode for every client configuration

# Adaptive Rendering Modes

An ideal mode for every client configuration

The best mode is automatically chosen



# Adaptive Rendering Modes

An ideal mode for every client configuration

The best mode is automatically chosen

Most users will get either CSR or LO-CSR

# Using MapKit JS on Your Website

Julien Quint, MapKit JS Engineer

# Using MapKit JS on Your Website

Agenda

# Using MapKit JS on Your Website

Agenda

Setting up your map

# Using MapKit JS on Your Website

## Agenda

Setting up your map

Navigating and annotating the map

# Using MapKit JS on Your Website

## Agenda

Setting up your map

Navigating and annotating the map

Enabling rich interactions with services

# Using MapKit JS on Your Website

## Agenda

Setting up your map

Navigating and annotating the map

Enabling rich interactions with services

# Setting Up Your Map

## Step 1: Import mapkit.js

```
<script src="https://cdn.apple-mapkit.com/mk/5.0.x/mapkit.js"></script>
```

```
<div id="map" style="height: 600px; width: 800px;"></div>
```

```
<script>
```

```
  // Initialize MapKit JS
```

```
  mapkit.init({
```

```
    authorizationCallback: done => fetch(...).then(...).then(done);
```

```
  });
```

```
  let map = new mapkit.Map("map");
```

```
</script>
```



# Setting Up Your Map

## Step 1: Import mapkit.js

```
<script src="https://cdn.apple-mapkit.com/mk/5.0.x/mapkit.js"></script>
```

```
<div id="map" style="height: 600px; width: 800px;"></div>
```

```
<script>
```

```
  // Initialize MapKit JS
```

```
  mapkit.init({
```

```
    authorizationCallback: done => fetch(...).then(...).then(done);
```

```
  });
```

```
  let map = new mapkit.Map("map");
```

```
</script>
```

# Setting Up Your Map

## Step 2: Create a map container

```
<script src="https://cdn.apple-mapkit.com/mk/5.0.x/mapkit.js"></script>
```

```
<div id="map" style="height: 600px; width: 800px;"></div>
```

```
<script>
```

```
  // Initialize MapKit JS
```

```
  mapkit.init({
```

```
    authorizationCallback: done => fetch(...).then(...).then(done);
```

```
  });
```

```
  let map = new mapkit.Map("map");
```

```
</script>
```

# Setting Up Your Map

## Step 2: Create a map container

```
<script src="https://cdn.apple-mapkit.com/mk/5.0.x/mapkit.js"></script>
```

```
<div id="map" style="height: 600px; width: 800px;"></div>
```

```
<script>
```

```
  // Initialize MapKit JS
```

```
  mapkit.init({
```

```
    authorizationCallback: done => fetch(...).then(...).then(done);
```

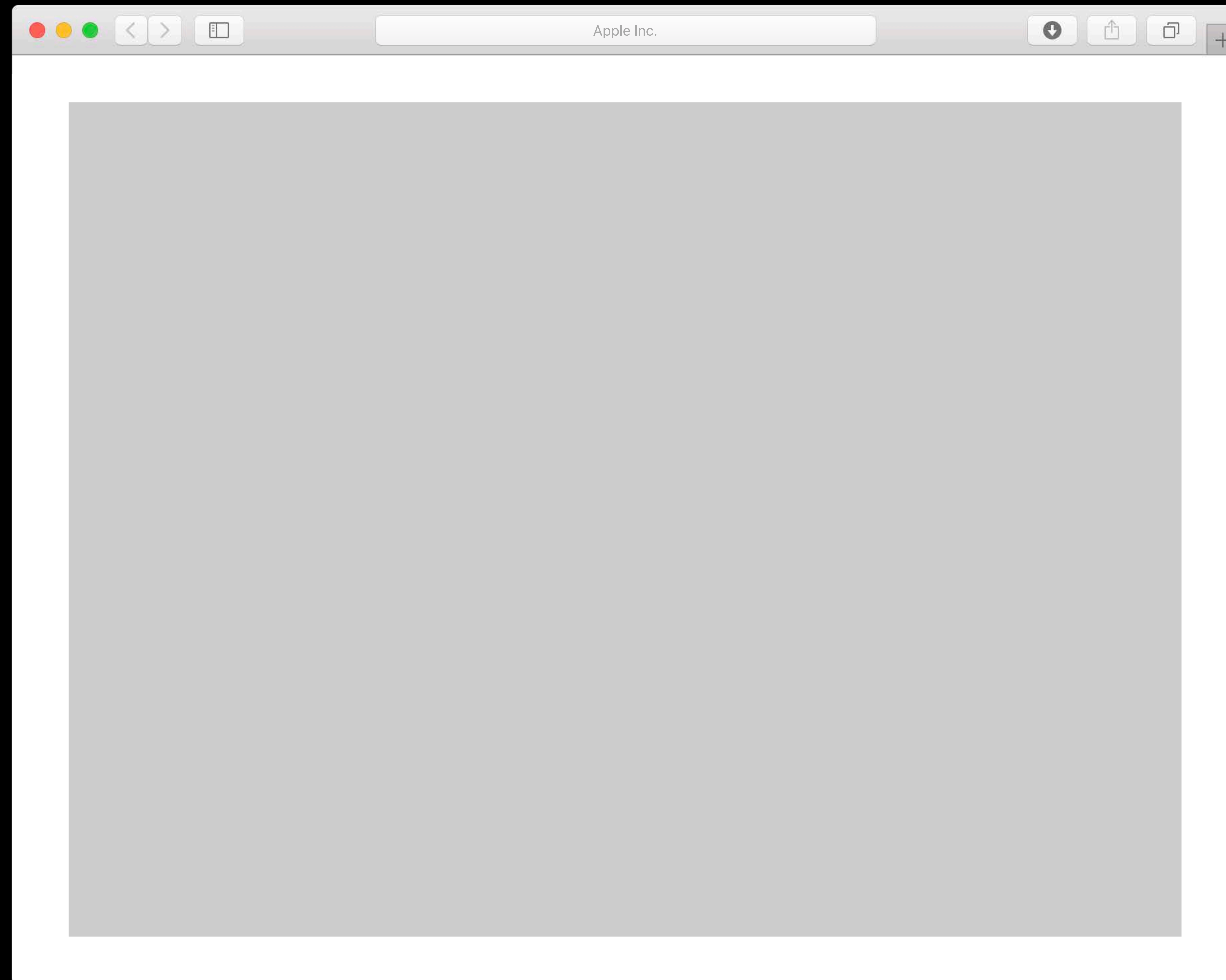
```
  });
```

```
  let map = new mapkit.Map("map");
```

```
</script>
```

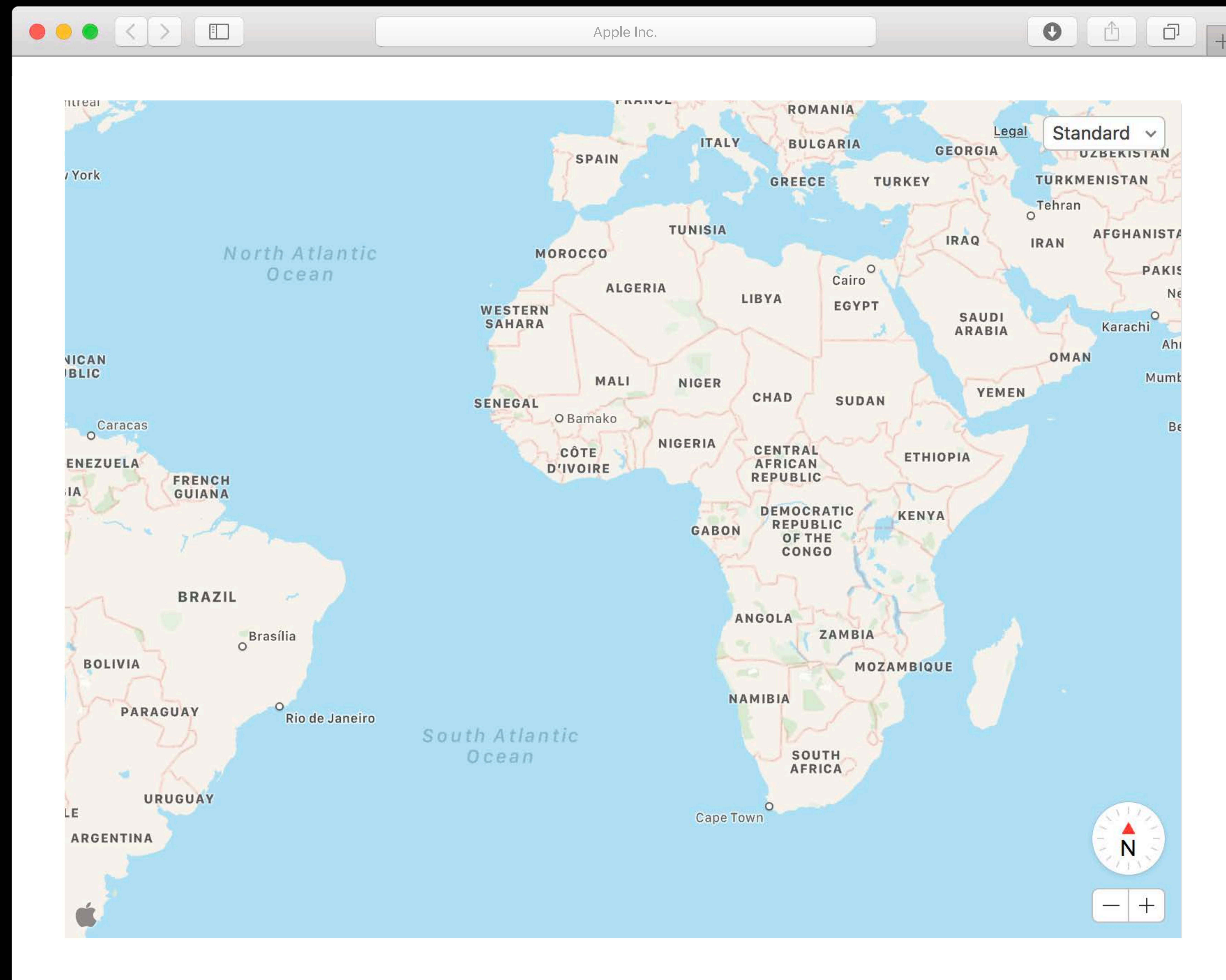
# Setting Up Your Map

Step 2: Create a map container



# Setting Up Your Map

## Step 2: Create a map container



# Setting Up Your Map

## Step 3: Initialize MapKit JS

```
<script src="https://cdn.apple-mapkit.com/mk/5.0.x/mapkit.js"></script>
```

```
<div id="map" style="height: 600px; width: 800px;"></div>
```

```
<script>
```

```
  // Initialize MapKit JS
```

```
  mapkit.init({
```

```
    authorizationCallback: done => fetch(...).then(...).then(done);
```

```
  });
```

```
  let map = new mapkit.Map("map");
```

```
</script>
```

# Setting Up Your Map

## Step 4: Create a Map object

```
<script src="https://cdn.apple-mapkit.com/mk/5.0.x/mapkit.js"></script>
```

```
<div id="map" style="height: 600px; width: 800px;"></div>
```

```
<script>
```

```
  // Initialize MapKit JS
```

```
  mapkit.init({
```

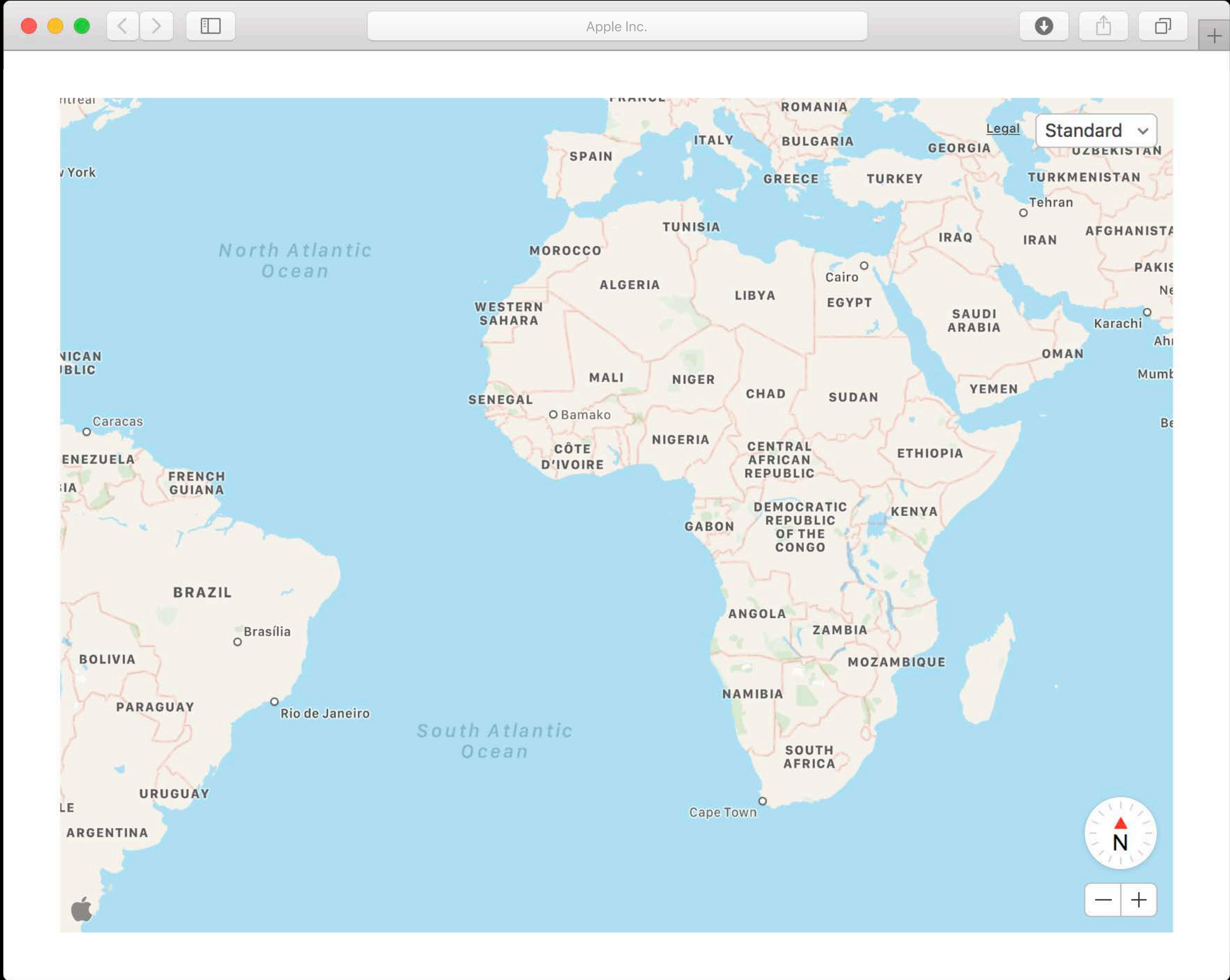
```
    authorizationCallback: done => fetch(...).then(...).then(done);
```

```
  });
```

```
  let map = new mapkit.Map("map");
```

```
</script>
```

# Setting Up Your Map



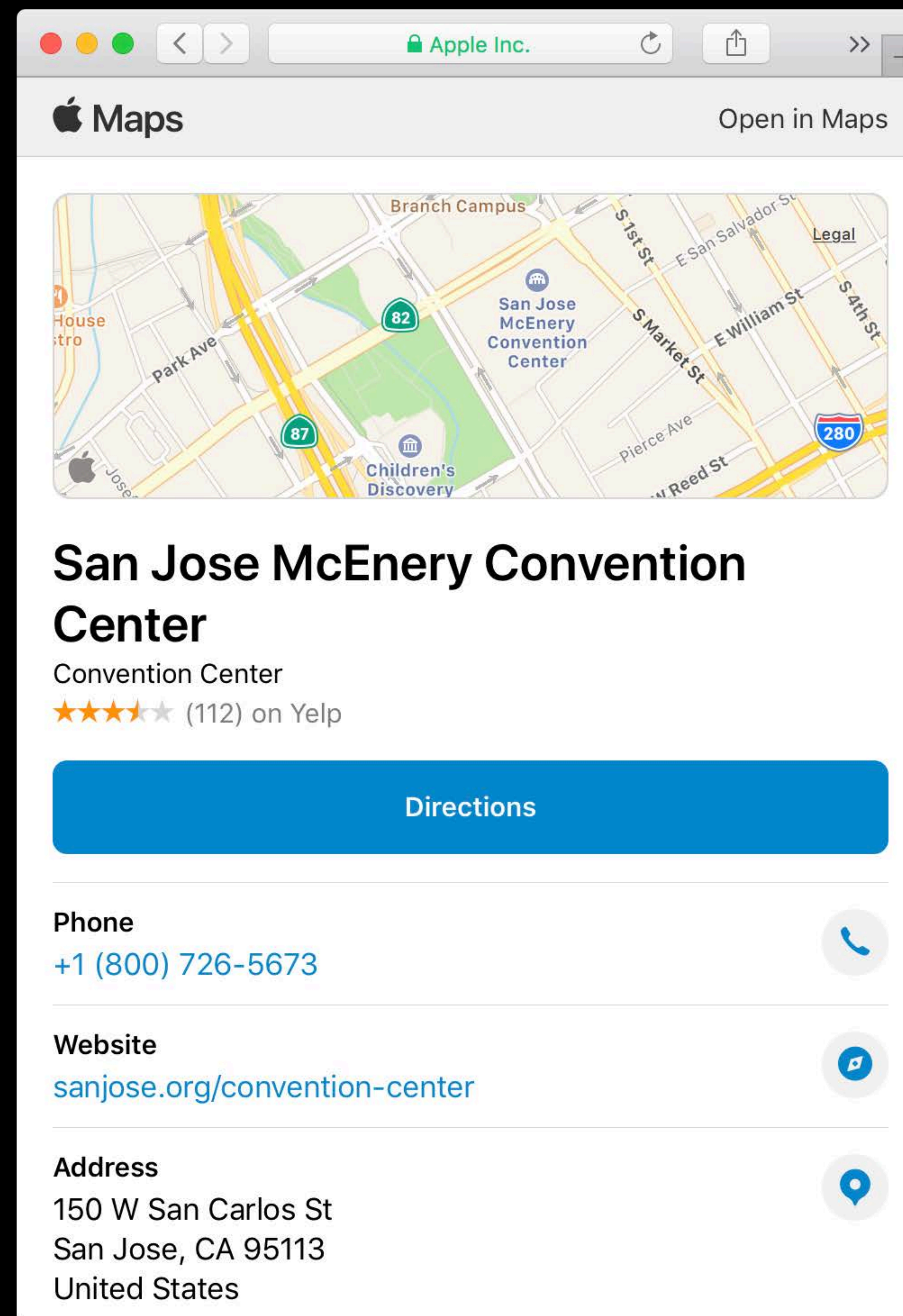


# Platform and Map Size-Specific Defaults

Minimal controls on smaller maps and touch-enabled platforms

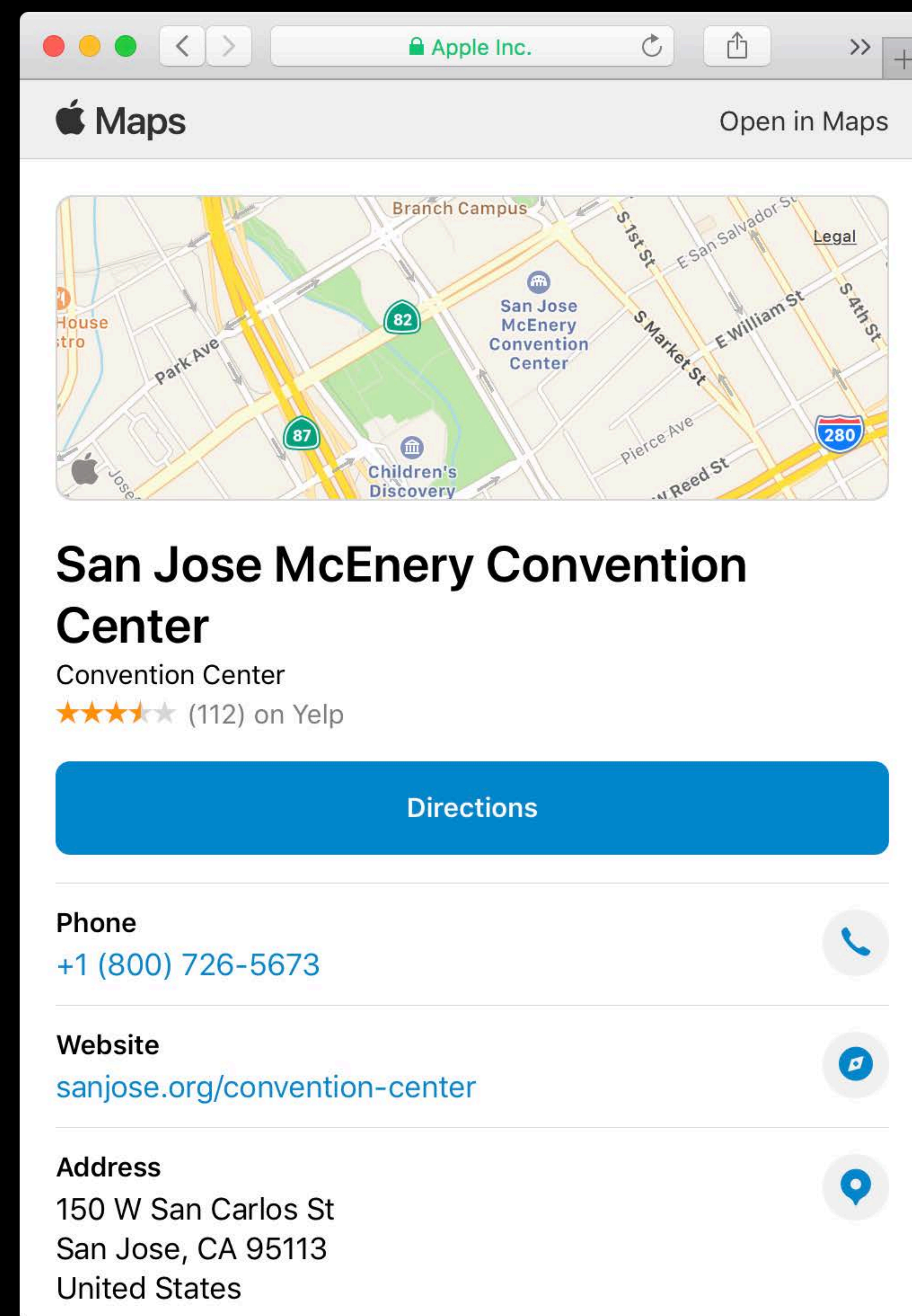
# Platform and Map Size-Specific Defaults

Minimal controls on smaller maps and touch-enabled platforms



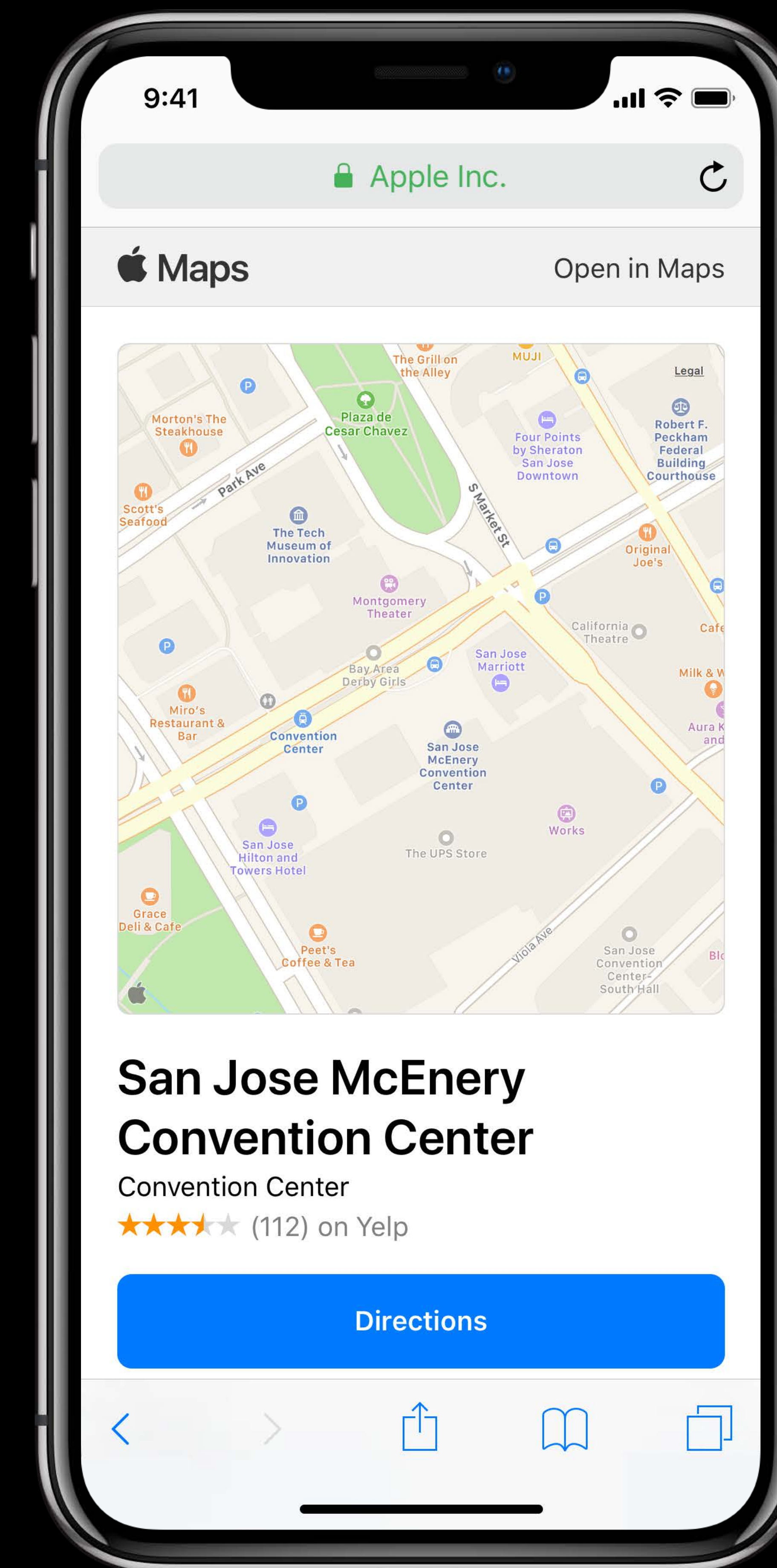
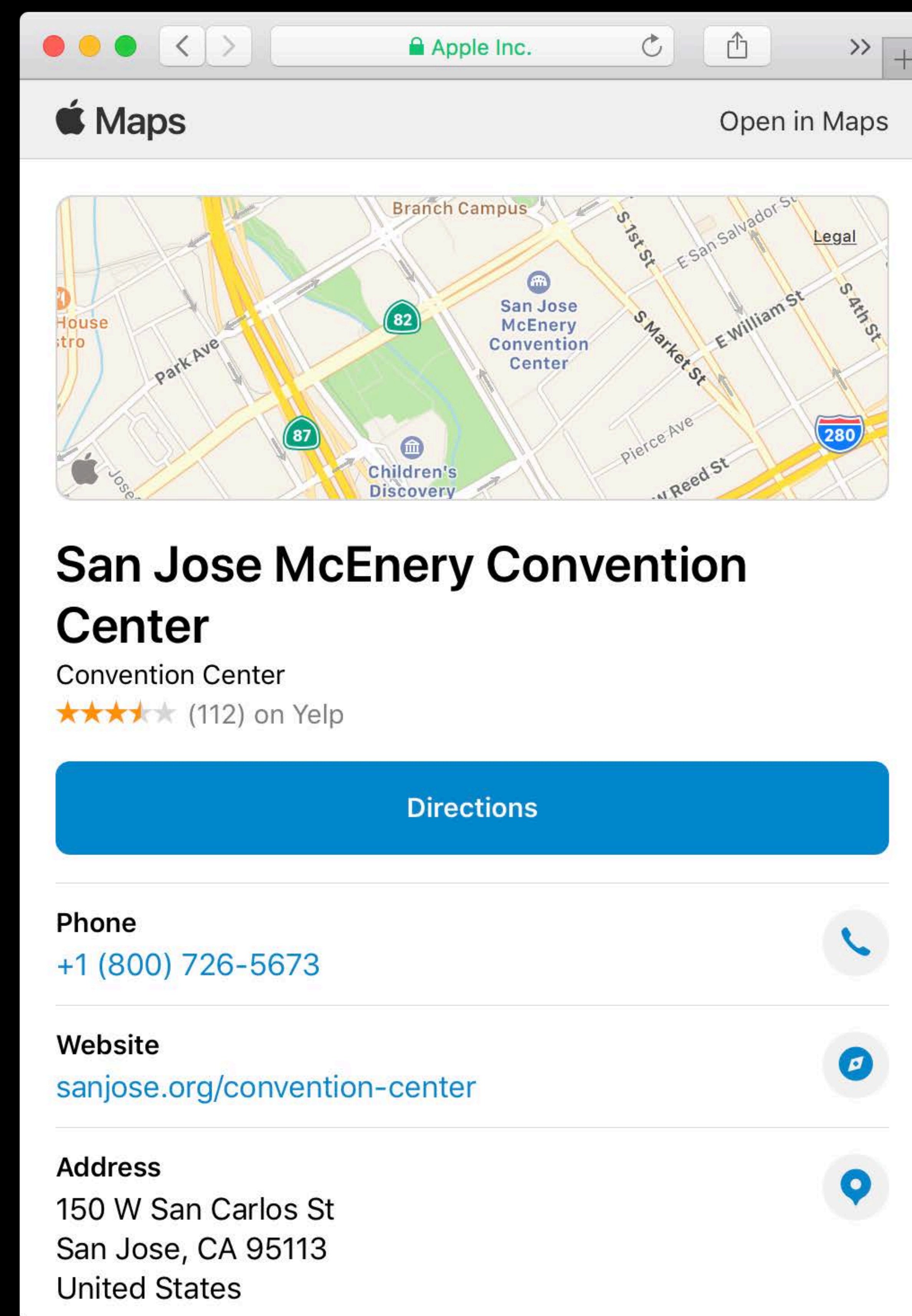
# Platform and Map Size-Specific Defaults

Minimal controls on smaller maps and touch-enabled platforms



# Platform and Map Size-Specific Defaults

Minimal controls on smaller maps and touch-enabled platforms

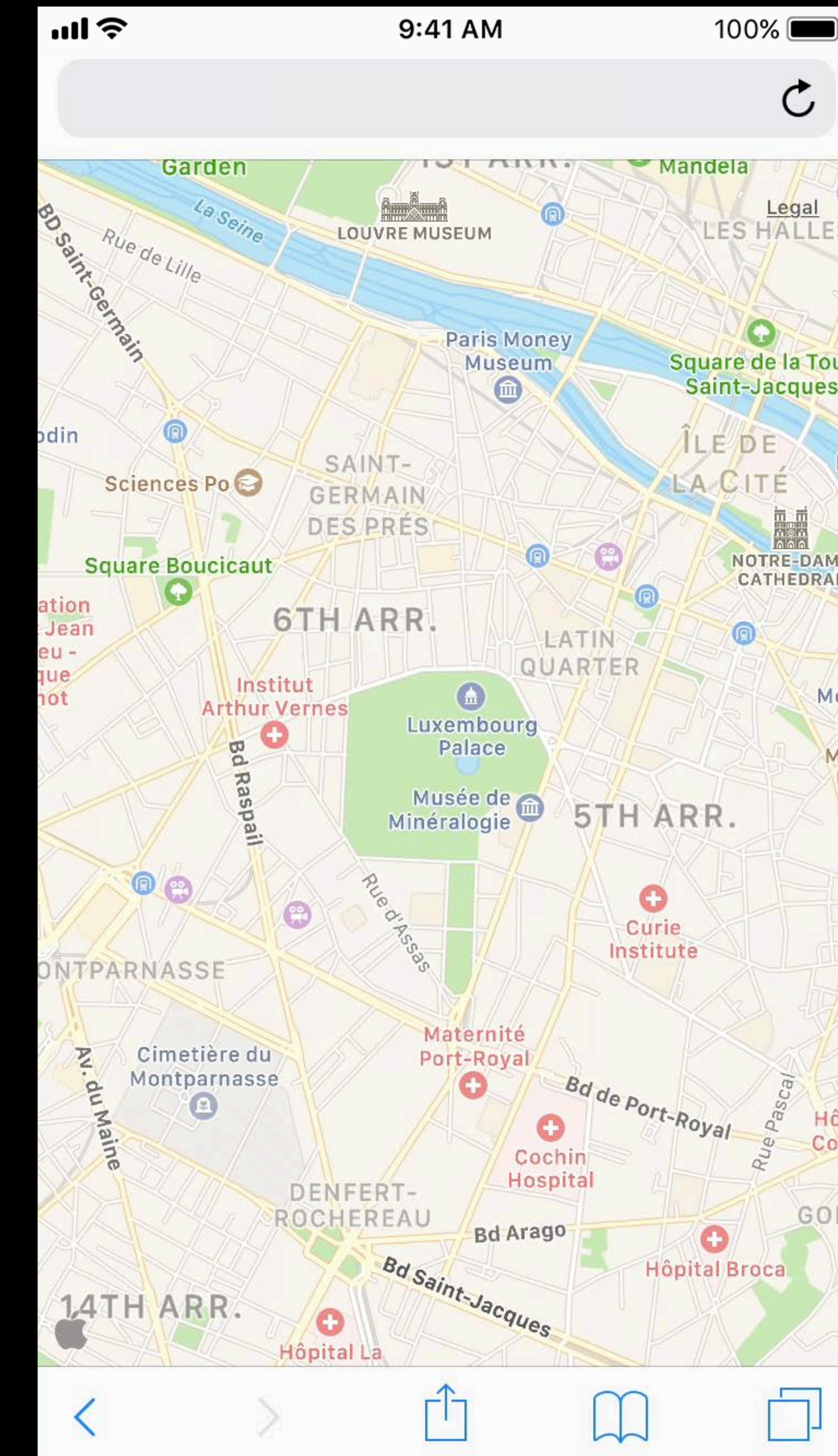


Apple logo and Legal text are always present

# Configurable Controls for Every Scenario

Adaptive controls

Compass and scale control are adaptive

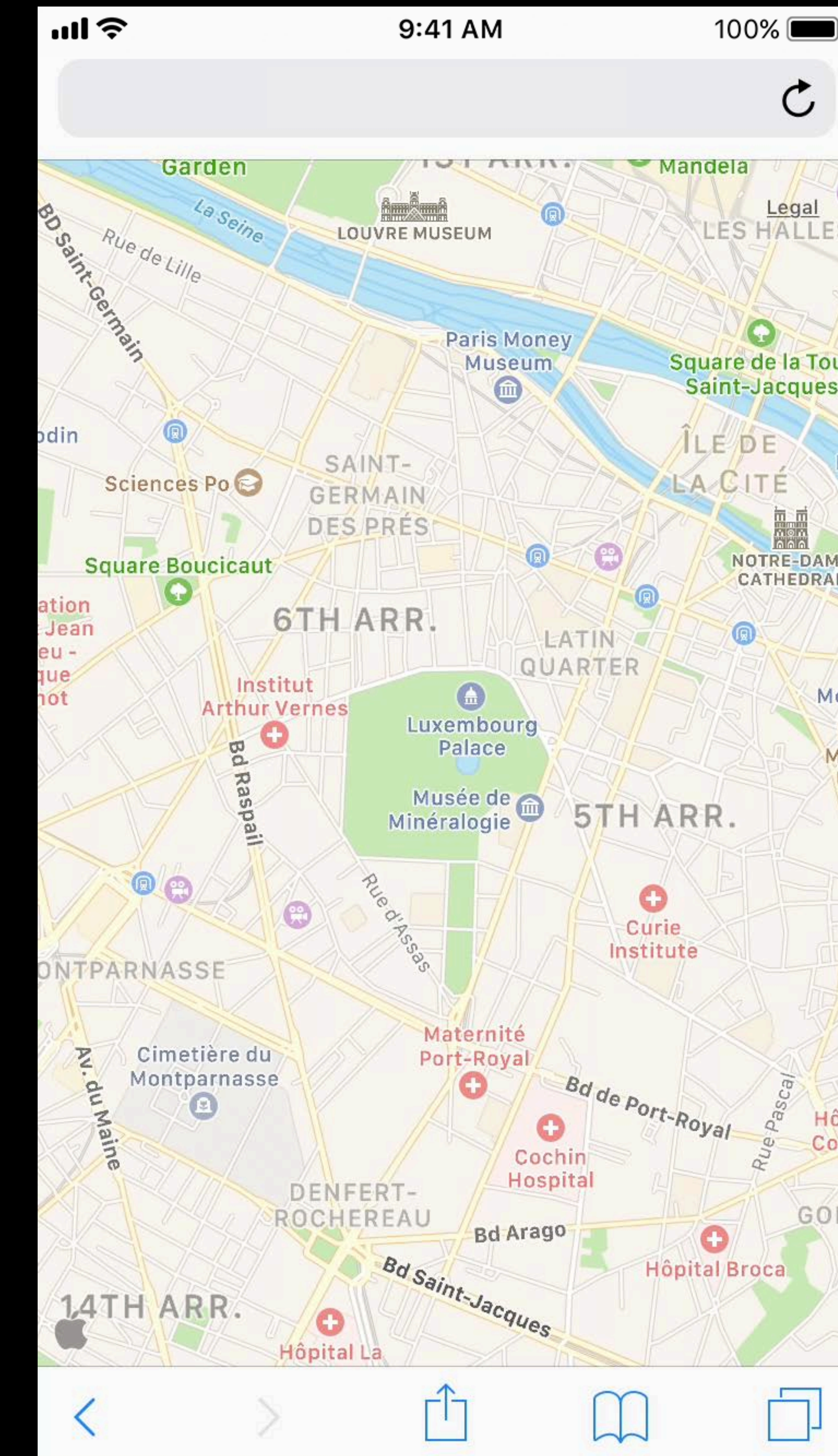


# Configurable Controls for Every Scenario

## Adaptive controls

Compass and scale control are adaptive

- On iOS, adaptive compass is shown when rotation is any non-zero value

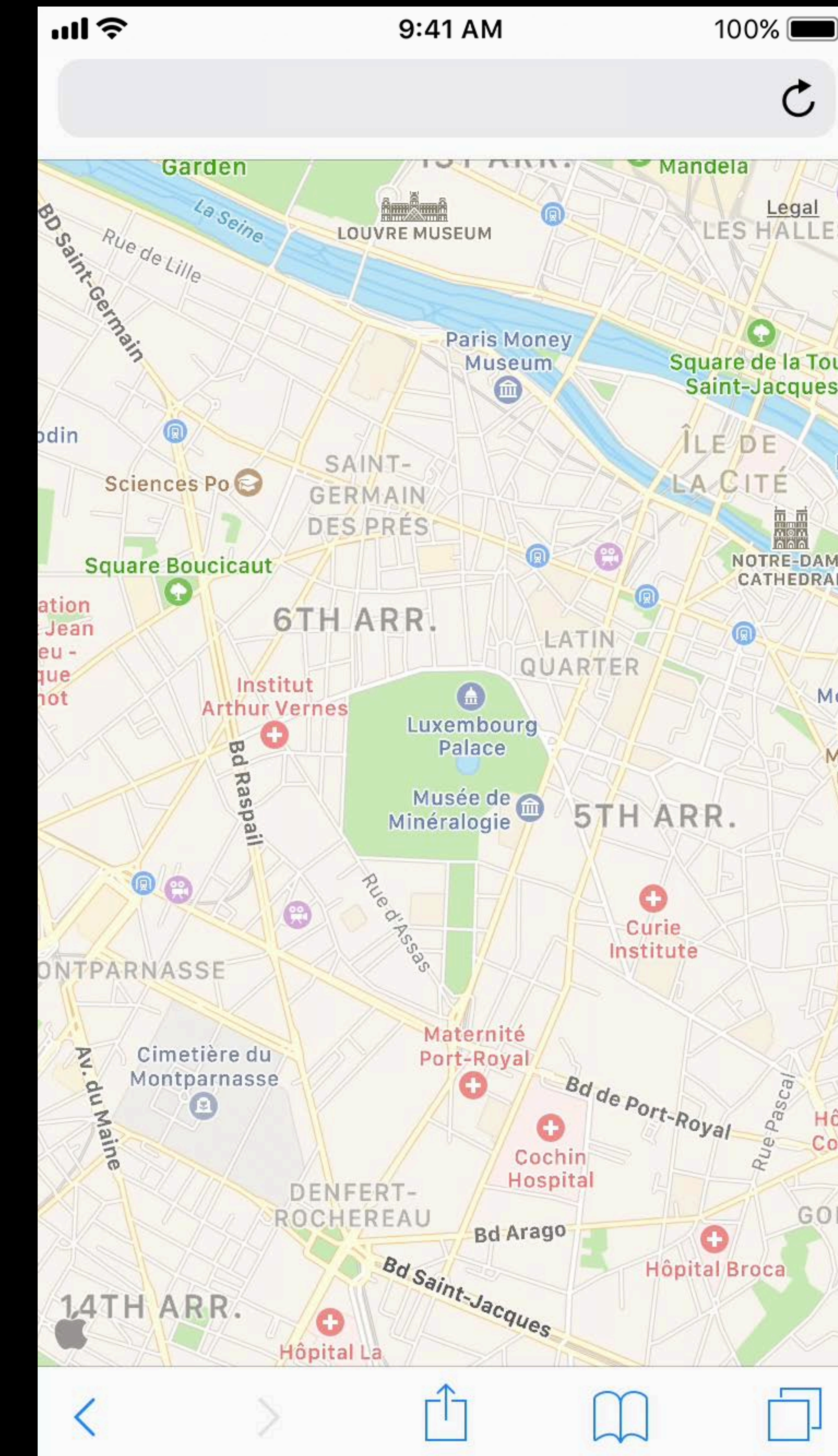


# Configurable Controls for Every Scenario

## Adaptive controls

Compass and scale control are adaptive

- On iOS, adaptive compass is shown when rotation is any non-zero value
- Adaptive scale is shown only while zooming

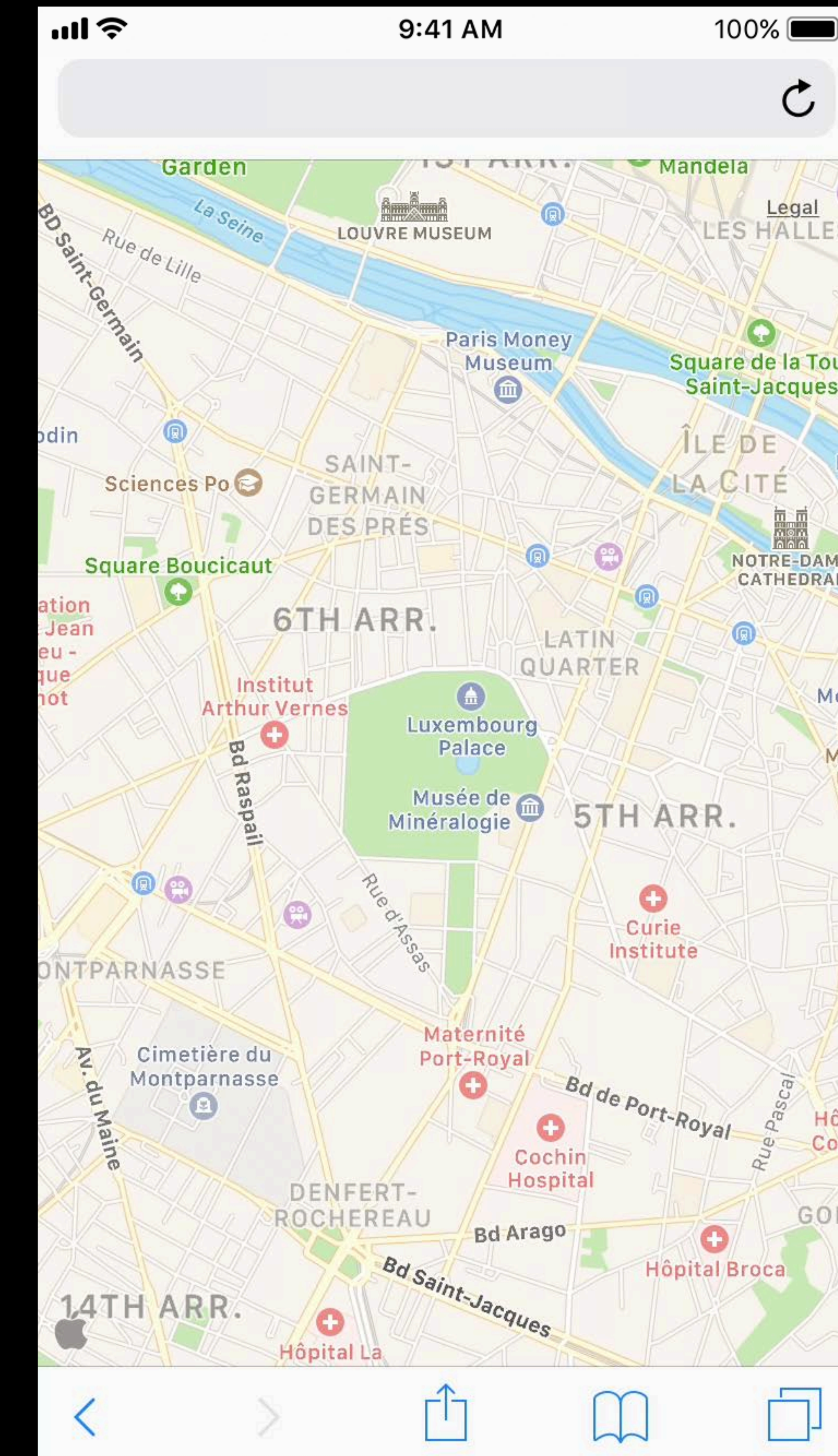


# Configurable Controls for Every Scenario

## Adaptive controls

Compass and scale control are adaptive

- On iOS, adaptive compass is shown when rotation is any non-zero value
- Adaptive scale is shown only while zooming





# Configurable Controls for Every Scenario

Hidden or visible

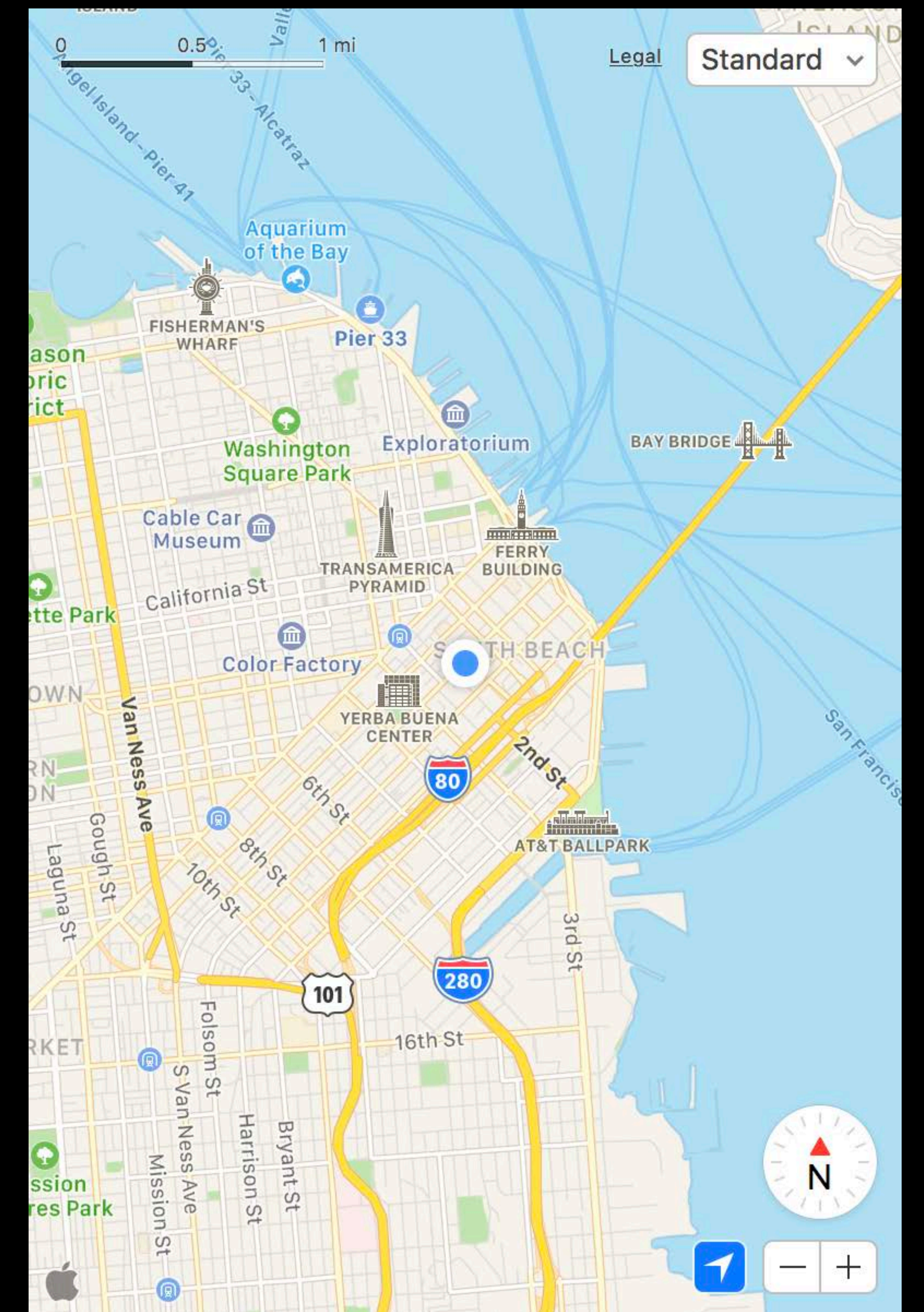
Remaining controls can be hidden or visible

- User location
- Zoom
- Map type

# Configurable Controls for Every Scenario

Hidden or visible

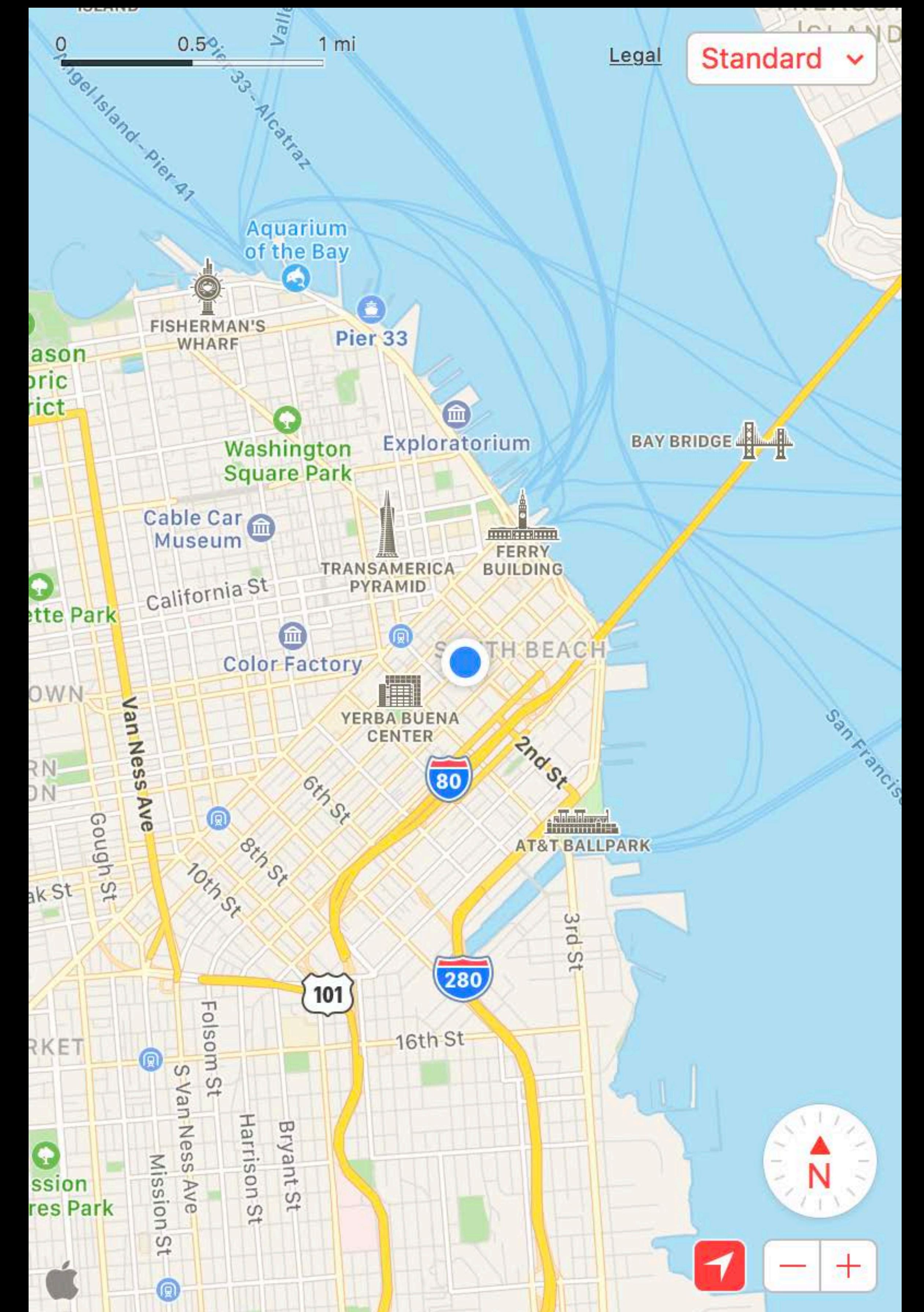
```
map.showsUserLocationControl = true;  
map.showsScale = mapkit.FeatureVisibility.Visible;
```



# Configurable Controls for Every Scenario

## Tinting

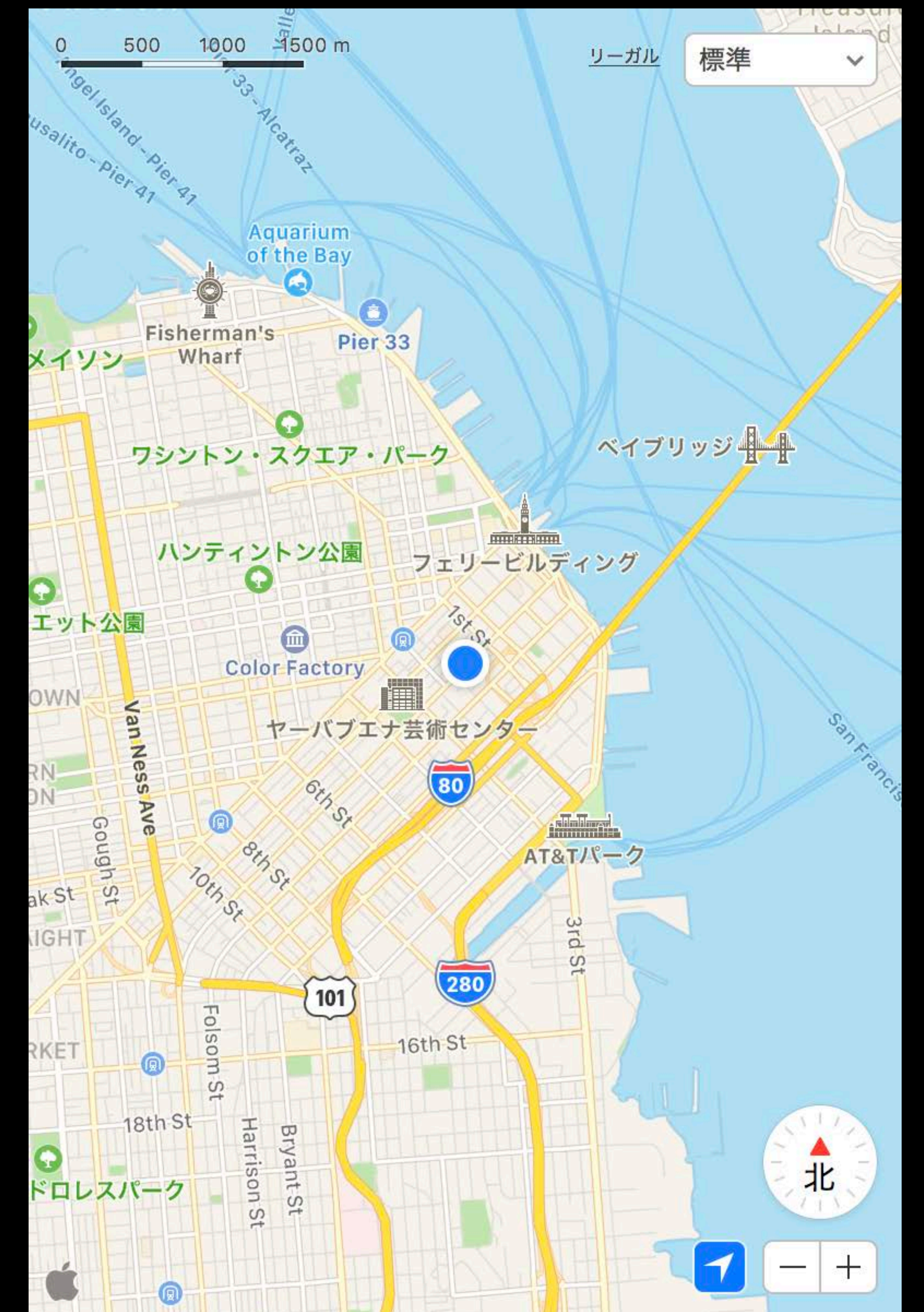
```
map.showsUserLocationControl = true;  
map.showsScale = mapkit.FeatureVisibility.Visible;  
map.tintColor = "#ff4040";
```



# Configurable Controls for Every Scenario

## Localization

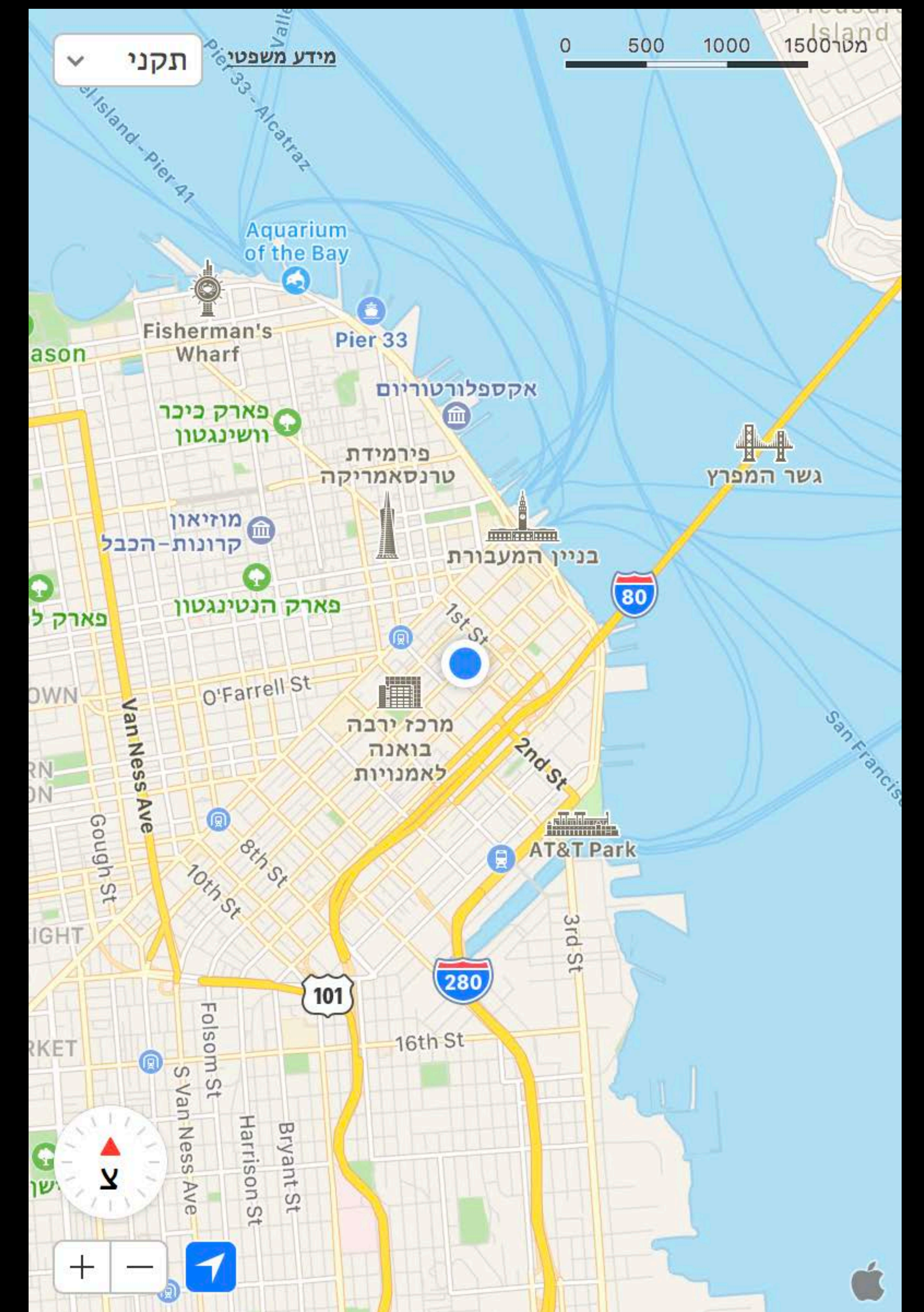
```
map.showsUserLocationControl = true;  
map.showsScale = mapkit.FeatureVisibility.Visible;  
mapkit.language = "ja-JA";
```



# Configurable Controls for Every Scenario

## Localization

```
map.showsUserLocationControl = true;  
map.showsScale = mapkit.FeatureVisibility.Visible;  
mapkit.language = "iw-IL";
```



# Configurable Controls for Every Scenario

## Disabling interactions

Panning (scrolling), zooming, and rotation via gestures can be disabled

To create a static, non-interactive map, set `is*Enabled` properties to `false`

```
map.isZoomEnabled = false;  
map.isScrollEnabled = false;  
map.isRotateEnabled = false;
```

# Using MapKit JS on Your Website

## Agenda

Setting up your map

Navigating and annotating the map

Enabling rich interactions with services

# Using MapKit JS on Your Website

## Agenda

Setting up your map

Navigating and annotating the map

Enabling rich interactions with services



# Navigating and Annotating the Map

Setting the center and region of the map

Marking locations with annotations

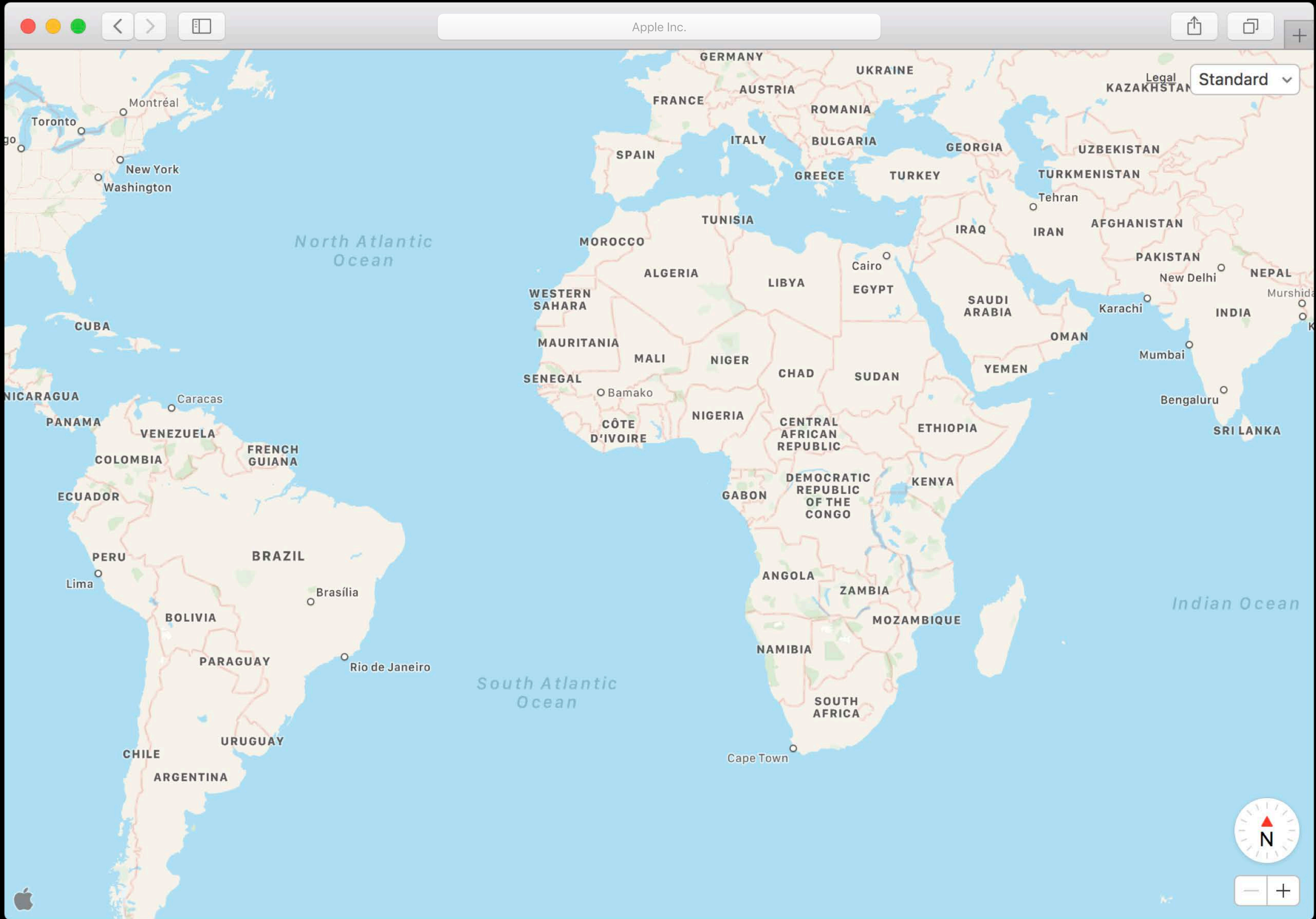
Covering areas with overlays

# Navigating and Annotating the Map

Setting the center and region of the map

Marking locations with annotations

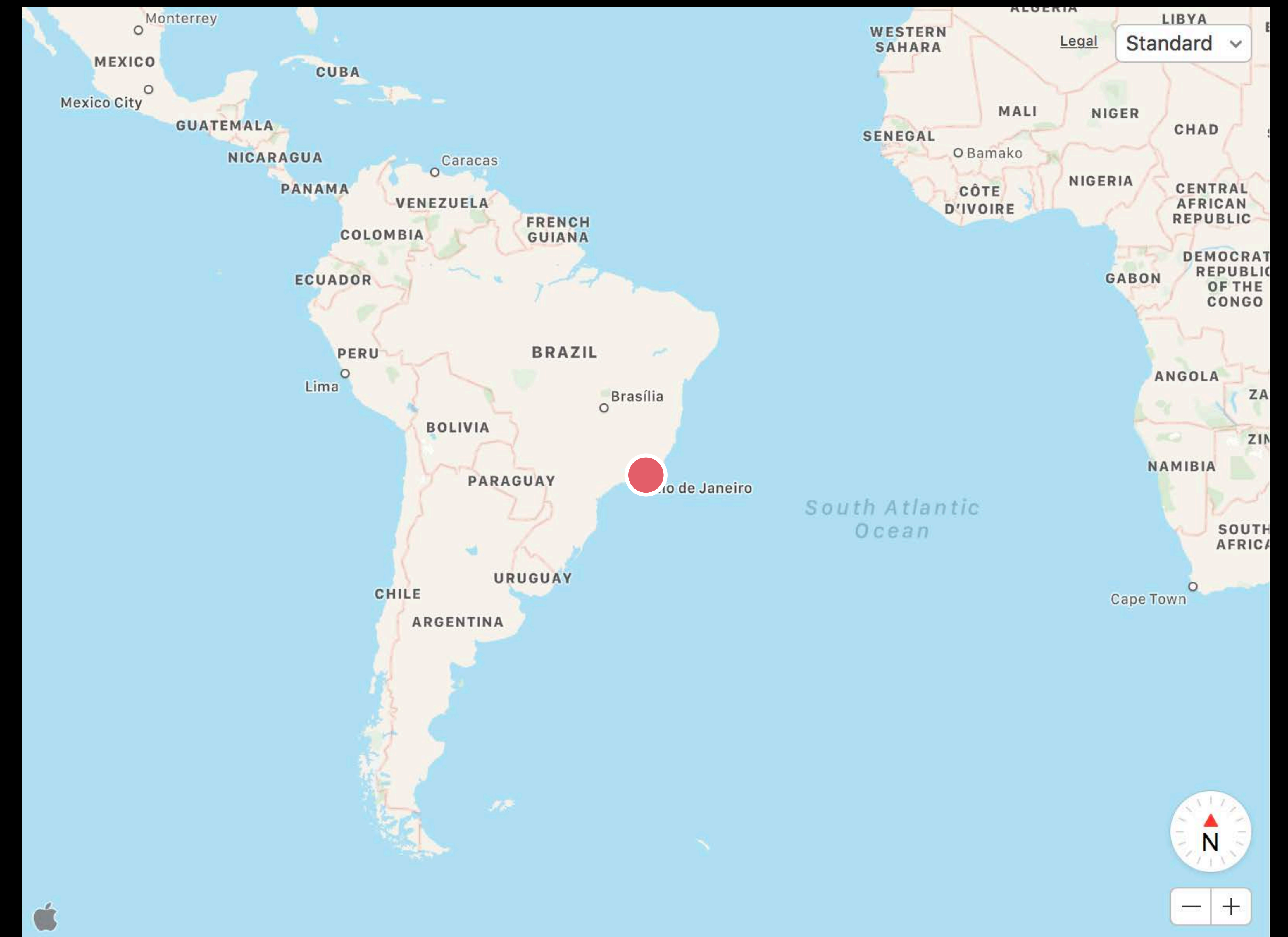
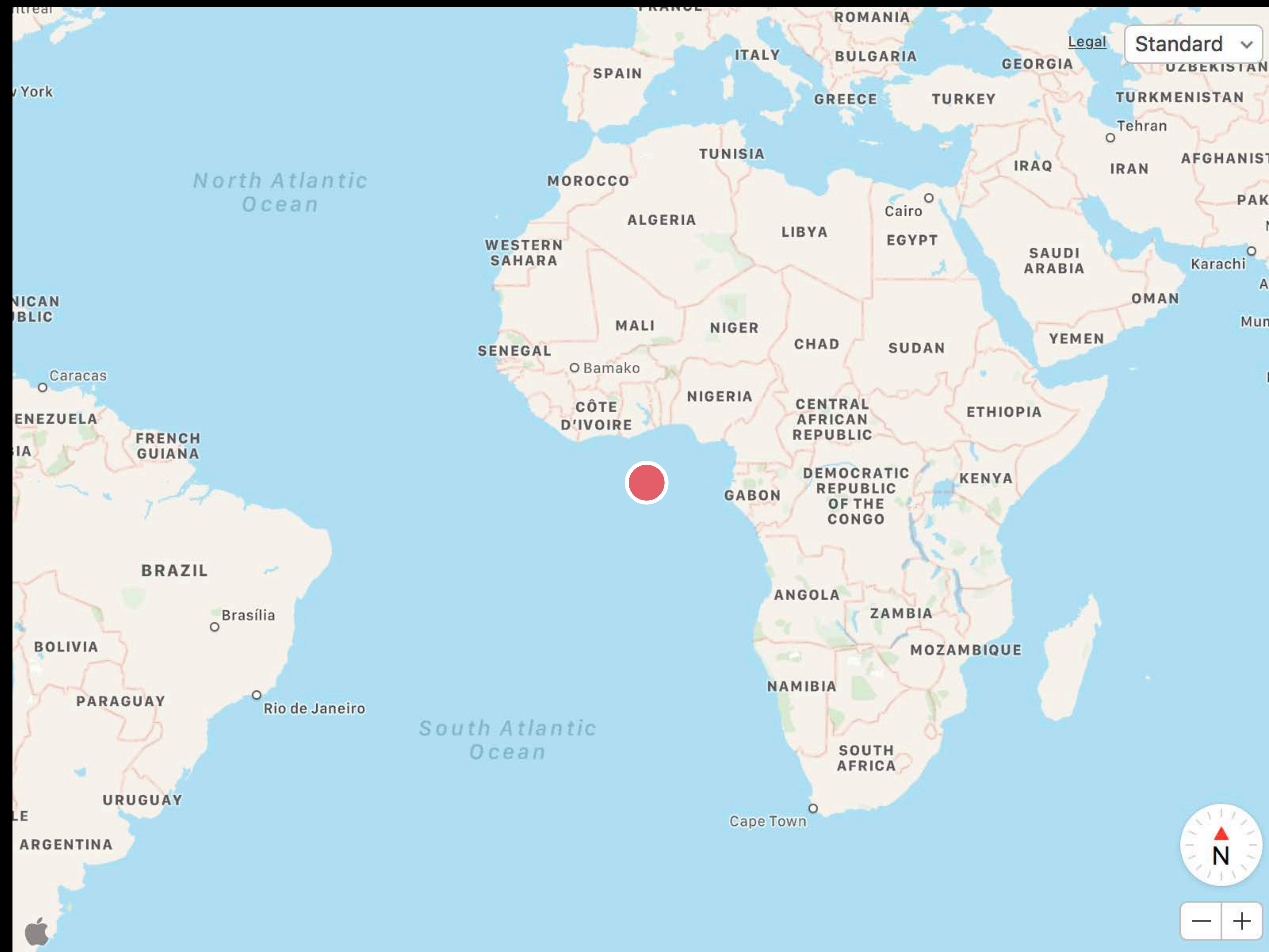
Covering areas with overlays



# Moving Around the Map

Setting the center of the map

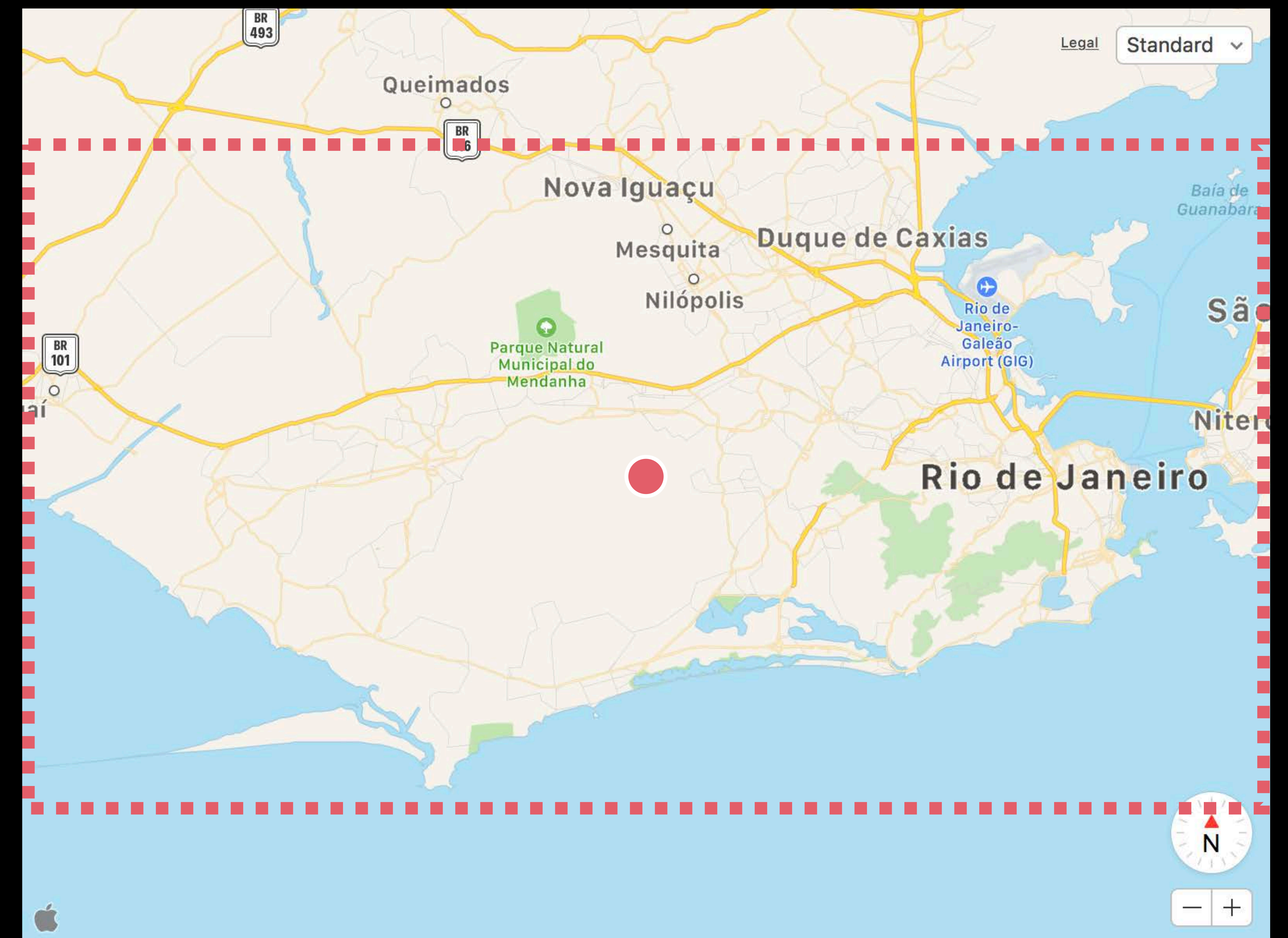
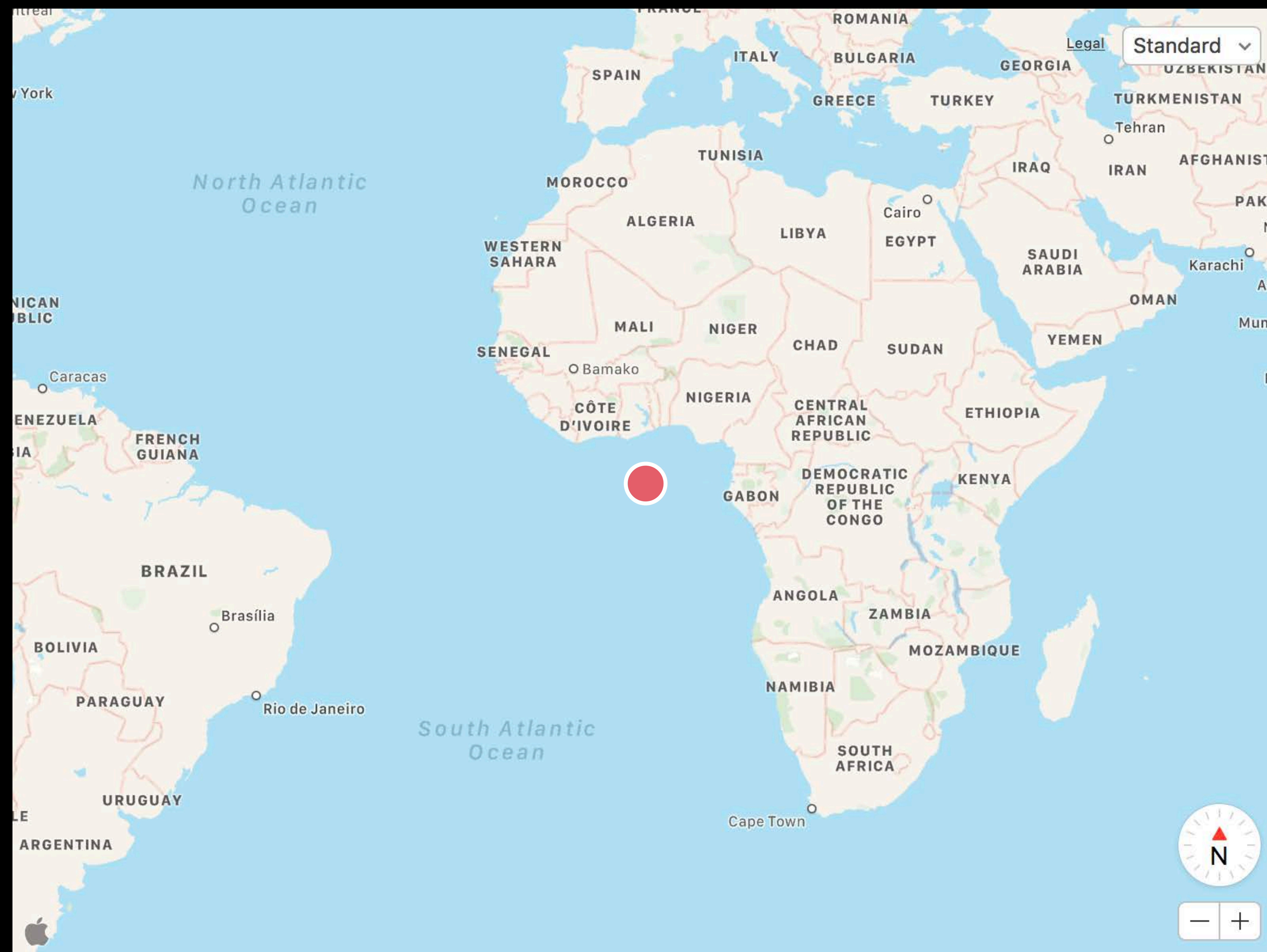
To pan, set the center point of the map to a coordinate



# Moving Around the Map

Setting the region of the map

To also change the scale, set the region of the map to a coordinate region



# Moving Around the Map

Setting center points and regions

The center is a `mapkit.Coordinate` (i.e., a latitude, longitude pair)

A `mapkit.CoordinateRegion` is a (center, coordinate span) pair

- Center is a `mapkit.Coordinate`
- `mapkit.CoordinateSpan` is a (latitude delta, longitude delta) pair

# Moving Around the Map

Always displaying the right region

Services return regions that enclose results

- Geocoding returns a `coordinate` and a `region`
- Search returns a `boundingRegion` that encloses its results

# Moving Around the Map

Always displaying the right region

Services return regions that enclose results

- Geocoding returns a `coordinate` and a `region`
- Search returns a `boundingRegion` that encloses its results

Use `map.showItems(items)` to set a region which encloses `items`

Region changes may be animated



# Responding to User Interactions

Map objects send event notifications for user interactions

- Region change begin and end
- Scroll, zoom, rotation begin and end

Following the model of DOM events

- `map.addListener("region-change-end", event => { ... })`
- `map.removeListener("zoom-start", event => { ... })`

# Navigating and Annotating The Map

Setting the center and region of the map

Marking locations with annotations

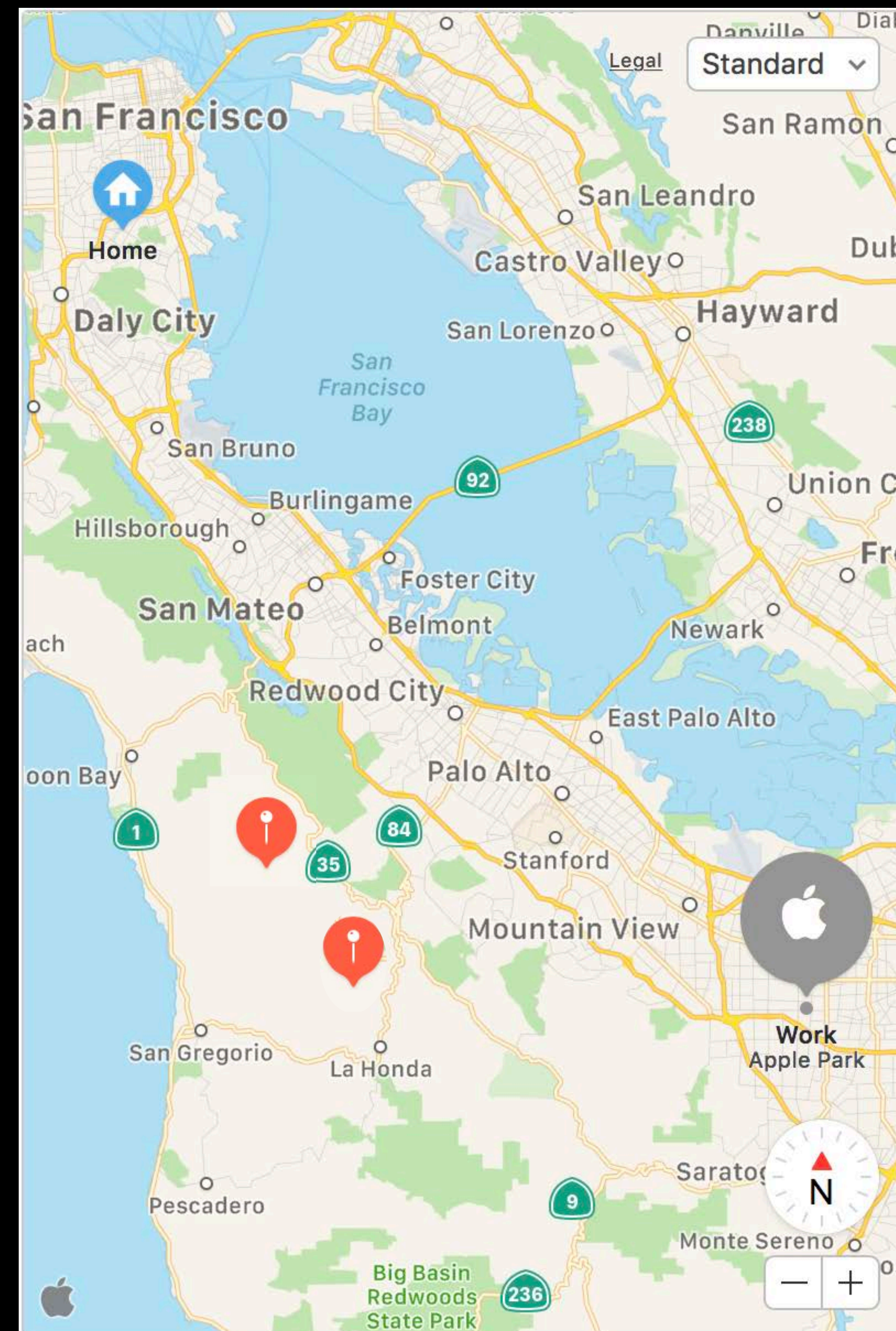
Covering areas with overlays

# Annotations

3 Types of annotations

# Annotations

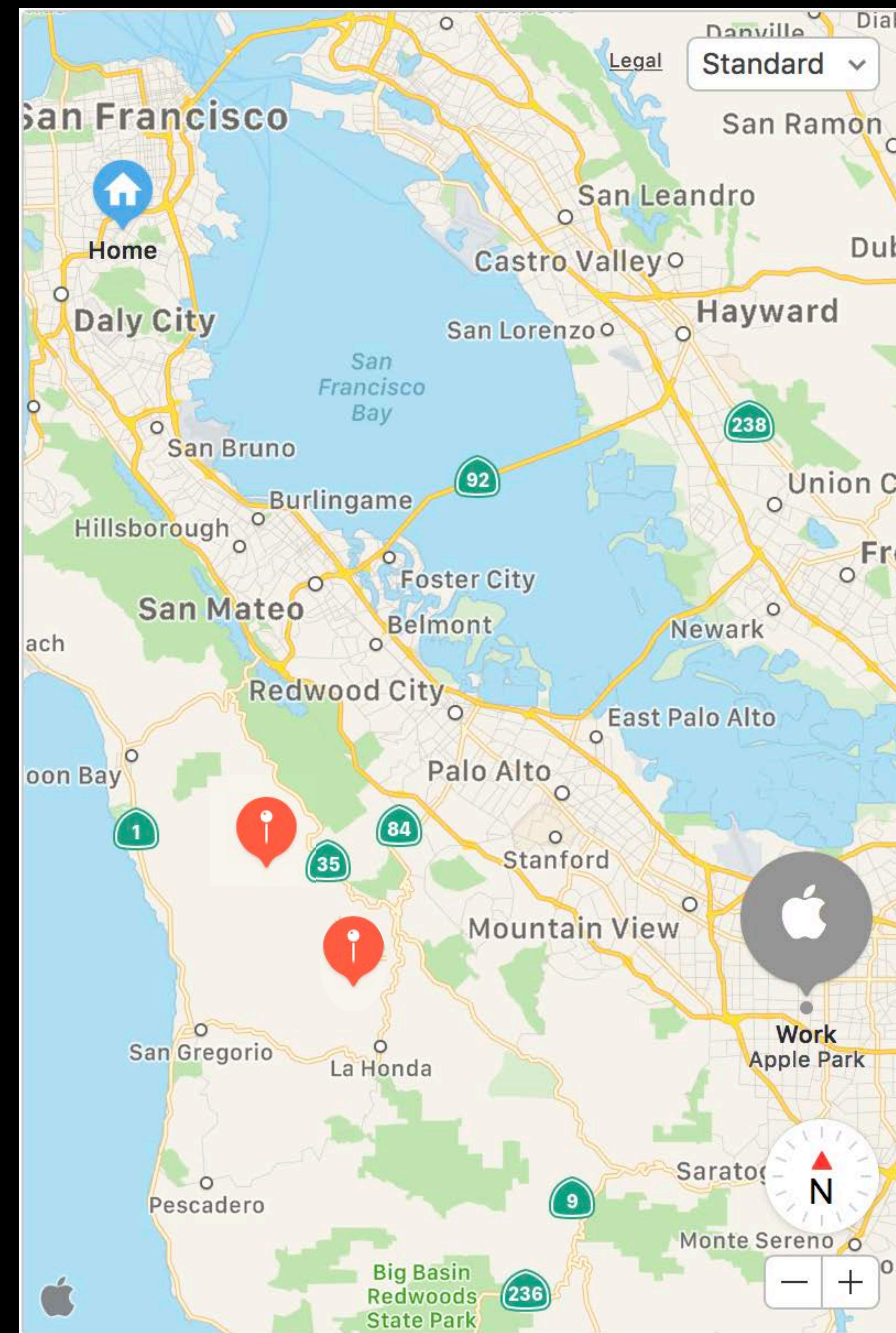
## 3 Types of annotations



Marker Annotation

# Annotations

## 3 Types of annotations



Marker Annotation

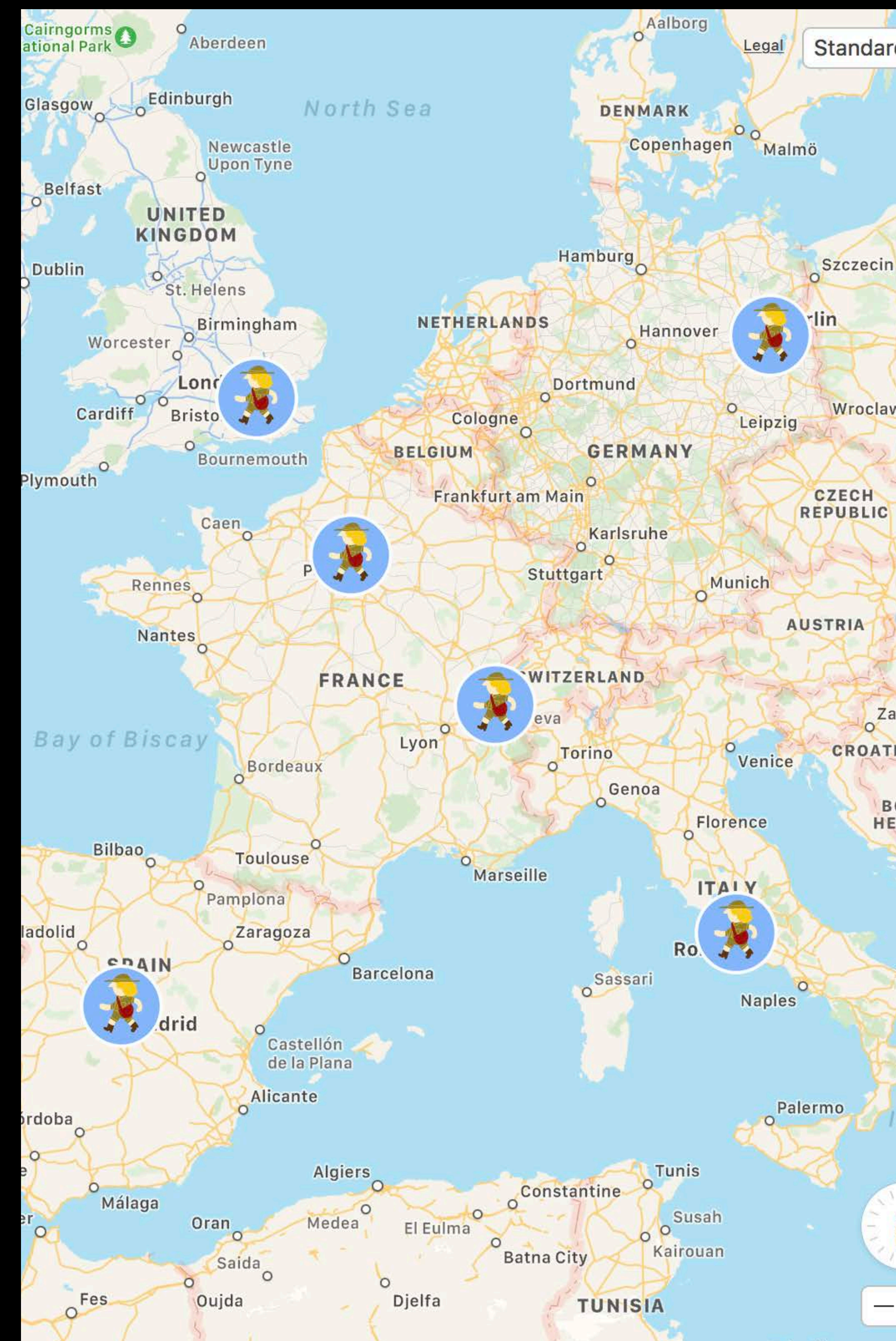
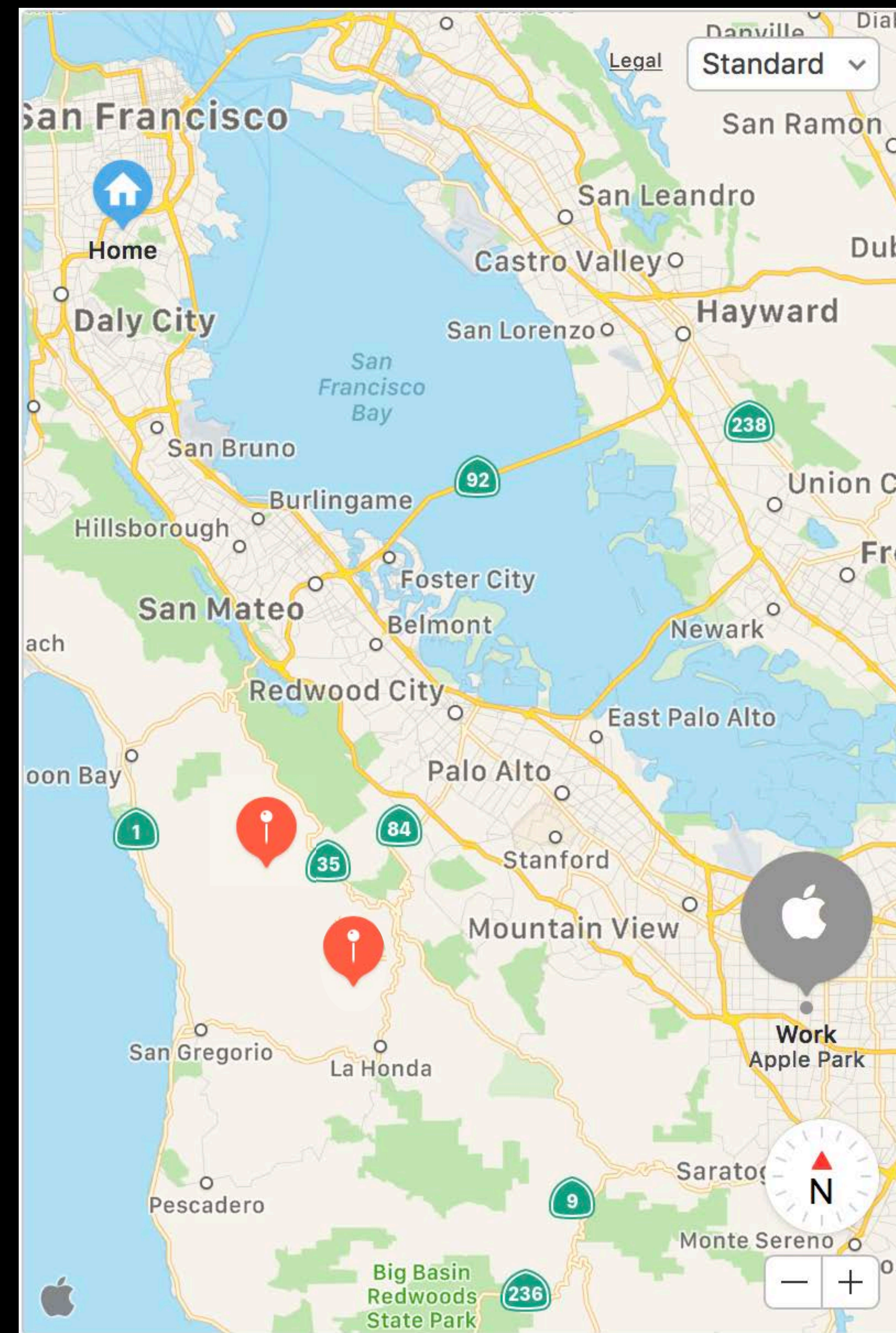


Image Annotation

# Annotations

## 3 Types of annotations



Marker Annotation

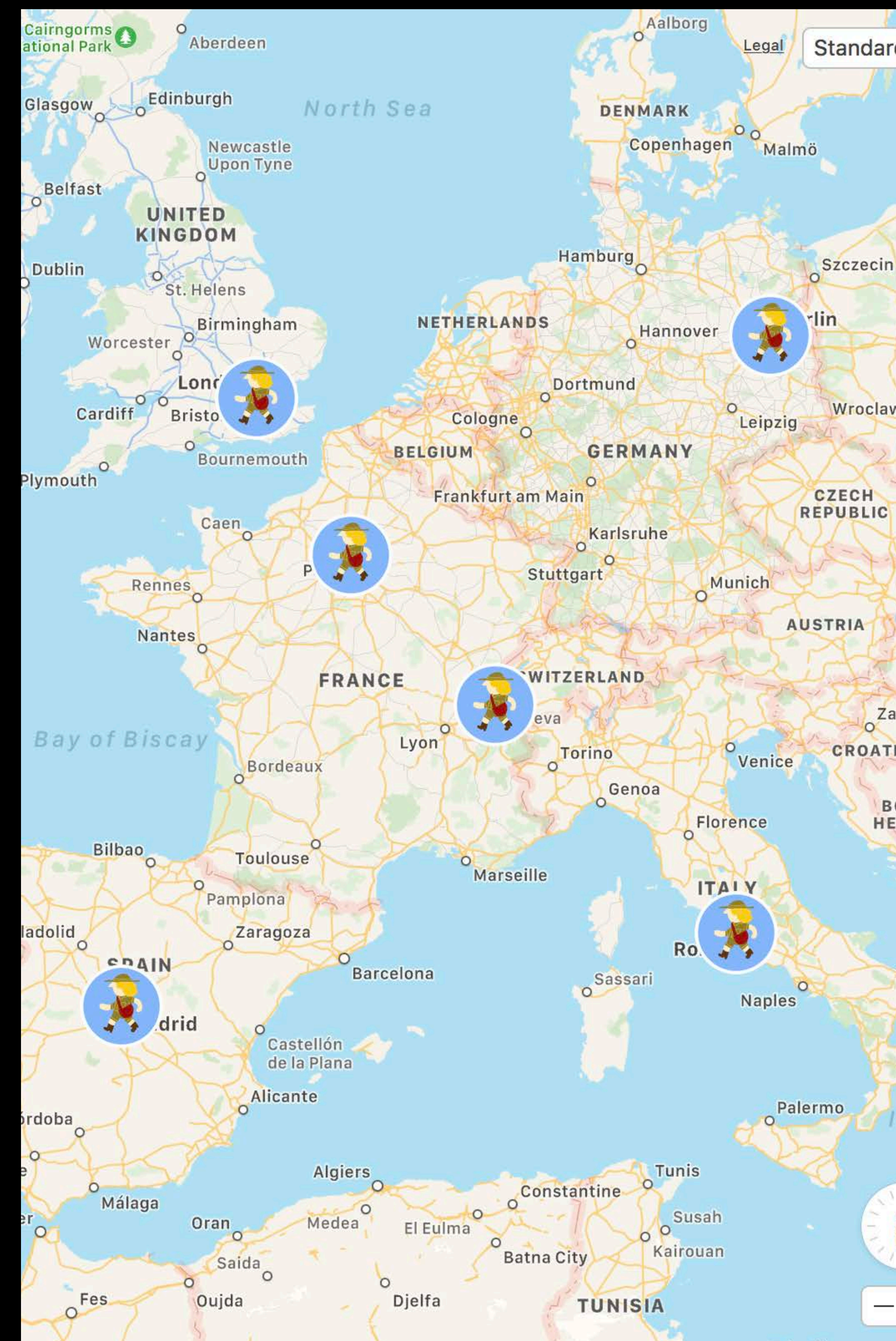
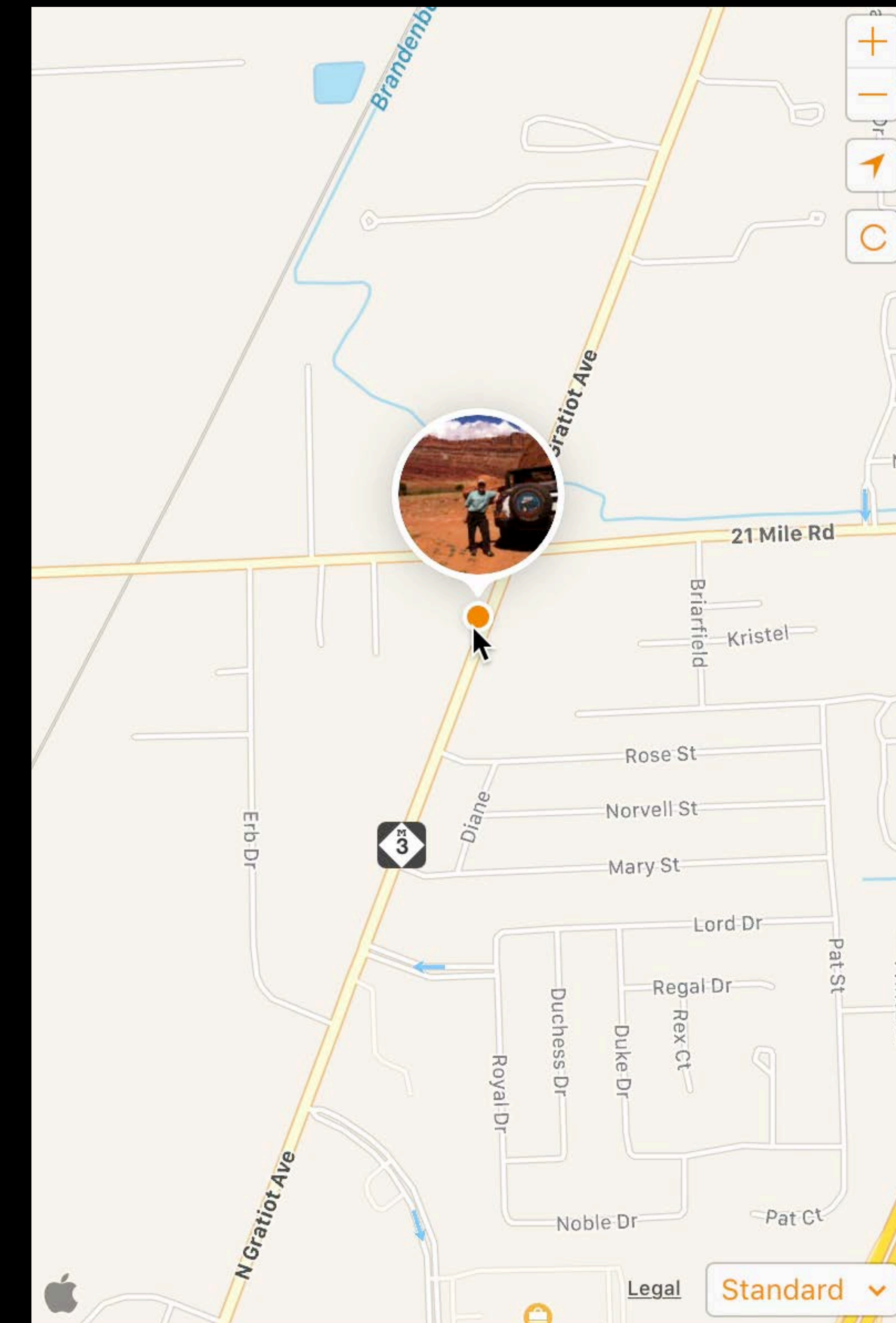


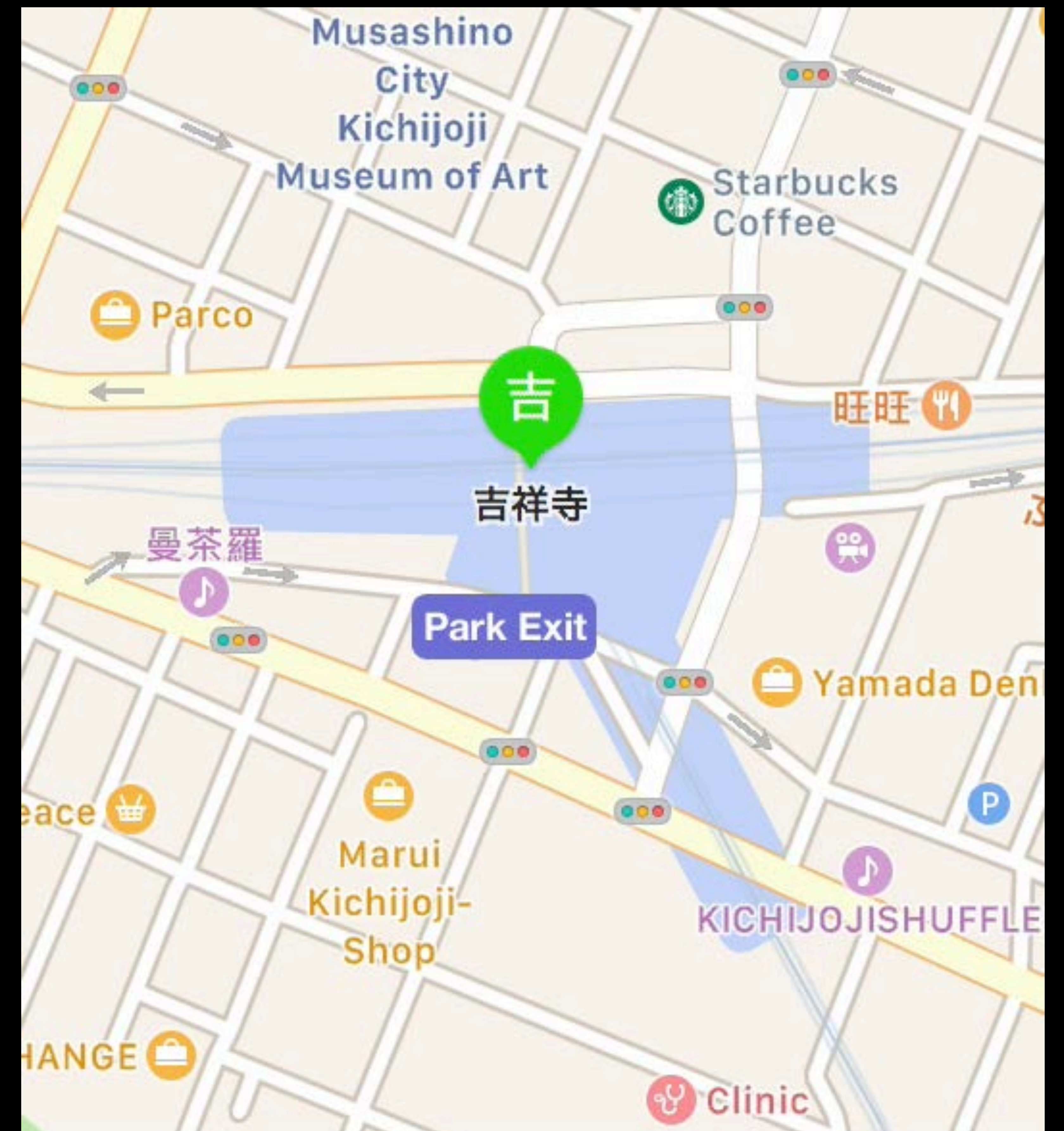
Image Annotation



Custom DOM Element Annotation

# Marker Annotations

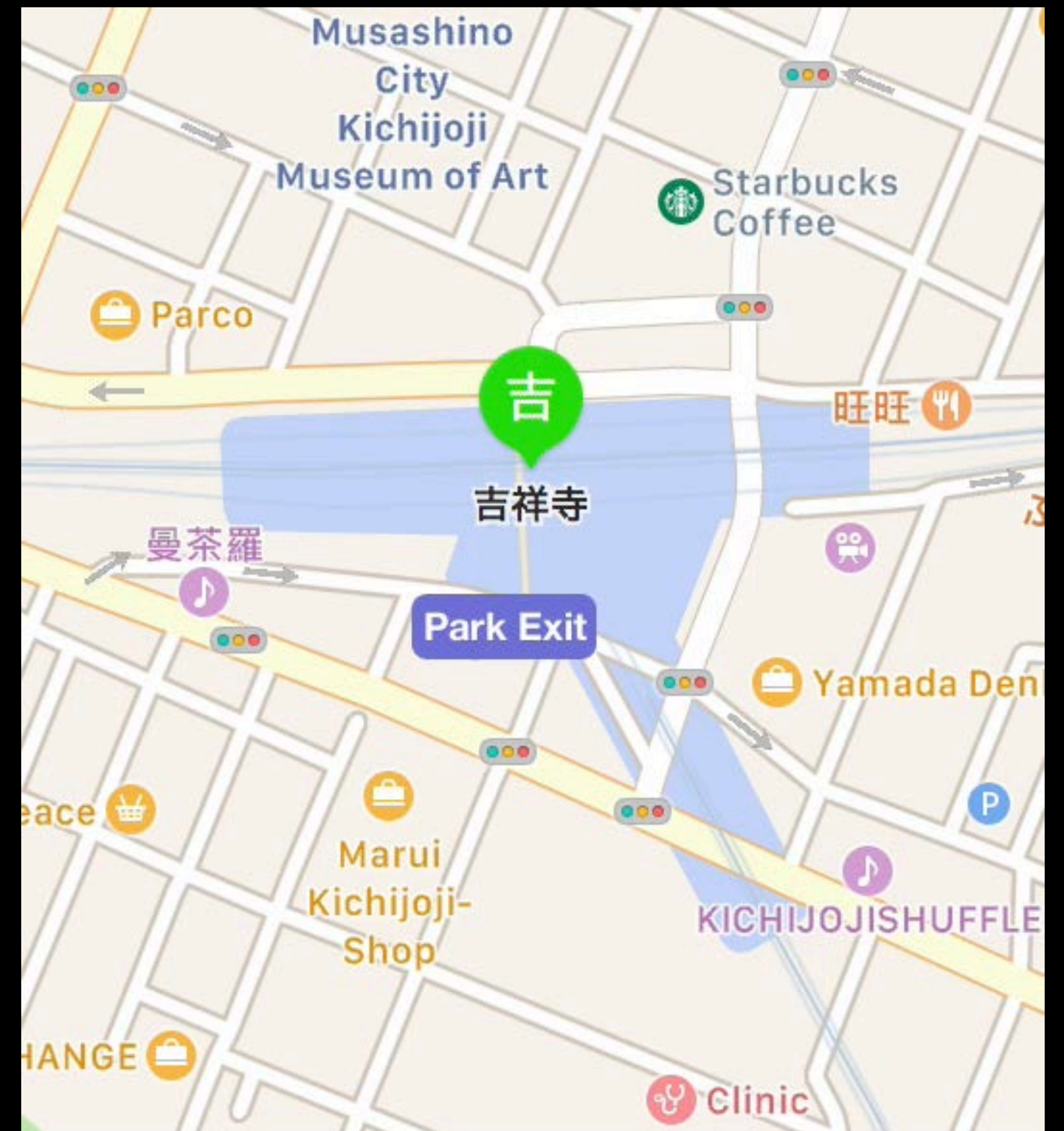
Advanced features, built-in



# Marker Annotations

Advanced features, built-in

Analogous to marker annotations on iOS





# Marker Annotations

Advanced features, built-in

Analogous to marker annotations on iOS

Built-in animation on selection and deselection



# Marker Annotations

Advanced features, built-in

Analogous to marker annotations on iOS

Built-in animation on selection and deselection

Automatically collide out underlying map labels



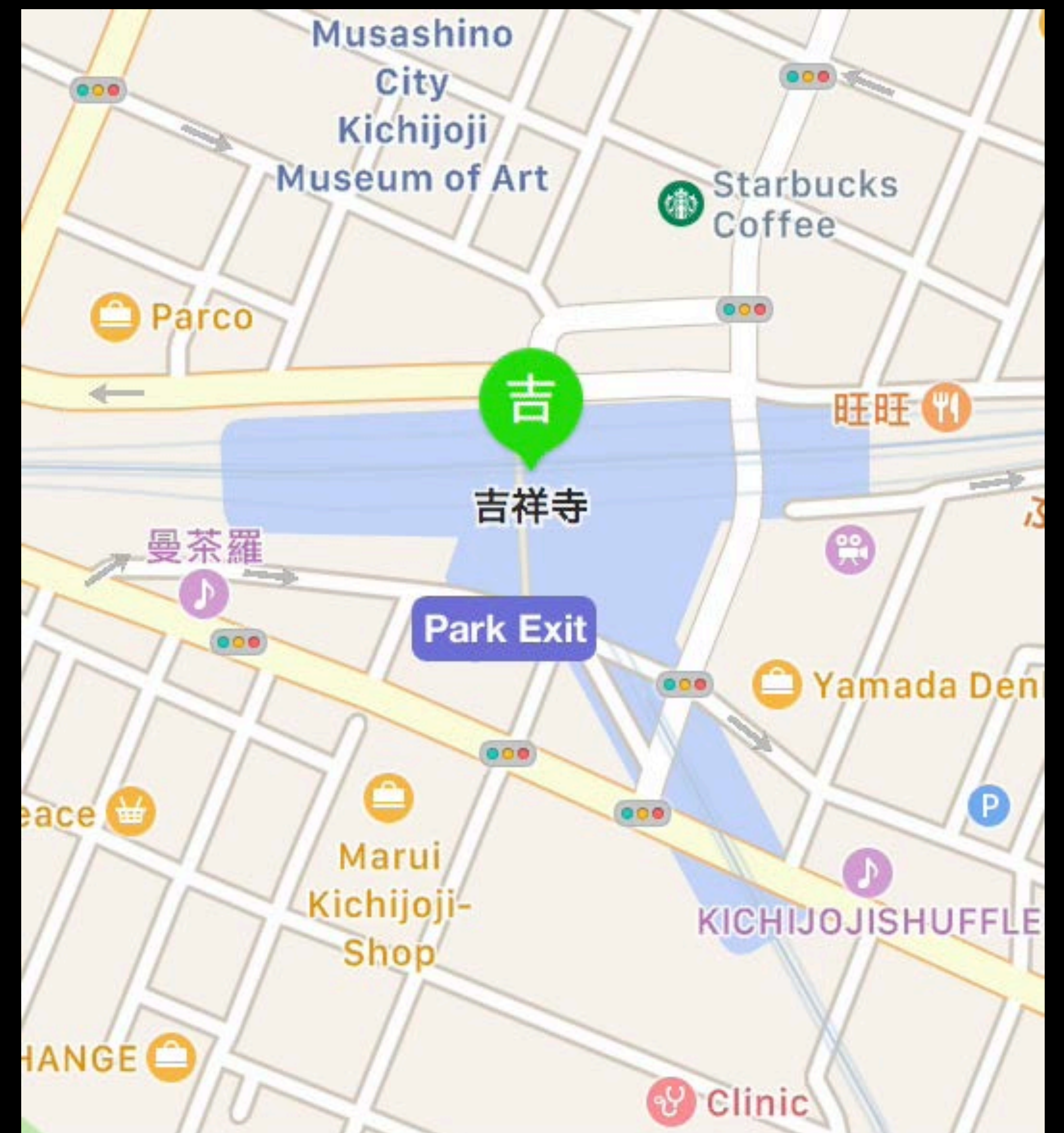
# Marker Annotations

Advanced features, built-in

Analogous to marker annotations on iOS

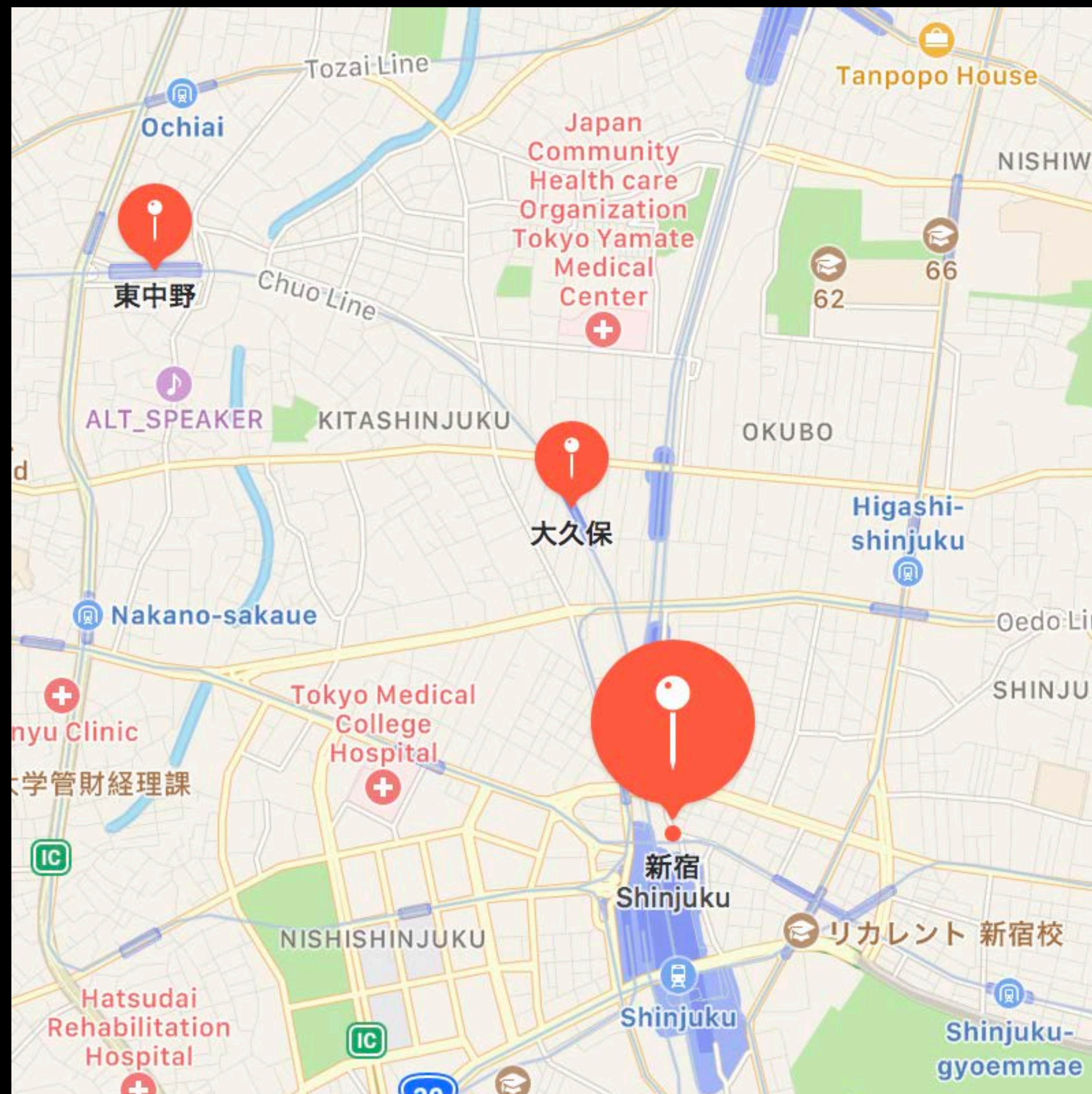
Built-in animation on selection and deselection

Automatically collide out underlying map labels

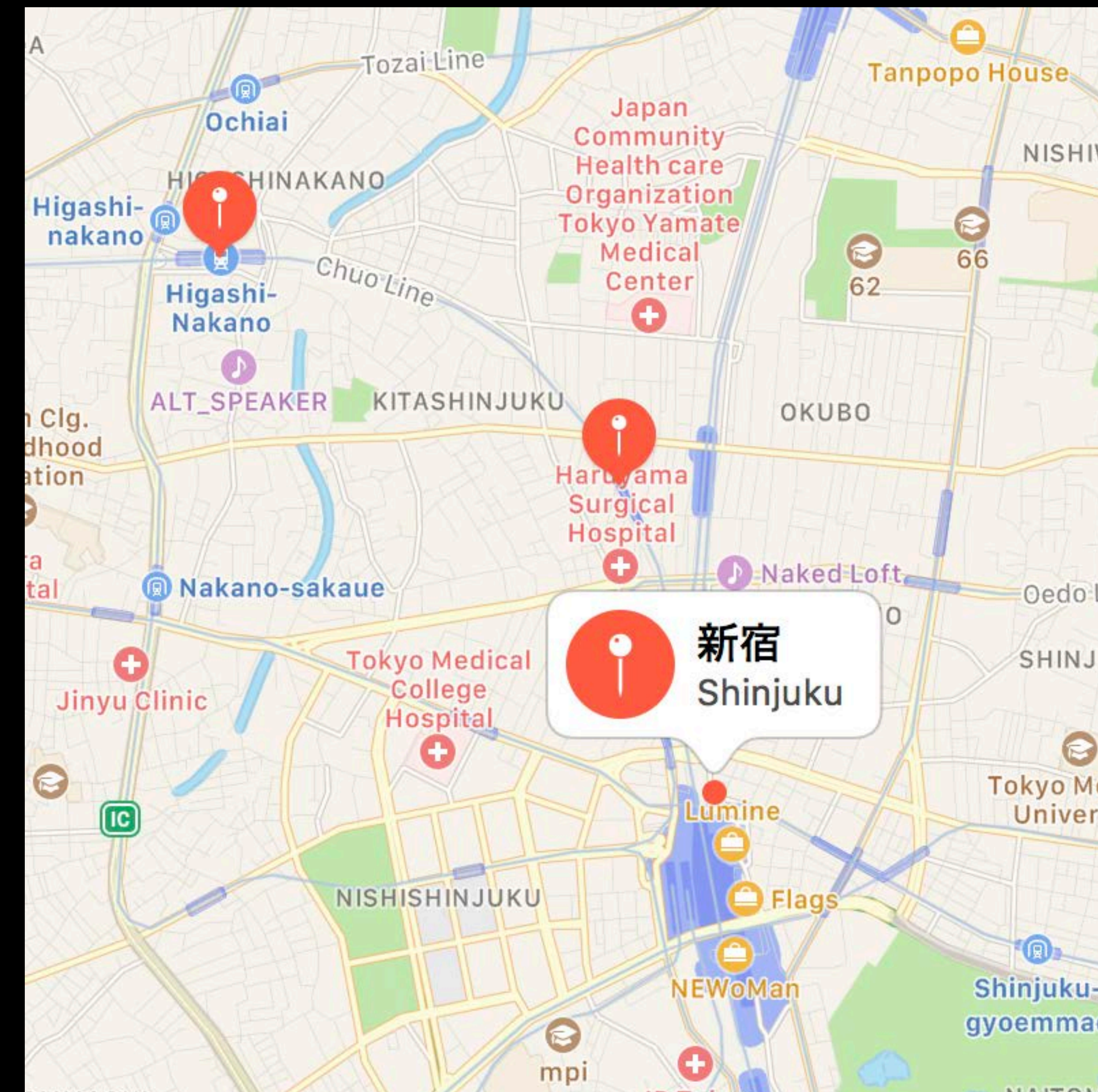


# Marker Annotations

Appearance automatically adapts for the current rendering mode



**Client-Side Rendering Mode**  
Title + subtitle display below balloon



**Sever-Side Rendering Mode**  
Title + subtitle display in callout

# Marker Annotations

Customize to match your site's look and feel

```
new mapkit.MarkerAnnotation(coordinate, {  
  title: "Fort de France",  
  subtitle: "Martinique"  
});
```



# Marker Annotations

Customize to match your site's look and feel

```
new mapkit.MarkerAnnotation(coordinate, {  
  title: "Fort de France",  
  subtitle: "Martinique",  
  color: "#53ac79"  
});
```



# Marker Annotations

Customize to match your site's look and feel

```
new mapkit.MarkerAnnotation(coordinate, {  
  title: "Fort de France",  
  subtitle: "Martinique",  
  color: "#53ac79",  
  glyphColor: "#fdc35d"  
});
```



# Marker Annotations

Customize to match your site's look and feel

```
new mapkit.MarkerAnnotation(coordinate, {  
  title: "Fort de France",  
  subtitle: "Martinique",  
  color: "#53ac79",  
  glyphColor: "#fdc35d",  
  glyphImage: {  
    2: "palm_tree_2x.png"  
  }  
});
```





# Marker Annotations

Customize to match your site's look and feel

```
new mapkit.MarkerAnnotation(coordinate, {
  title: "Fort de France",
  subtitle: "Martinique",
  color: "#53ac79",
  glyphColor: "#fdc35d",
  glyphImage: {
    2: "palm_tree_2x.png"
  },
  selectedGlyphImage: {
    2: "palm_tree_selected_2x.png"
  }
});
```



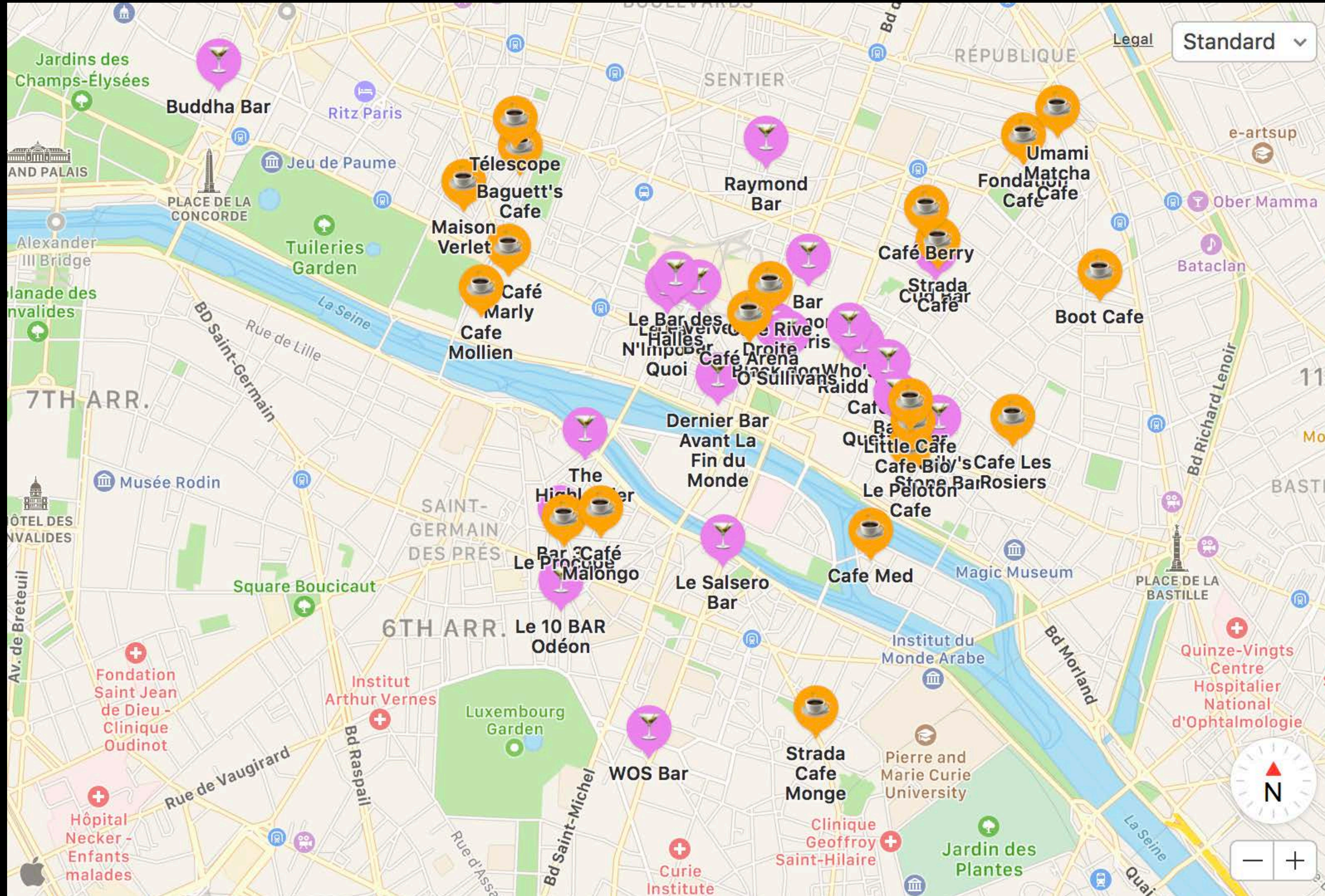
# Marker Annotations

Customize to match your site's look and feel

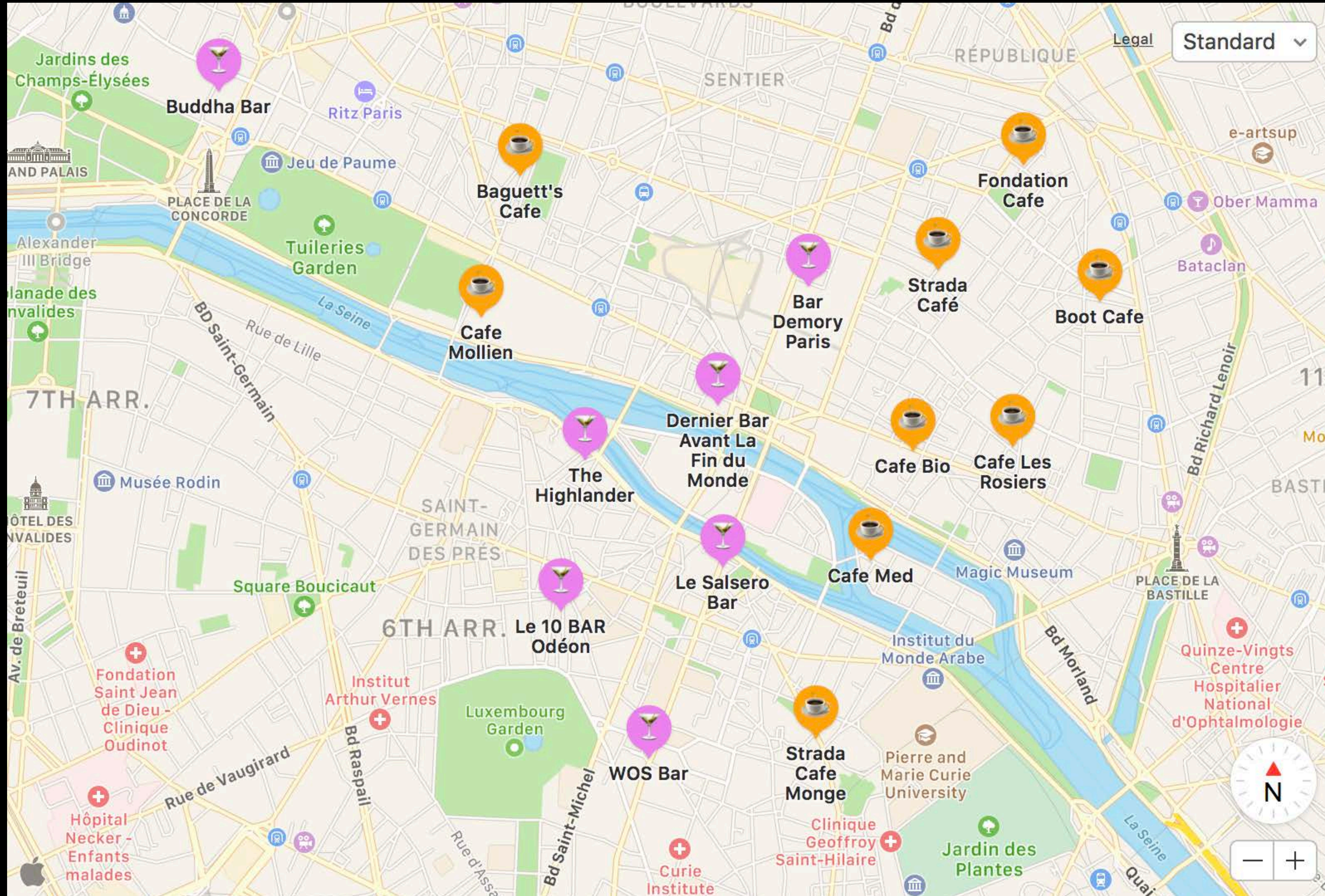
```
new mapkit.MarkerAnnotation(coordinate, {
  title: "Fort de France",
  subtitle: "Martinique",
  color: "#53ac79",
  glyphColor: "#fdc35d",
  glyphImage: {
    2: "palm_tree_2x.png"
  },
  selectedGlyphImage: {
    2: "palm_tree_selected_2x.png"
  },
  glyphText: "M"
});
```



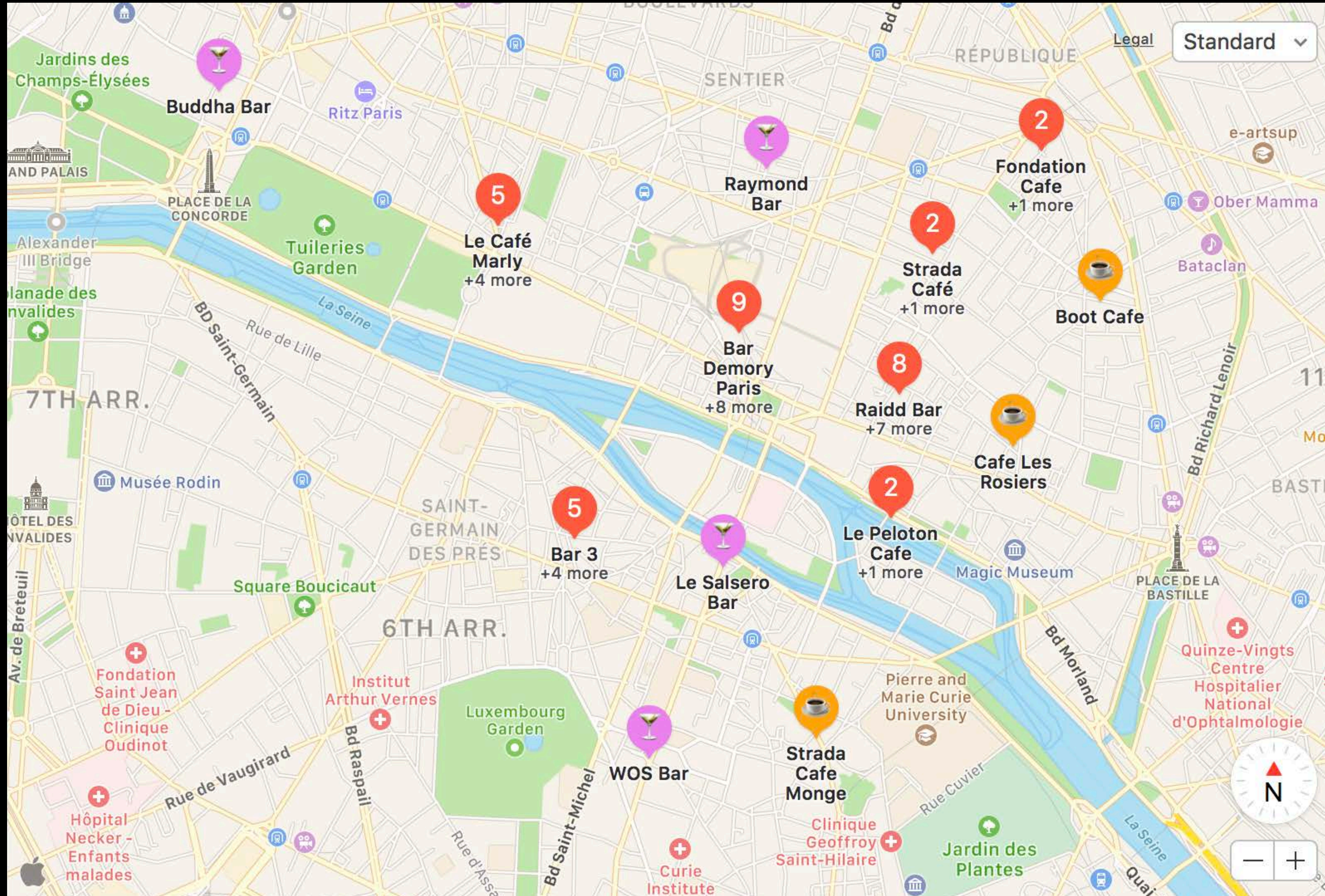
# Managing Clutter



# Annotation Collision with displayPriority



# Annotation Clustering with clusteringIdentifier



***Demo***

Using marker annotations

Julien Quint, MapKit JS Engineering

# Demo Recap

How to create marker annotations from Javascript objects

Set the display priority of annotations to unclutter the map

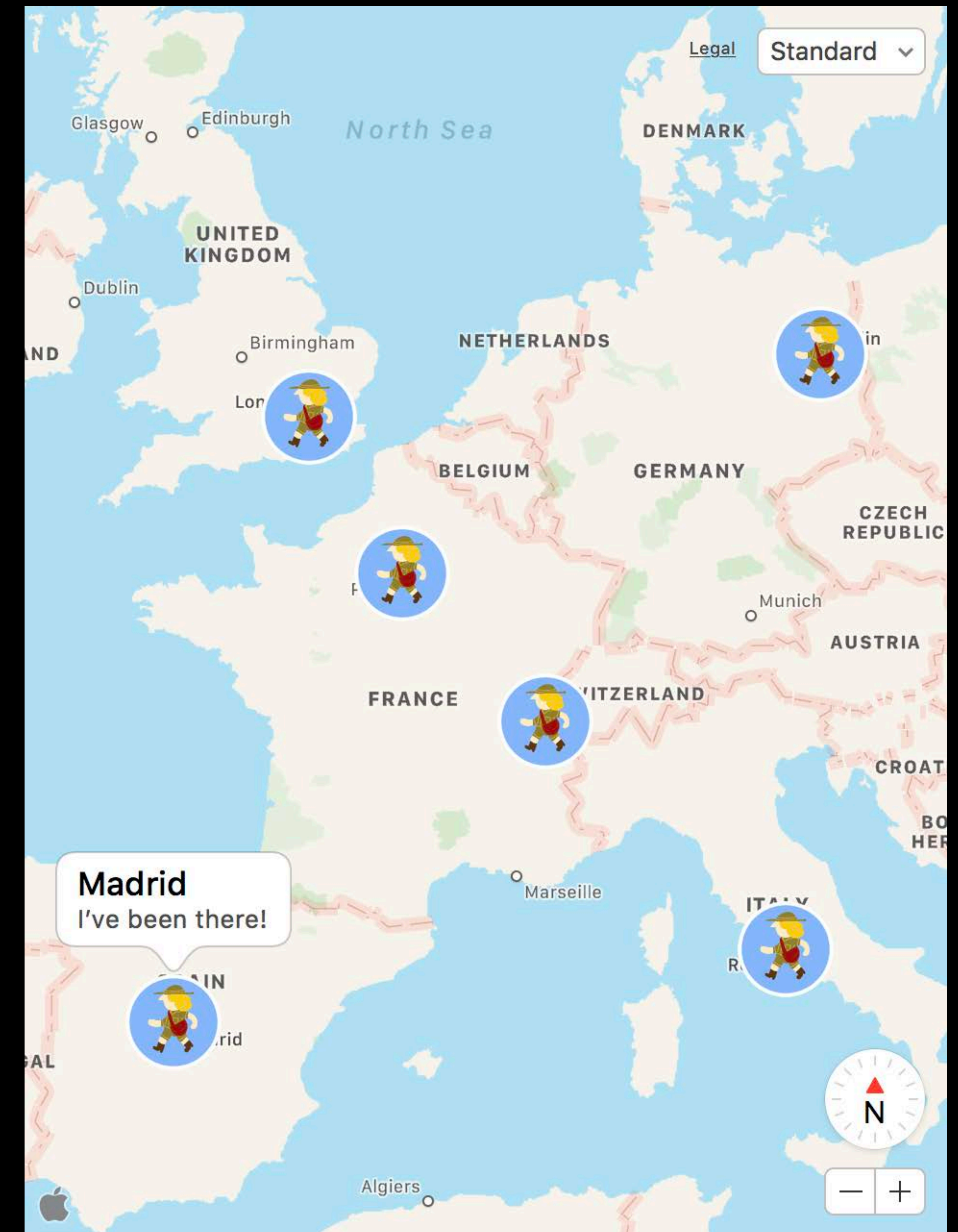
Style the annotations with a color, and glyph image or a glyph text

# Image Annotation

Use a raster image for the annotation

Title and subtitle shown in a callout bubble

```
new mapkit.ImageAnnotation(coordinate, {  
  title: "Madrid",  
  subtitle: "I've been there!",  
  url: {  
    1: "explorer.png",  
    2: "explorer_2x.png"  
  }  
});
```



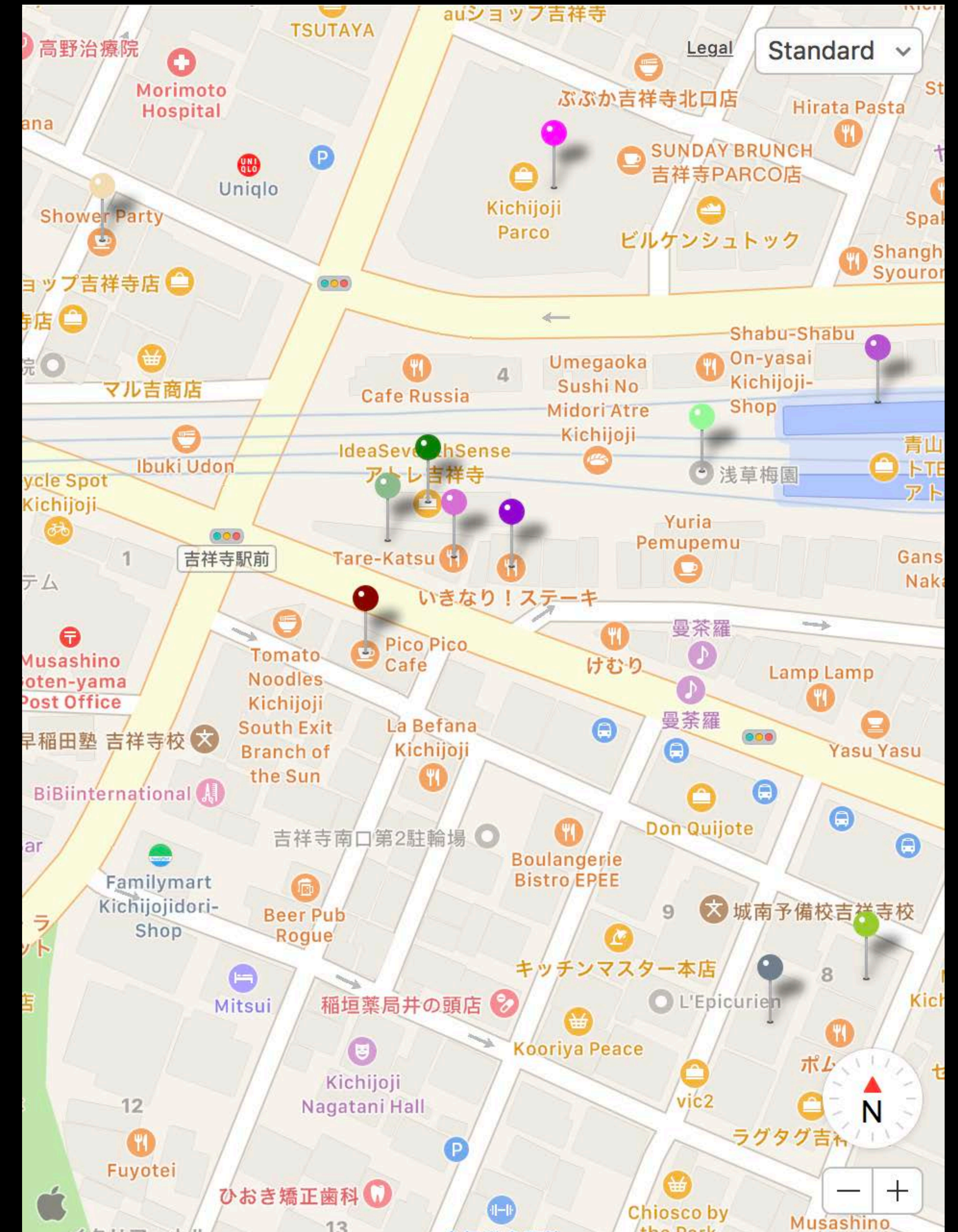


# Custom Annotation

Use a DOM element for the annotation

Elements are created on demand

```
new mapkit.Annotation(  
  coordinate,  
  (coordinate, options) => {  
    let canvas = document.createElement("canvas");  
    let context = canvas.getContext("2d");  
    // Draw the pin image and tint it  
    return canvas;  
  },  
  options  
);
```



# Navigating and Annotating the Map

Setting the center and region of the map

Marking locations with annotations

Covering areas with overlays

# Overlays

Show distances around a point with a circle overlay

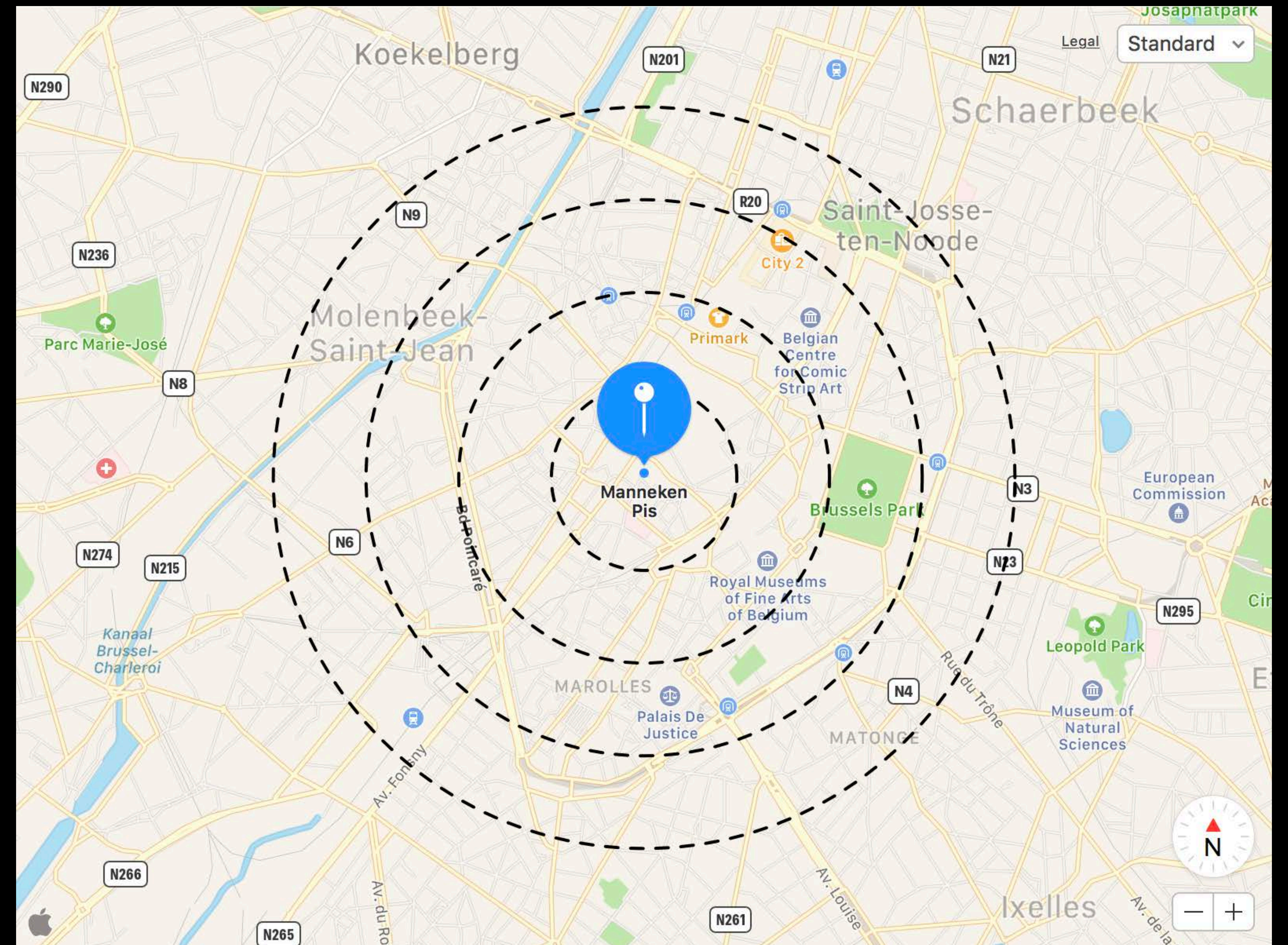
Show a route with a polyline overlay

Show geographical areas with polygon overlays

# Circle Overlay

```
let style = new mapkit.Style({
  lineWidth: 2,
  lineDash: [8, 8],
  strokeColor: "black",
  fillColor: null,
});

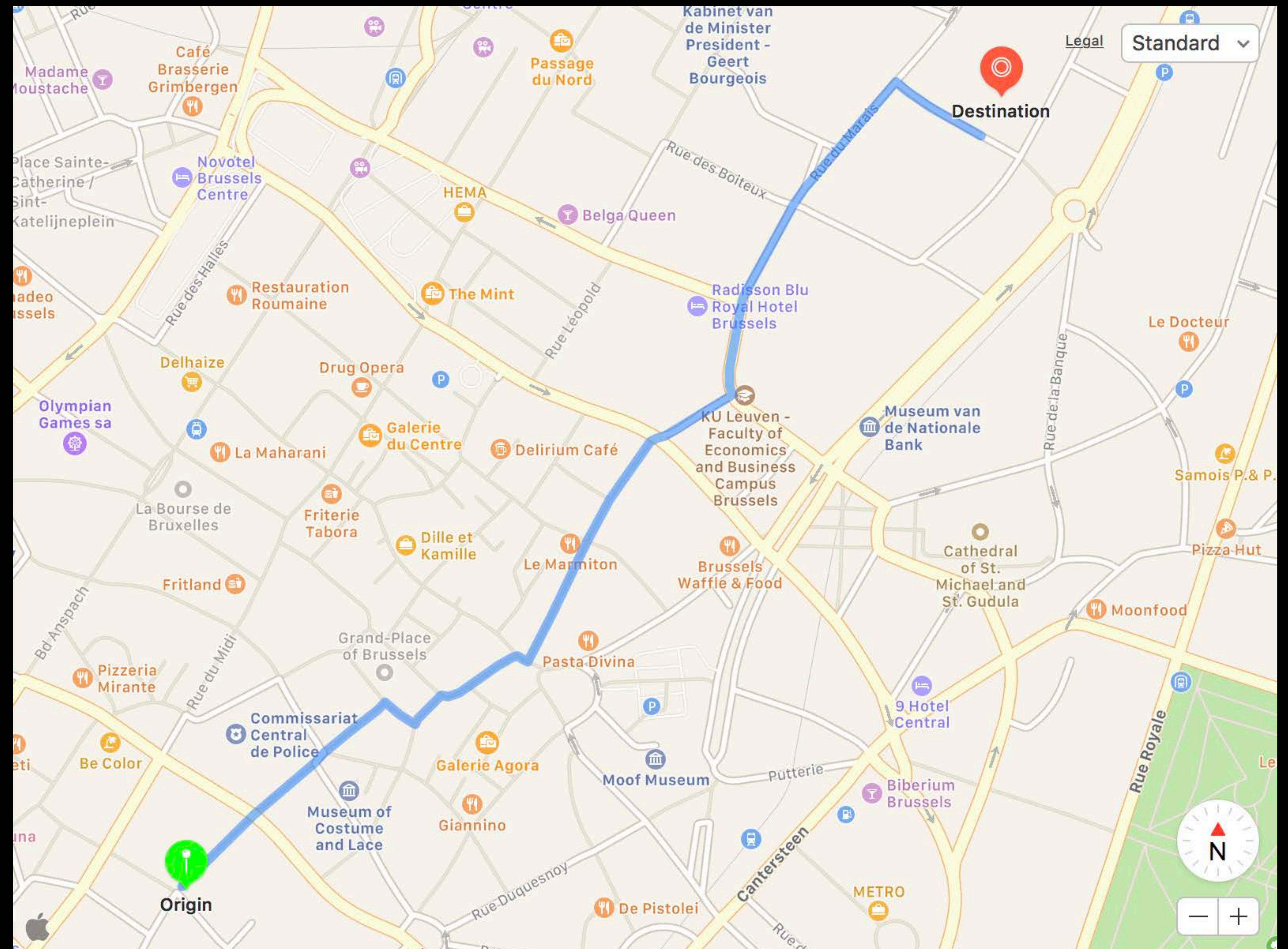
new mapkit.CircleOverlay(
  coordinate,
  radius, {
    style
  }
);
```



# Polyline Overlay

```
let style = new mapkit.Style({  
  strokeOpacity: 0.5,  
  lineWidth: 6  
});
```

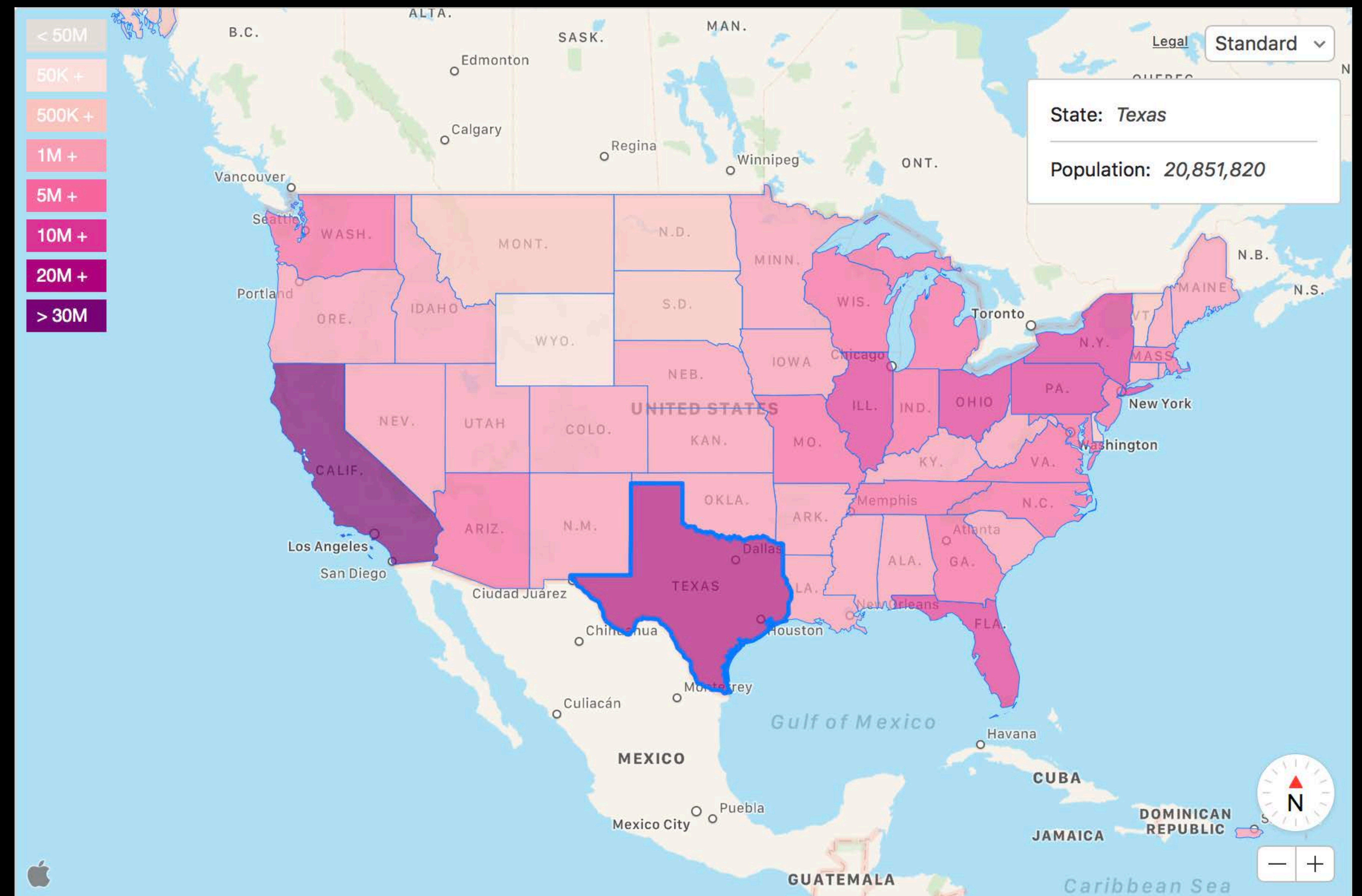
```
new mapkit.PolylineOverlay(  
  points, {  
    style  
  }  
);
```



# Polygon Overlay

```
let data = {  
  name: "Texas",  
  population: 20851820  
};
```

```
new mapkit.PolygonOverlay(  
  points, {  
    style,  
    data  
  }  
);
```

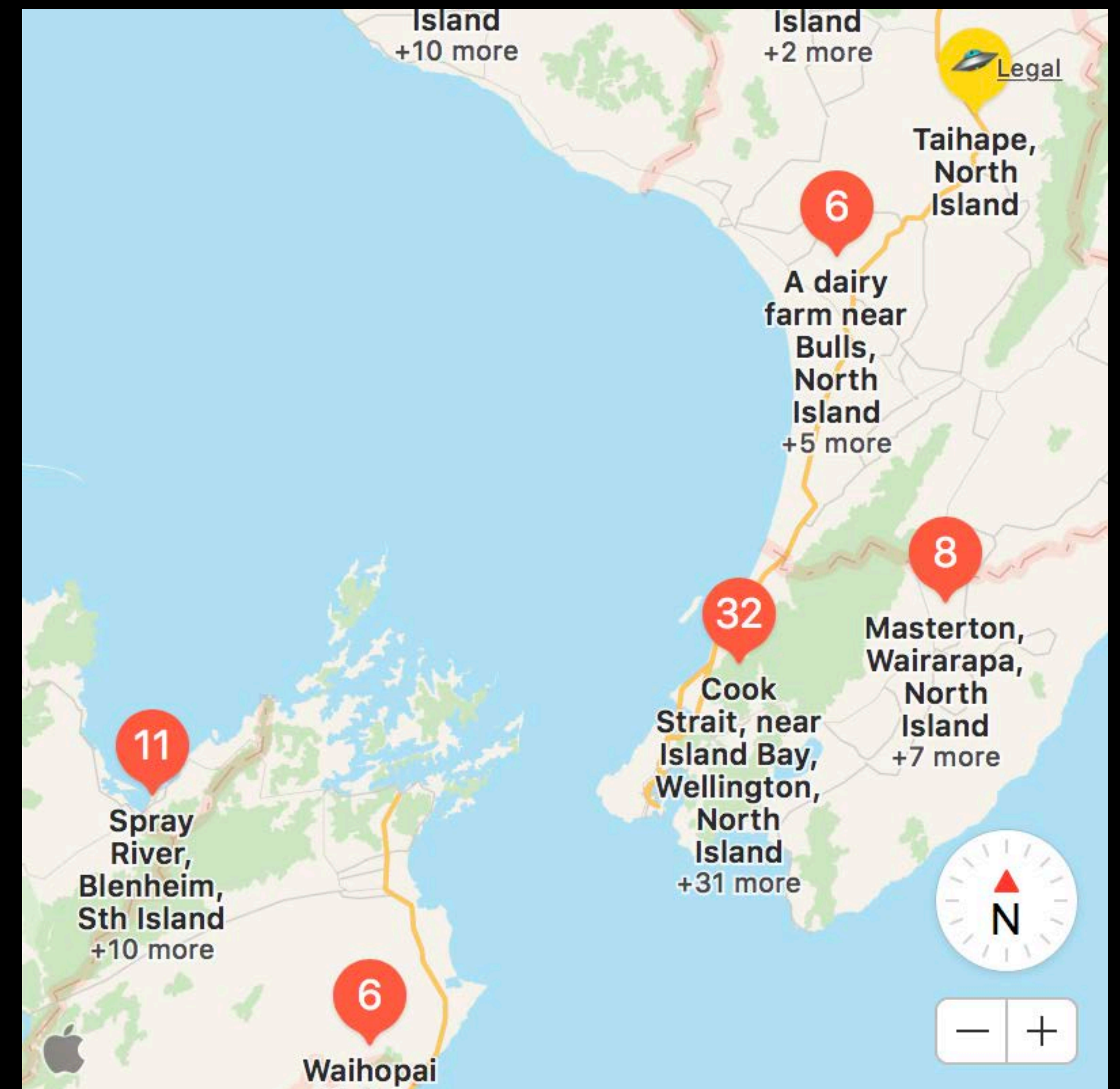


# GeoJSON Import

Import data from GeoJSON

Create annotations and overlays from GeoJSON geometry

Customize annotations and overlays through delegate methods



# Responding to User Interactions

Map, annotation and overlay objects send event notifications for user interactions

- Selection and deselection of annotations and overlays
- Dragging of annotations

Following the model of DOM events

- `map.addEventListener("select", event => { ... })`
- `annotation.removeEventListener("dragging", event => { ... })`



# Using MapKit JS on Your Website

## Agenda

Setting up your map

Navigating and annotating the map

Enabling rich interactions with services

# Enabling Rich Interactions with Services

Melody Kelly, MapKit JS Engineer

# Enabling Rich Interactions with Services

MapKit JS provides interfaces to Apple Maps services

- Geocoding
- Search and Search autocomplete
- Directions

# Using Apple Maps Services in MapKit JS

Using any service requires the same four steps:

1. Create an instance of service object
2. Specify request parameters and options
3. Make a request
4. Handle the response
  - Response is returned asynchronously via a callback function

# Geocoding

```
let geocoder = new mapkit.Geocoder();
geocoder.lookup("McEnergy Convention Center", (error, data) => {
  if (error) {
    return;
  }
  map.addAnnotation(new mapkit.MarkerAnnotation(data.results[0].coordinate));
});

geocoder.reverseLookup(new mapkit.Coordinate(37.3298857, -121.8888872),
  (error, data) => { ... });
```

# Geocoding

```
let geocoder = new mapkit.Geocoder();
geocoder.lookup("McEnergy Convention Center", (error, data) => {
    if (error) {
        return;
    }
    map.addAnnotation(new mapkit.MarkerAnnotation(data.results[0].coordinate));
});

geocoder.reverseLookup(new mapkit.Coordinate(37.3298857, -121.8888872),
    (error, data) => { ... });
```

# Geocoding

```
let geocoder = new mapkit.Geocoder({ getUserLocation: true });
geocoder.lookup("McEney Convention Center", (error, data) => {
  if (error) {
    return;
  }
  map.addAnnotation(new mapkit.MarkerAnnotation(data.results[0].coordinate));
});

geocoder.reverseLookup(new mapkit.Coordinate(37.3298857, -121.8888872),
  (error, data) => { ... });
```

# Geocoding

```
let geocoder = new mapkit.Geocoder({ getUserLocation: true });
geocoder.lookup("McEnergy Convention Center", (error, data) => {
  if (error) {
    return;
  }
  map.addAnnotation(new mapkit.MarkerAnnotation(data.results[0].coordinate));
});

geocoder.reverseLookup(new mapkit.Coordinate(37.3298857, -121.8888872),
  (error, data) => { ... });
```



# Geocoding

```
let geocoder = new mapkit.Geocoder({ getUserLocation: true });
geocoder.lookup("McEney Convention Center", (error, data) => {
  if (error) {
    return;
  }
  map.addAnnotation(new mapkit.MarkerAnnotation(data.results[0].coordinate));
});

geocoder.reverseLookup(new mapkit.Coordinate(37.3298857, -121.8888872),
  (error, data) => { ... });
```

# Geocoding

```
let geocoder = new mapkit.Geocoder({ getUserLocation: true });
geocoder.lookup("McEney Convention Center", (error, data) => {
  if (error) {
    return;
  }
  map.addAnnotation(new mapkit.MarkerAnnotation(data.results[0].coordinate));
});

geocoder.reverseLookup(new mapkit.Coordinate(37.3298857, -121.8888872),
  (error, data) => { ... });
```

# Geocoding

```
let geocoder = new mapkit.Geocoder({ getUserLocation: true });
geocoder.lookup("McEnergy Convention Center", (error, data) => {
  if (error) {
    return;
  }
  map.addAnnotation(new mapkit.MarkerAnnotation(data.results[0].coordinate));
});

geocoder.reverseLookup(new mapkit.Coordinate(37.3298857, -121.8888872),
  (error, data) => { ... });
```

# Service Context

Provide context to search or geocoder by setting properties in the constructor or request

- `getUserLocation`
- `coordinate`
- `region`

# Search

```
let search = new mapkit.Search({ getUserLocation: true });
search.search("coffee", (error, data) => {
  if (error) {
    // handle search error
    return;
  }
  let annotations = data.places.map(place => {
    let annotation = new mapkit.MarkerAnnotation(place.coordinate);
    annotation.title = place.name;
    return annotation;
  });
  map.showItems(annotations);
});
```

# Search

```
let search = new mapkit.Search({ getUserLocation: true });
search.search("coffee", (error, data) => {
  if (error) {
    // handle search error
    return;
  }
  let annotations = data.places.map(place => {
    let annotation = new mapkit.MarkerAnnotation(place.coordinate);
    annotation.title = place.name;
    return annotation;
  });
  map.showItems(annotations);
});
```

# Search

```
let search = new mapkit.Search({ getUserLocation: true });
search.search("coffee", (error, data) => {
  if (error) {
    // handle search error
    return;
  }
  let annotations = data.places.map(place => {
    let annotation = new mapkit.MarkerAnnotation(place.coordinate);
    annotation.title = place.name;
    return annotation;
  });
  map.showItems(annotations);
});
```

# Search

```
let search = new mapkit.Search({ getUserLocation: true });
search.search("coffee", (error, data) => {
  if (error) {
    // handle search error
    return;
  }
  let annotations = data.places.map(place => {
    let annotation = new mapkit.MarkerAnnotation(place.coordinate);
    annotation.title = place.name;
    return annotation;
  });
  map.showItems(annotations);
});
```



# Search

```
let search = new mapkit.Search({ getUserLocation: true });
search.search("coffee", (error, data) => {
  if (error) {
    // handle search error
    return;
  }
  let annotations = data.places.map(place => {
    let annotation = new mapkit.MarkerAnnotation(place.coordinate);
    annotation.title = place.name;
    return annotation;
  });
  map.showItems(annotations);
});
```

# Search

```
let search = new mapkit.Search({ getUserLocation: true });
search.search("coffee", (error, data) => {
  if (error) {
    // handle search error
    return;
  }
  let annotations = data.places.map(place => {
    let annotation = new mapkit.MarkerAnnotation(place.coordinate);
    annotation.title = place.name;
    return annotation;
  });
  map.showItems(annotations);
});
```

# Search

```
let search = new mapkit.Search({ getUserLocation: true });
search.search("coffee", (error, data) => {
  if (error) {
    // handle search error
    return;
  }
  let annotations = data.places.map(place => {
    let annotation = new mapkit.MarkerAnnotation(place.coordinate);
    annotation.title = place.name;
    return annotation;
  });
  map.showItems(annotations);
});
```

# Search

```
let search = new mapkit.Search({ getUserLocation: true });
search.search("coffee", (error, data) => {
  if (error) {
    // handle search error
    return;
  }
  let annotations = data.places.map(place => {
    let annotation = new mapkit.MarkerAnnotation(place.coordinate);
    annotation.title = place.name;
    return annotation;
  });
  map.showItems(annotations);
});
```

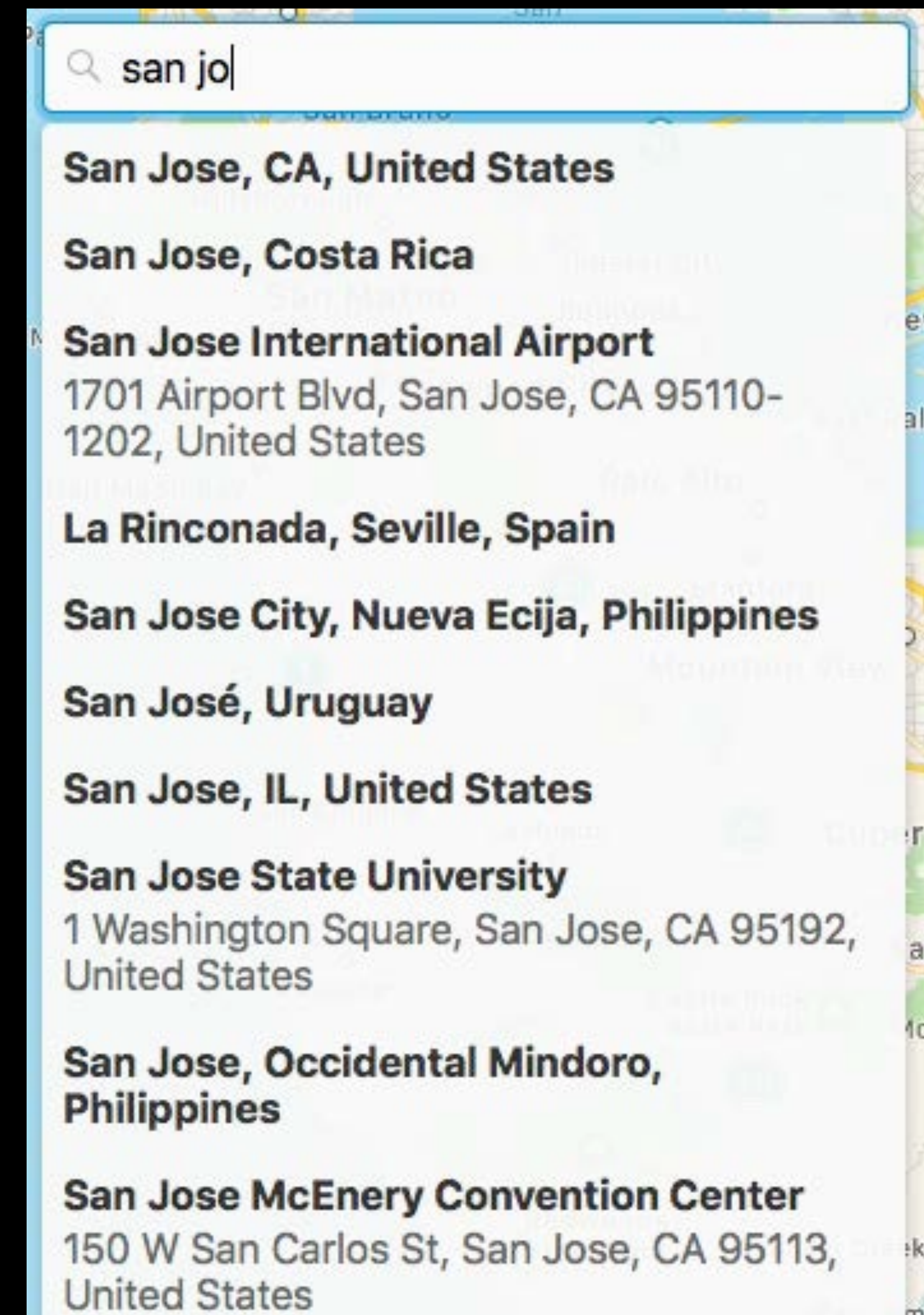
# Search AutoComplete

```
Search.autocomplete("san", (error, data) => {  
  if (error) {  
    // handle search autocomplete error  
    return;  
  }  
  // handle data  
});
```



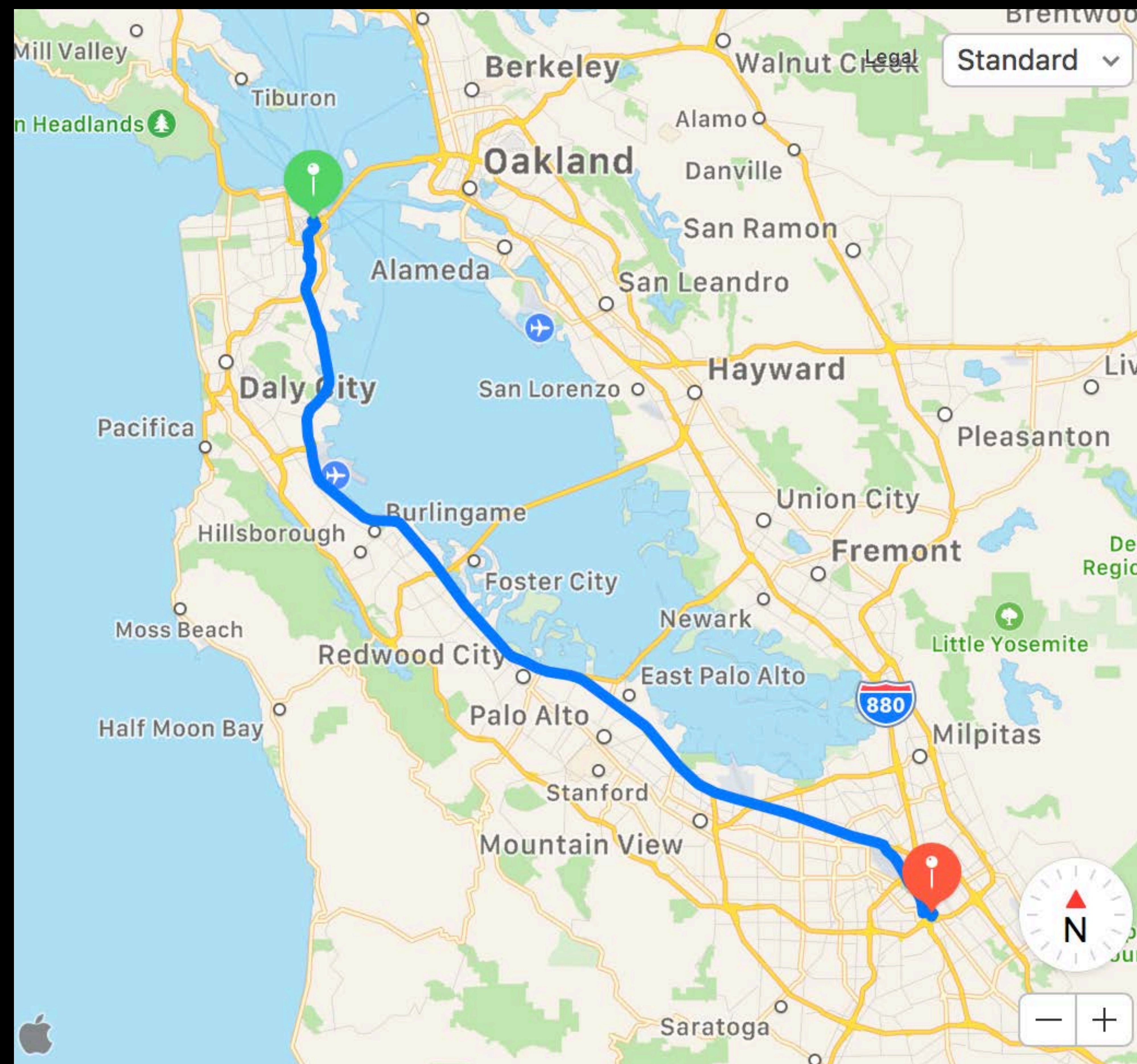
# Search AutoComplete

```
Search.autocomplete("san", (error, data) => {  
  if (error) {  
    // handle search autocomplete error  
    return;  
  }  
  // handle data  
});
```



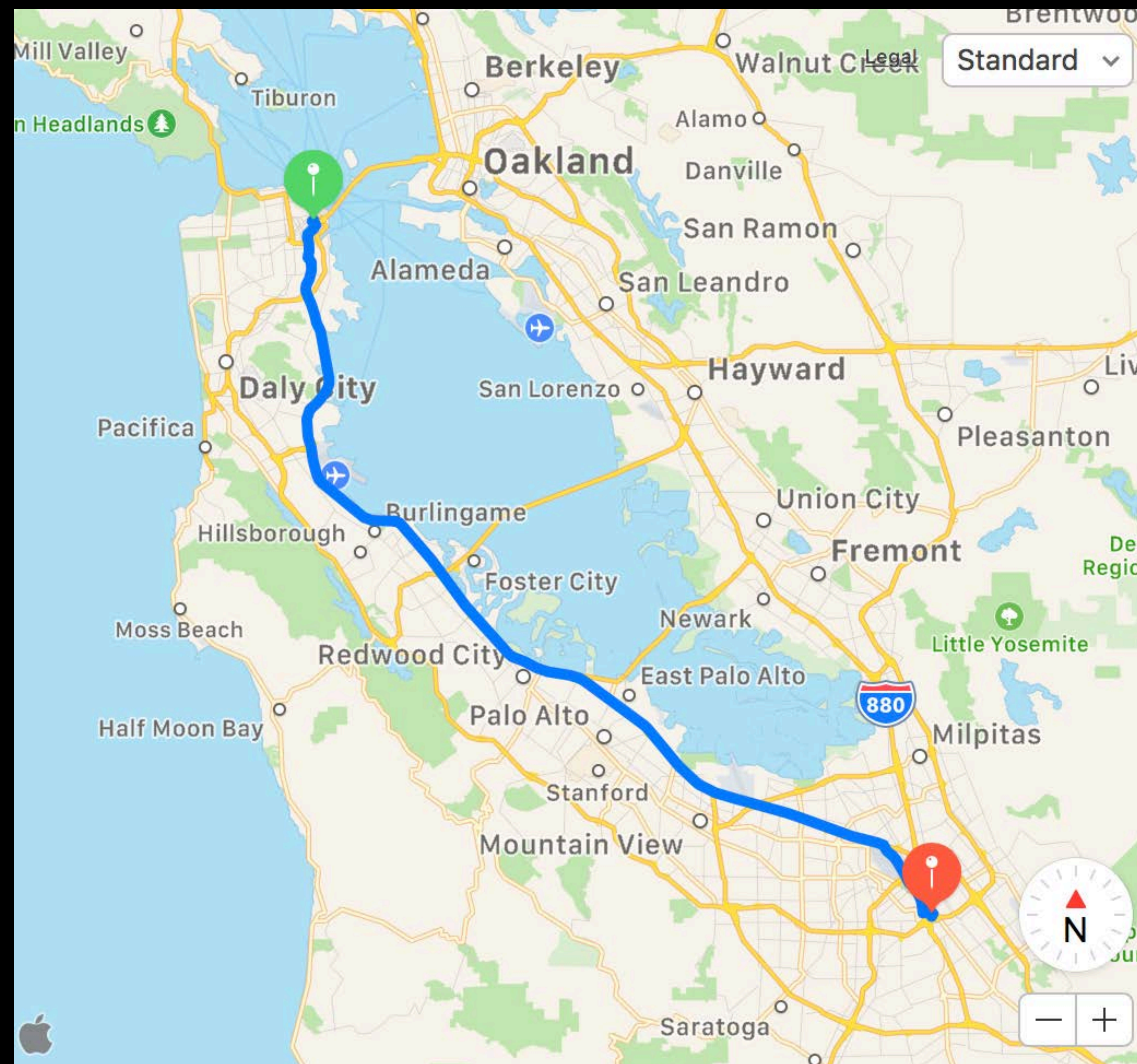
# Directions

```
let directions = new mapkit.Directions();
directions.route({
  origin: "747 Howard St, San Francisco",
  destination: "150 W San Carlos St, San Jose"
}, (error, data) => {
  // display route steps and path
});
```



# Directions

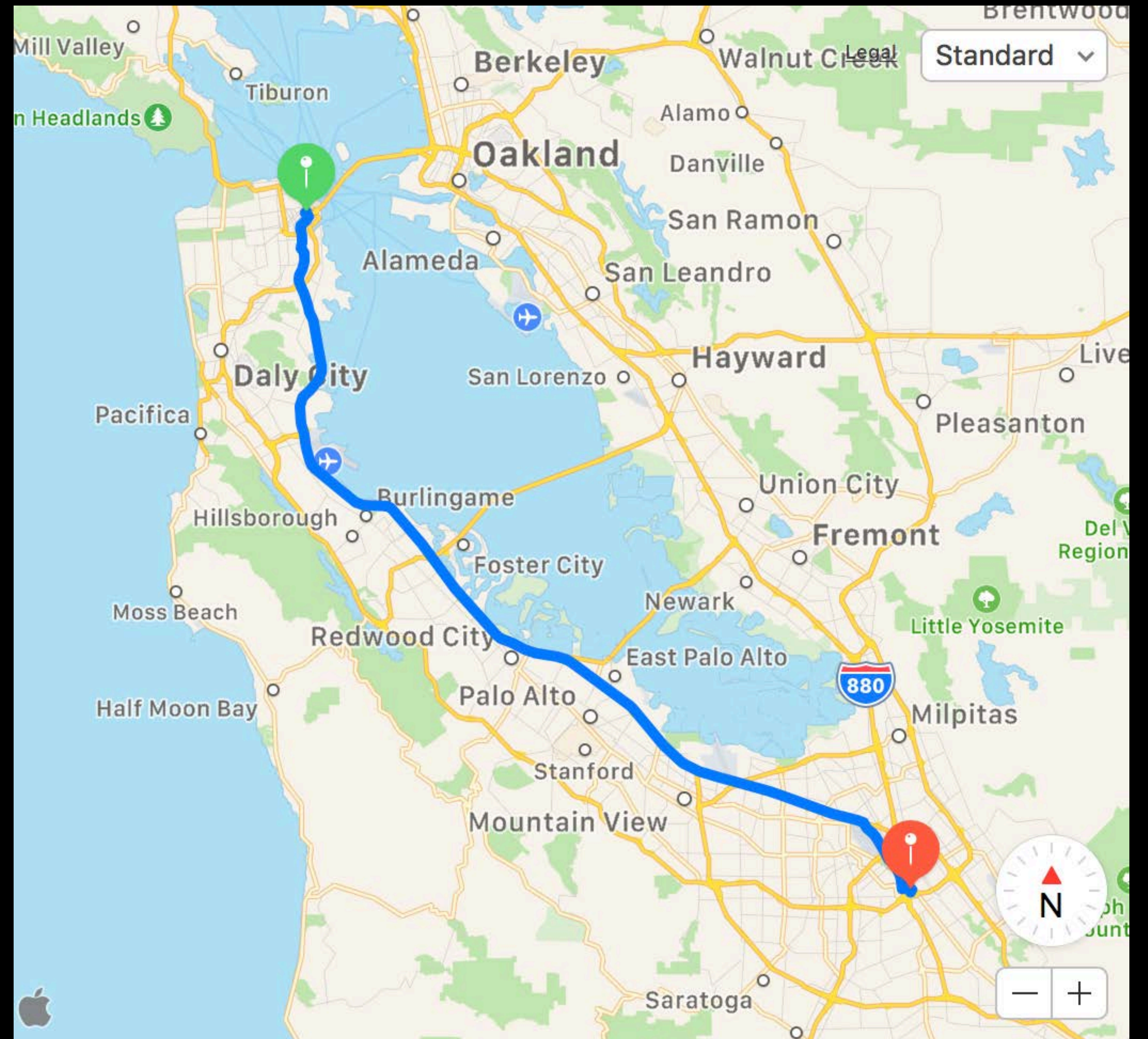
```
let directions = new mapkit.Directions();
directions.route({
  origin: "747 Howard St, San Francisco",
  destination: "150 W San Carlos St, San Jose"
}, (error, data) => {
  // display route steps and path
});
```





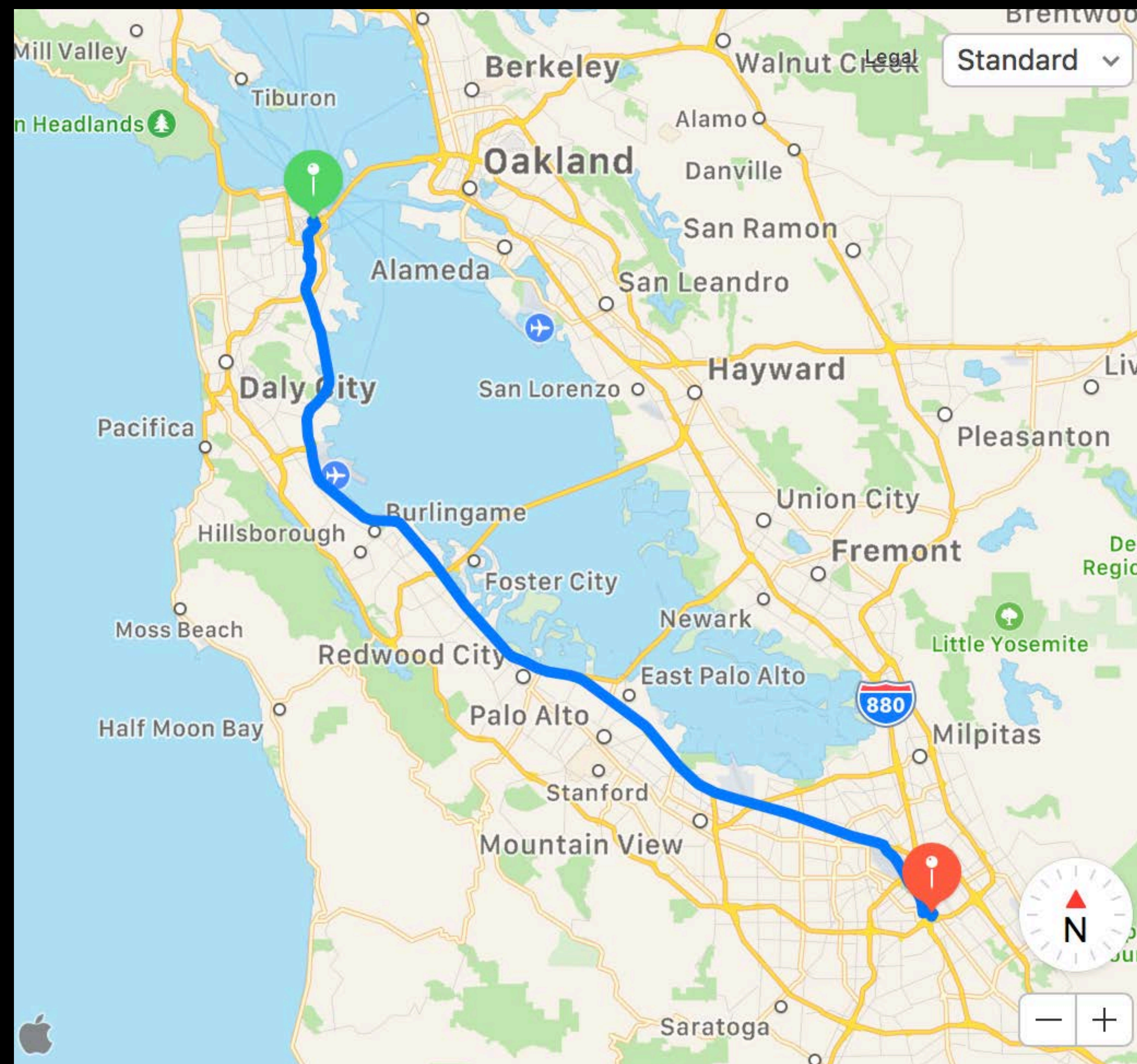
# Directions

```
let directions = new mapkit.Directions(  
  { language: "fi-FI" });  
directions.route({  
  origin: "747 Howard St, San Francisco",  
  destination: "150 W San Carlos St, San Jose"  
}, (error, data) => {  
  // display route steps and path  
});
```



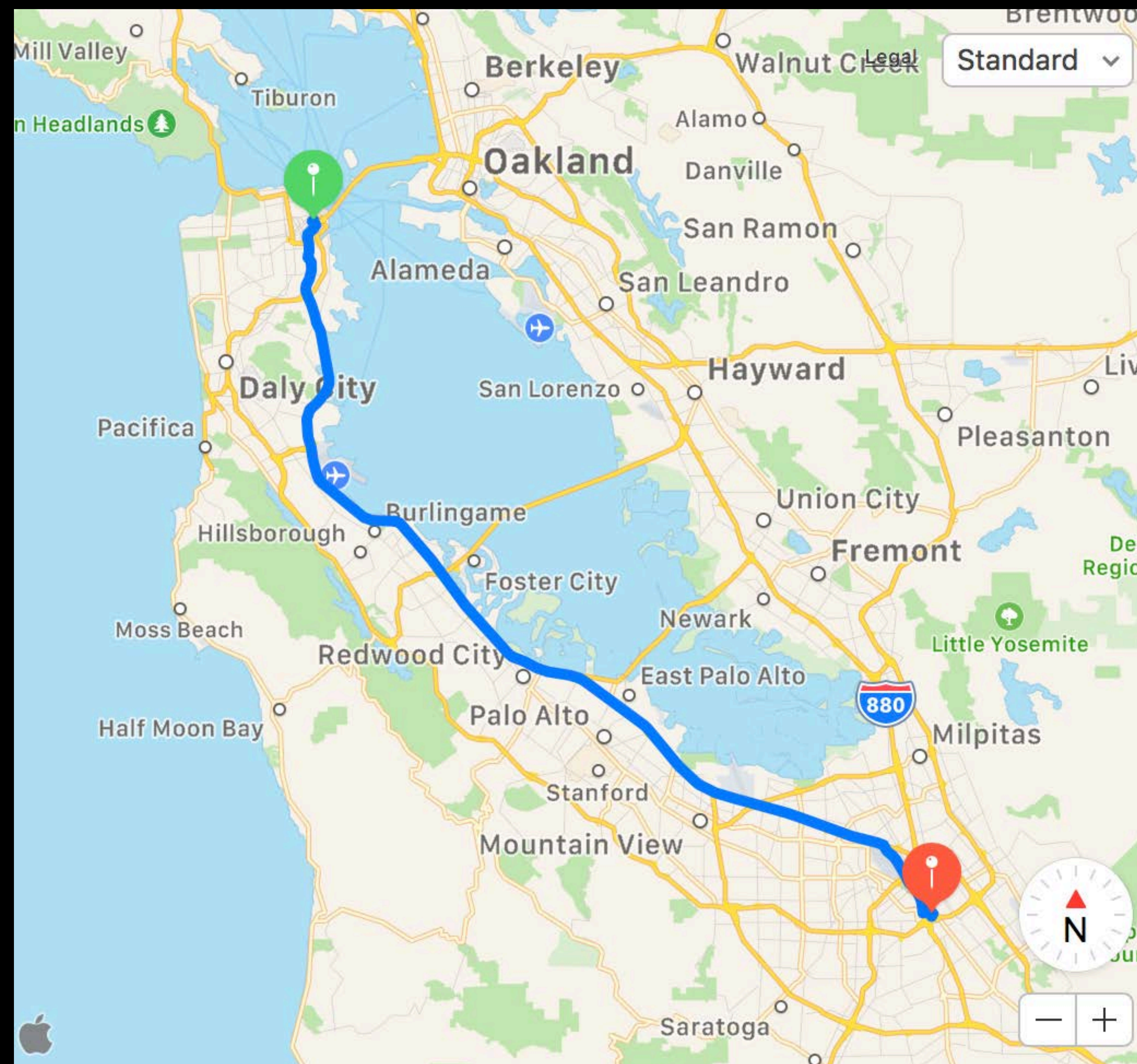
# Directions

```
let directions = new mapkit.Directions();
directions.route({
  origin: "747 Howard St, San Francisco",
  destination: "150 W San Carlos St, San Jose"
}, (error, data) => {
  // display route steps and path
});
```



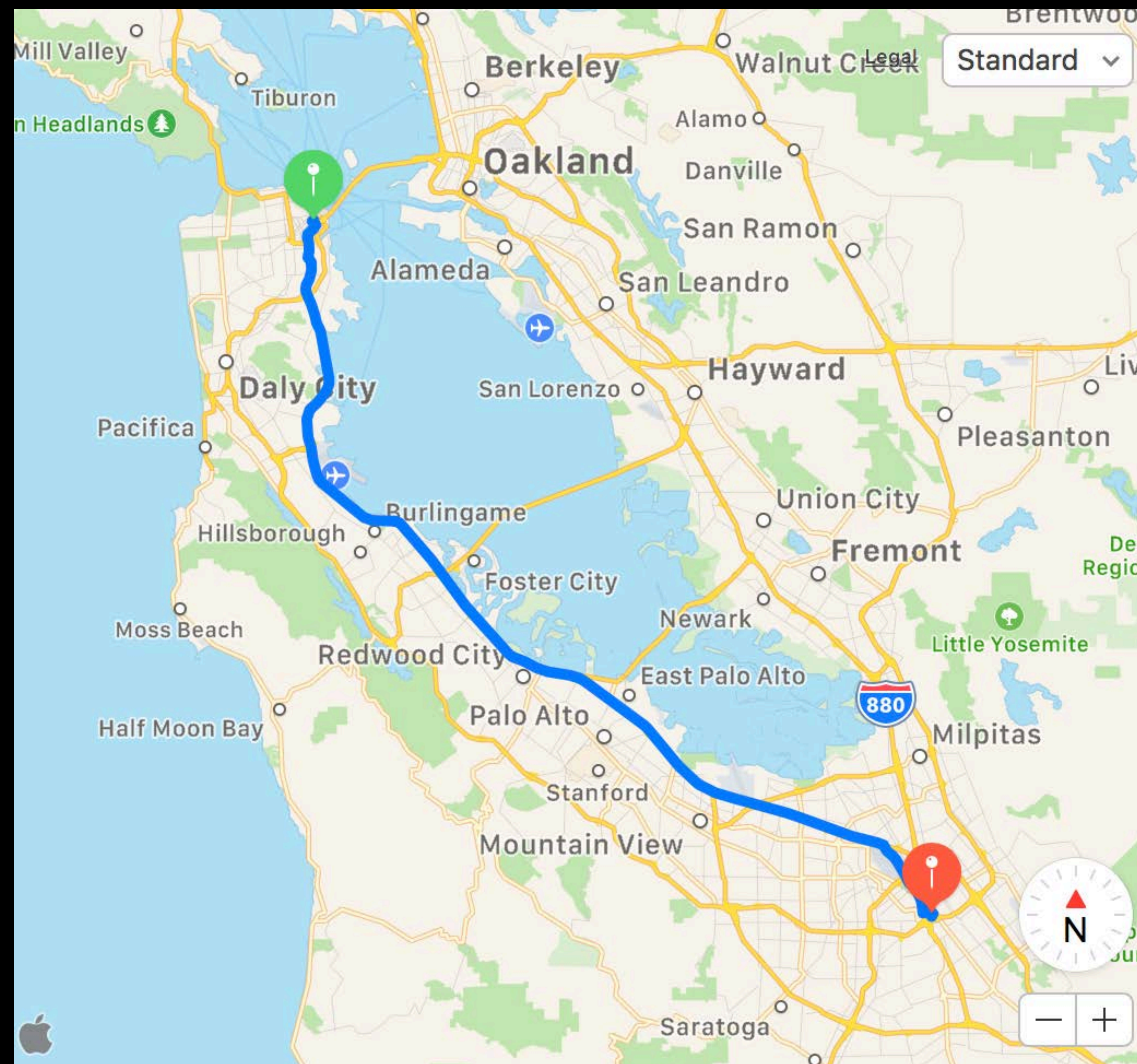
# Directions

```
let directions = new mapkit.Directions();
directions.route({
  origin: "747 Howard St, San Francisco",
  destination: "150 W San Carlos St, San Jose"
}, (error, data) => {
  // display route steps and path
});
```



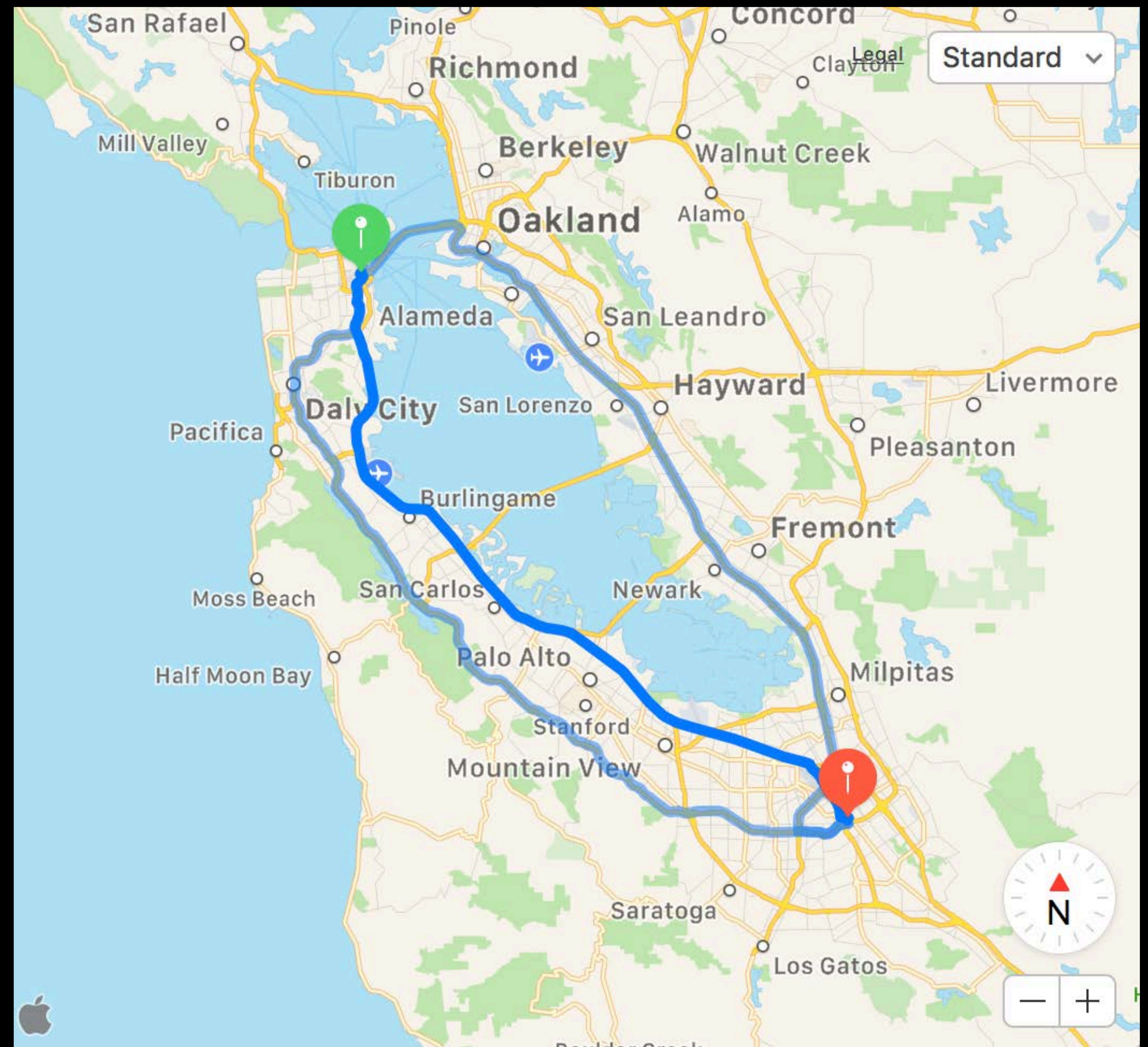
# Directions

```
let directions = new mapkit.Directions();
directions.route({
  origin: "747 Howard St, San Francisco",
  destination: "150 W San Carlos St, San Jose",
  transportType: Directions.Transport.Walking
}, (error, data) => {
  // display route steps and path
});
```



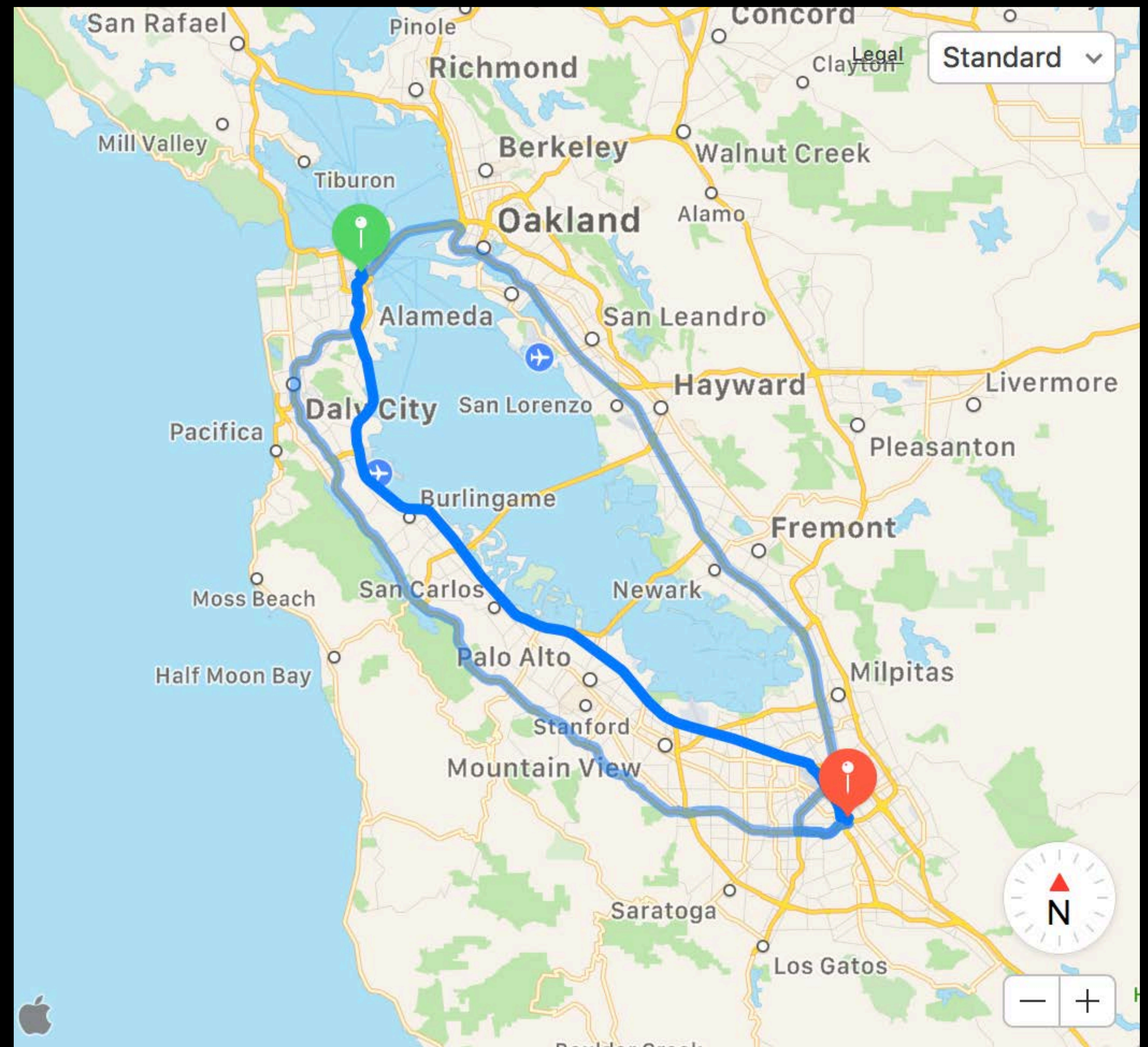
# Directions

```
let directions = new mapkit.Directions();
directions.route({
  origin: "747 Howard St, San Francisco",
  destination: "150 W San Carlos St, San Jose",
  requestsAlternateRoutes: true
}, (error, data) => {
  // display route steps and path
});
```



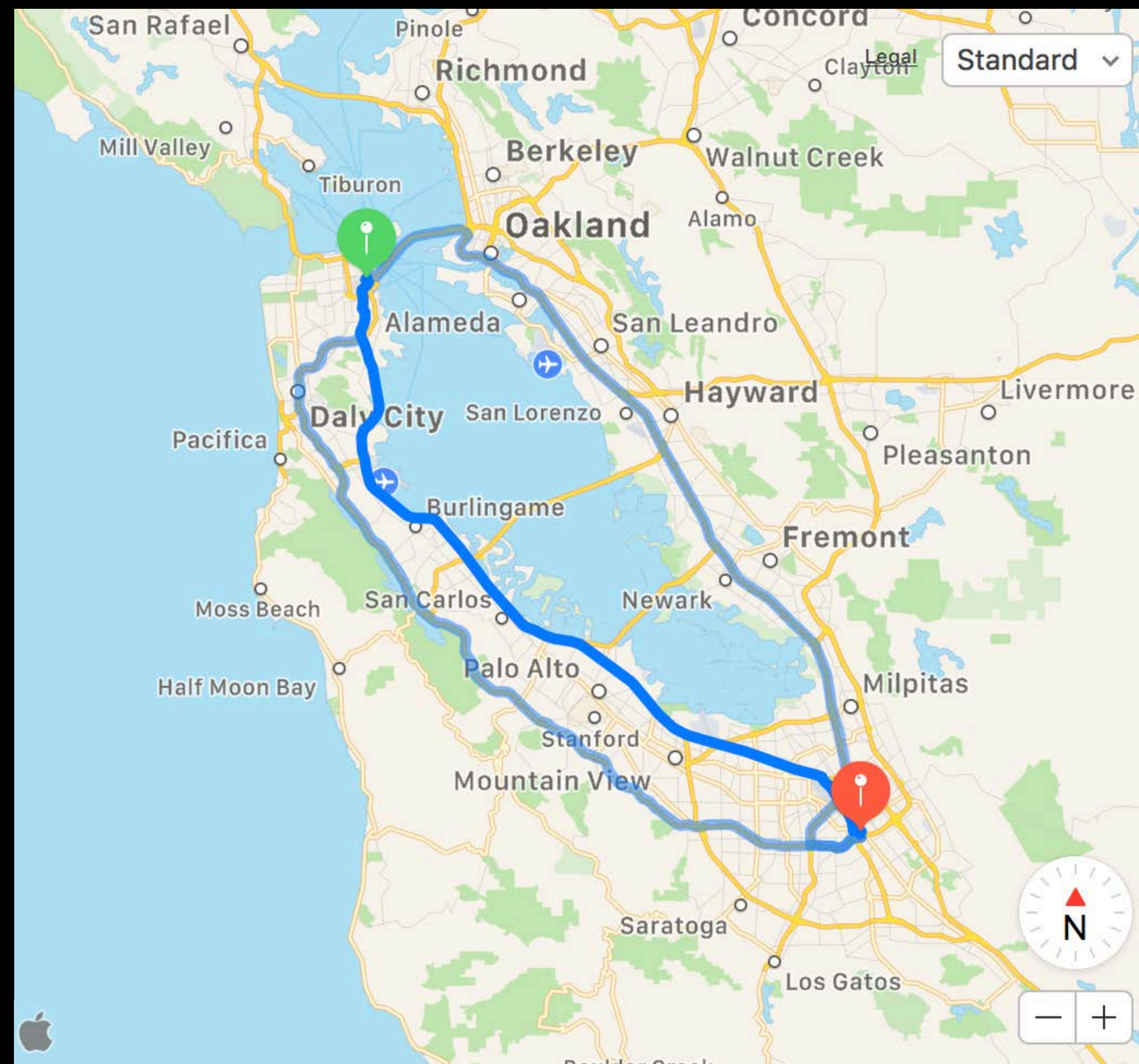
# Directions

```
let directions = new mapkit.Directions();
directions.route({
  origin: "747 Howard St, San Francisco",
  destination: "150 W San Carlos St, San Jose",
  requestsAlternateRoutes: true
}, (error, data) => {
  // display route steps and path
});
```



# Directions

```
let directions = new mapkit.Directions();
directions.route({
  origin: "747 Howard St, San Francisco",
  destination: "150 W San Carlos St, San Jose",
  requestsAlternateRoutes: true
}, (error, data) => {
  // display route steps and path
});
```



***Demo***

Directions and user interactions

Julien Quint, MapKit JS Engineering



# Demo Recap

How to react to user events

Get driving directions from the MapKit JS service

Draw polyline overlays

Implement selection behavior for overlays

# Key Takeaways

# Key Takeaways

MapKit JS delivers a top-notch map experience on the web

# Key Takeaways

MapKit JS delivers a top-notch map experience on the web

Unify on a single map provider with familiar, flexible APIs

# Key Takeaways

MapKit JS delivers a top-notch map experience on the web

Unify on a single map provider with familiar, flexible APIs

Get a MapKit JS key at [developer.apple.com](https://developer.apple.com) and try it out today!

<https://developer.apple.com/maps/mapkitjs>

# More Information

<https://developer.apple.com/wwdc18/212>

---

Getting and Using a MapKit JS Key

WWDC 2018 Video

---

Mapping and Location Technologies Lab

Technology Lab 3

Wednesday 9:00AM

---

Mapping and Location Technologies Lab

Technology Lab 4

Friday 9:00AM

---

 **WWDC18**