

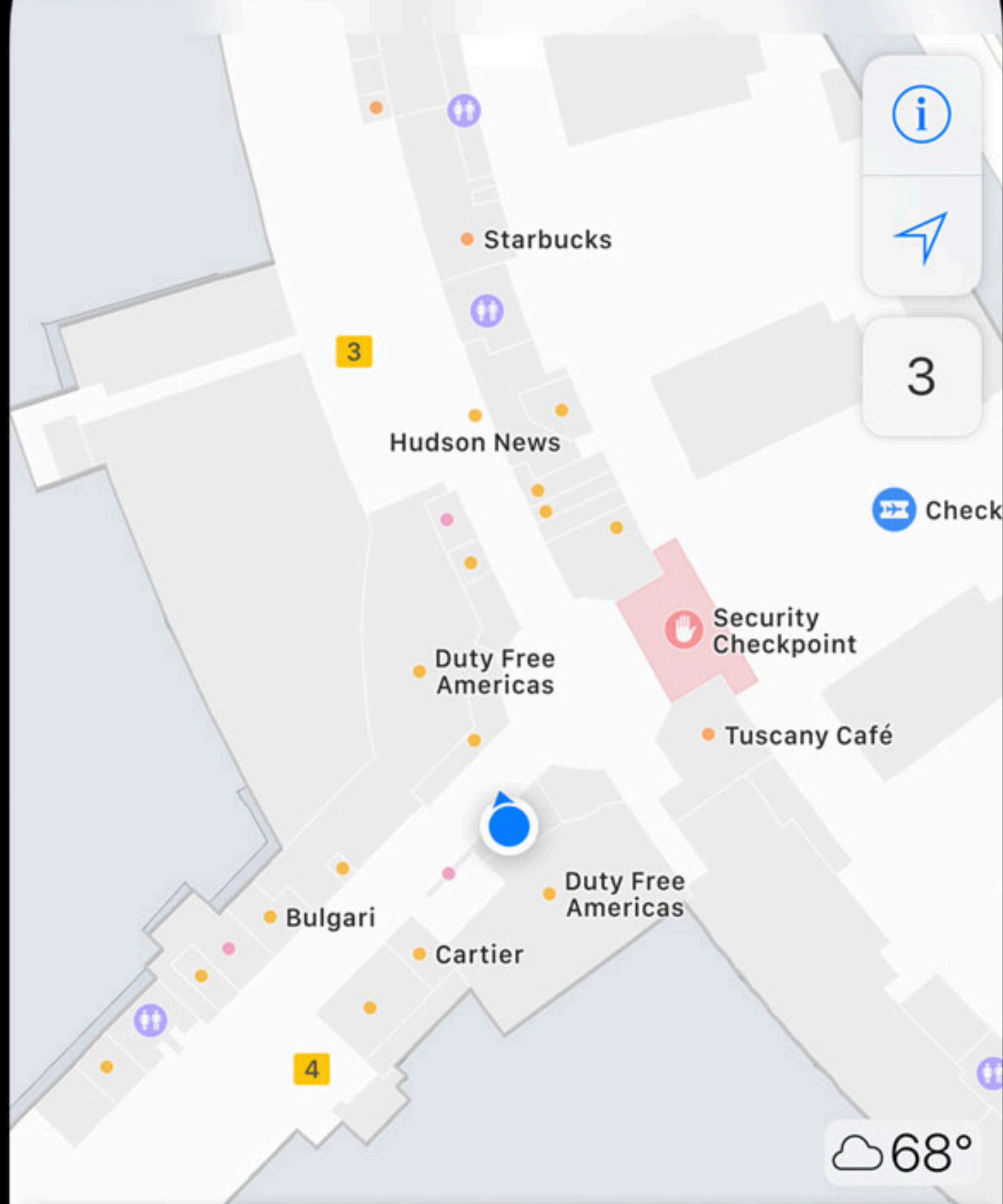
#WWDC19

# Adding Indoor Maps to Your App and Website

Stephane Godbillon, Maps Engineer

Mithilesh Kumar, Maps Engineer

9:41



3



Check

68°

### John F. Kennedy



### International Airport

International Airport

★★★★★ (1592) on Yelp



Terminals



Check-ins



Gates



Bag Claims



Food



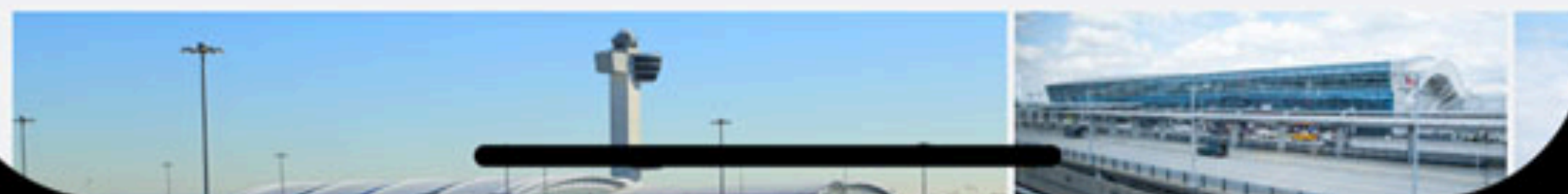
Drinks



Shops



Restrooms





Indoor Mapping Data Format (IMDF)

Indoor Maps Program

Displaying Indoor Maps in an iOS App

Displaying Indoor Maps in a Web App

Indoor Mapping Data Format (IMDF)

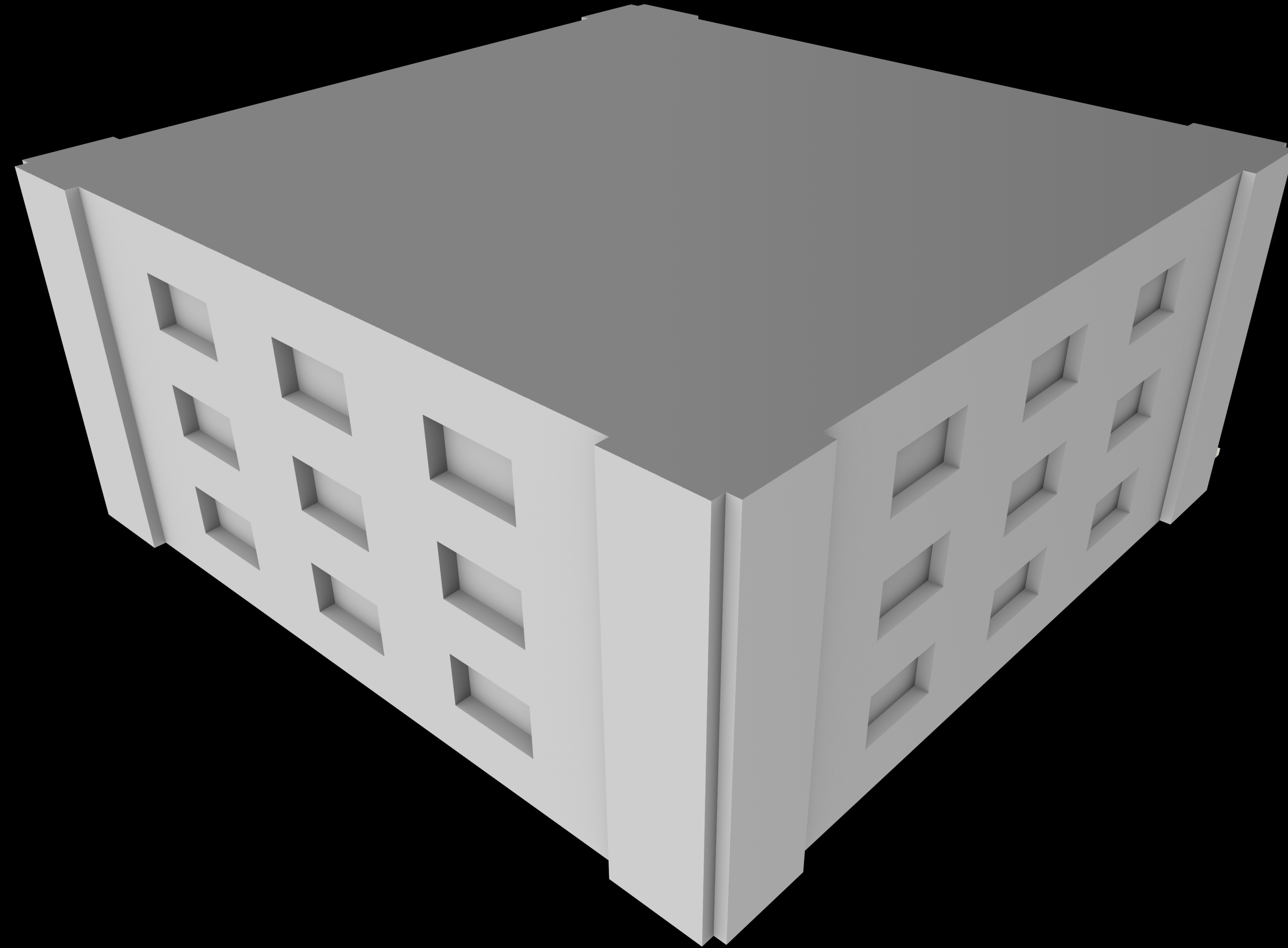
Indoor Maps Program

Displaying Indoor Maps in an iOS App

Displaying Indoor Maps in a Web App

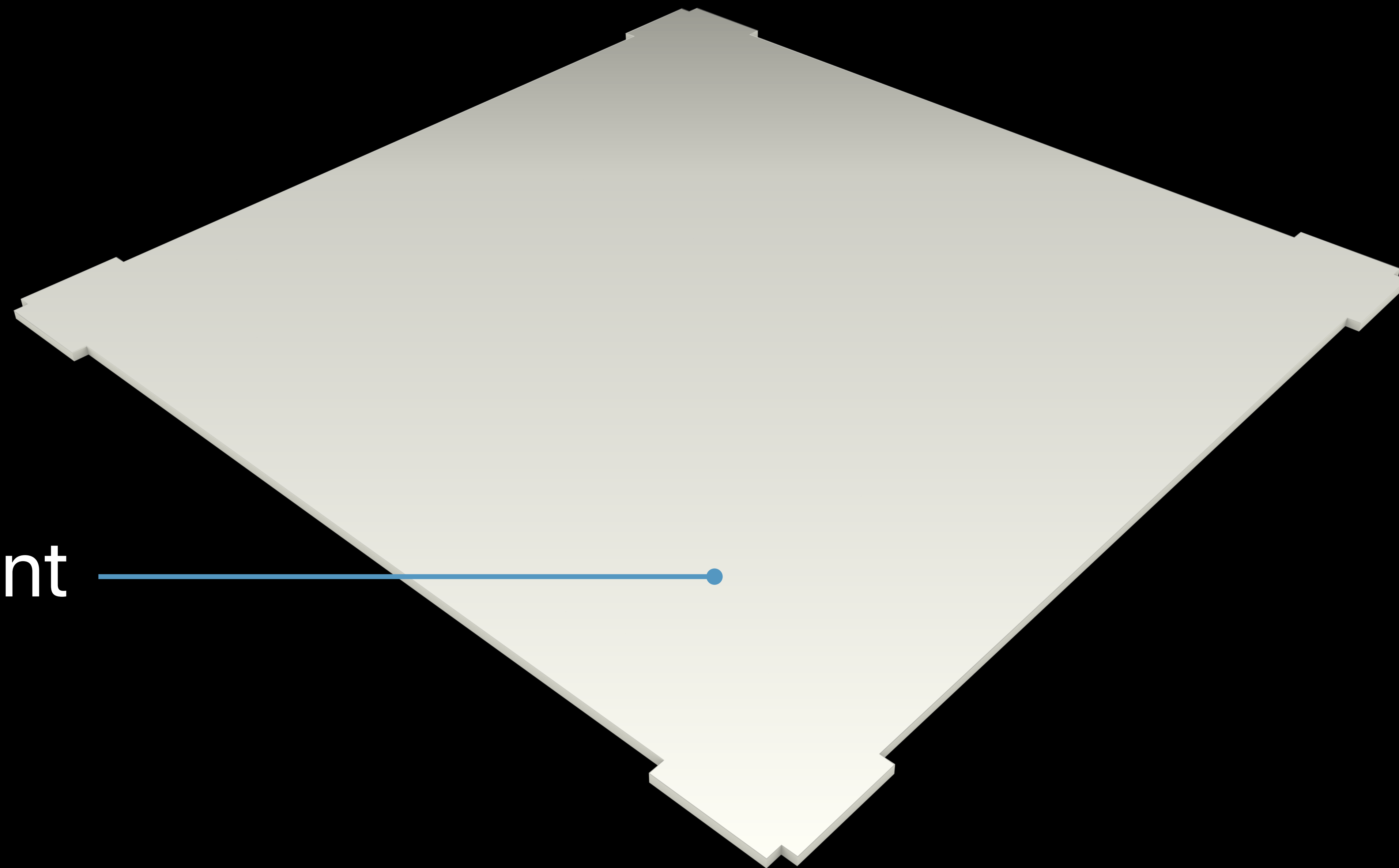


# Indoor Map



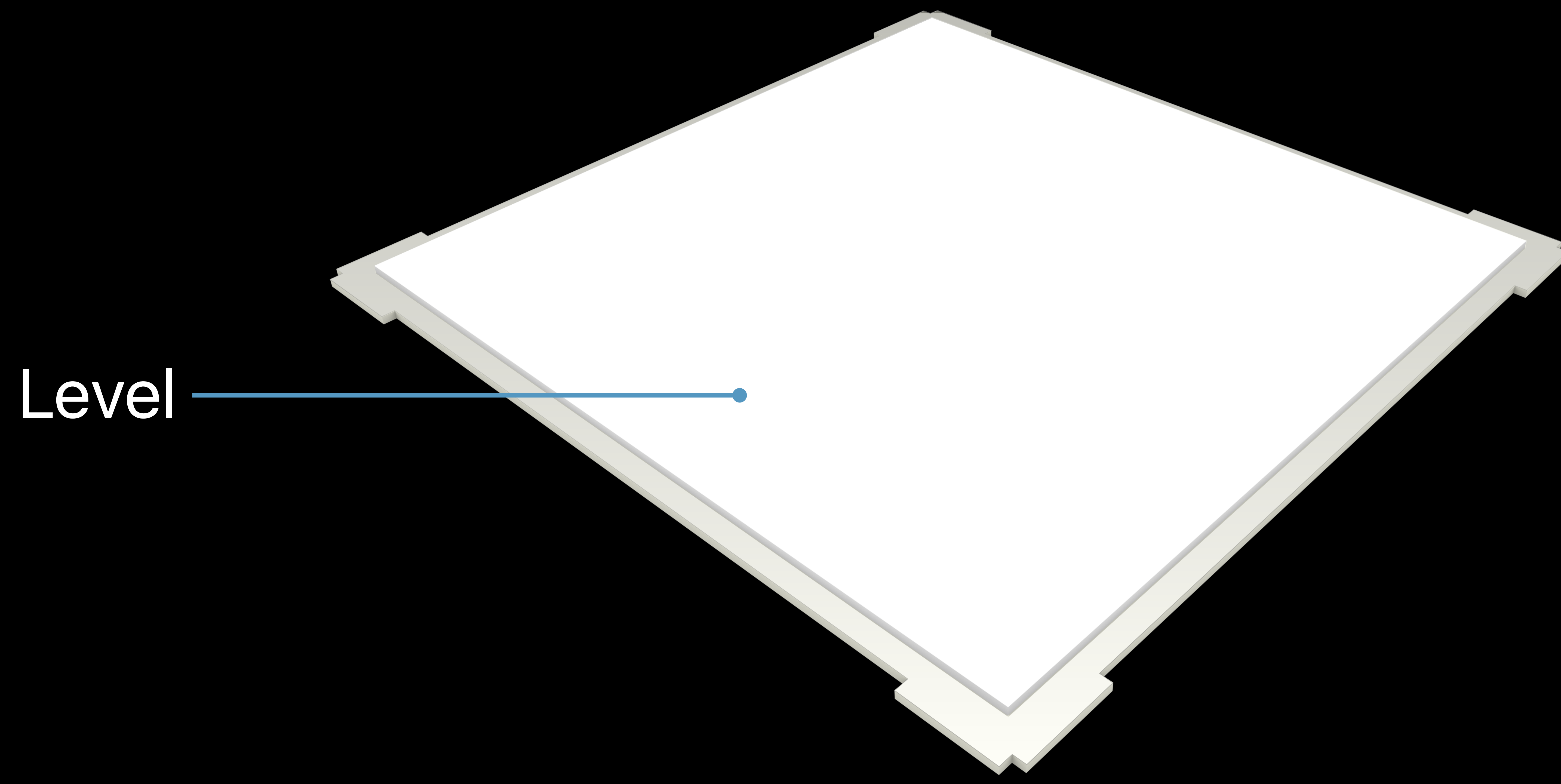
# Indoor Map

Building Footprint



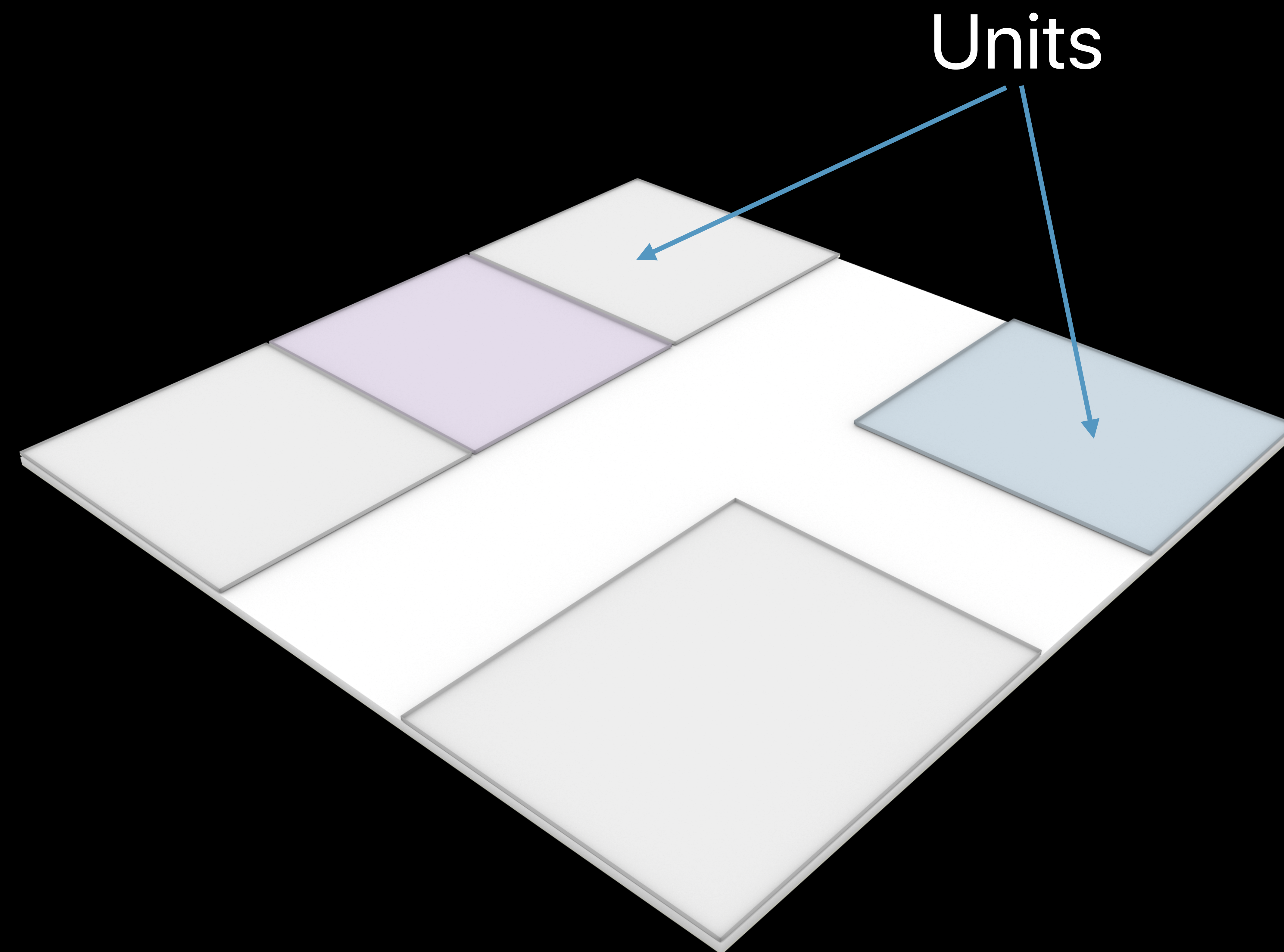


# Indoor Map



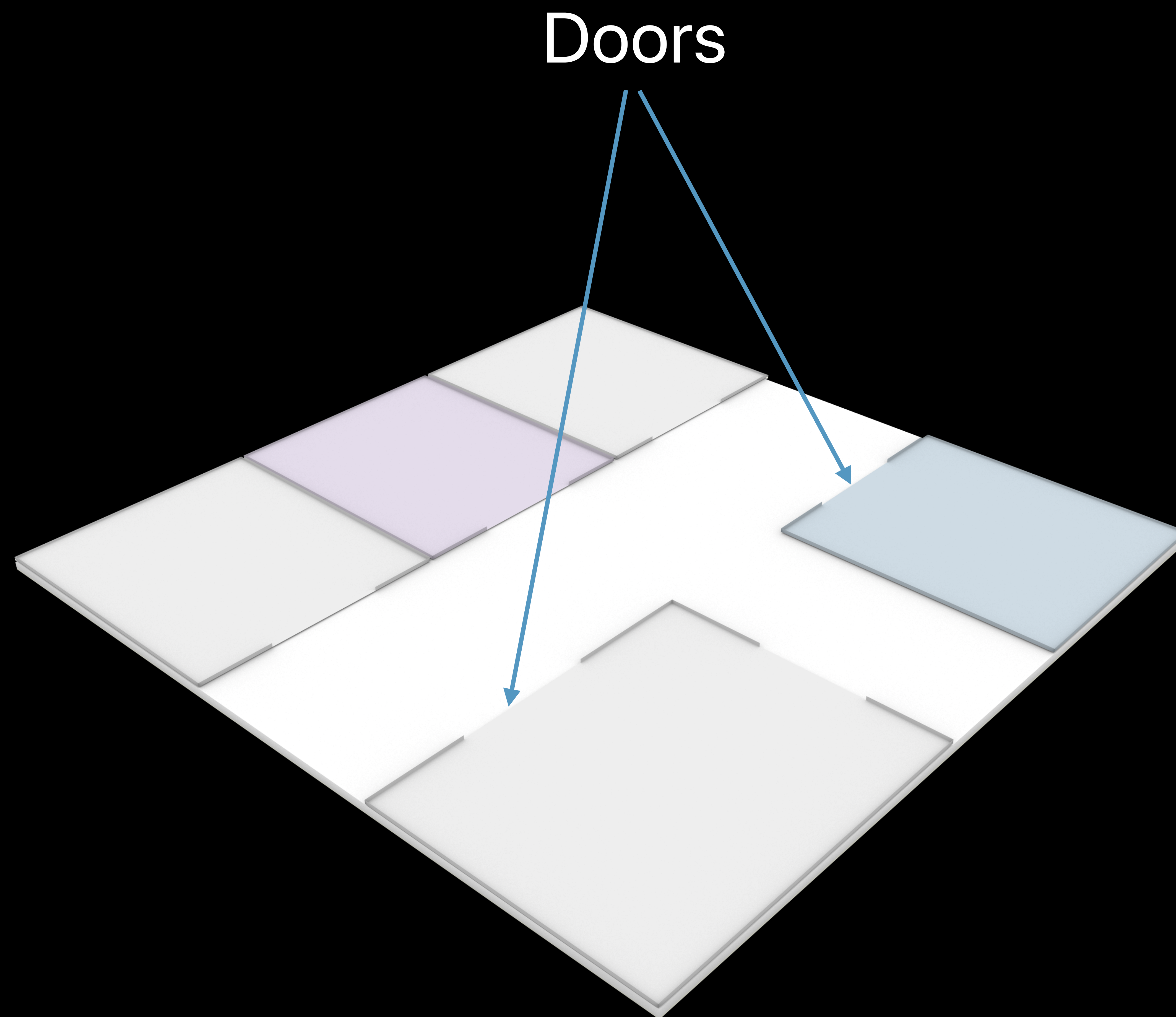
Level

# Indoor Map



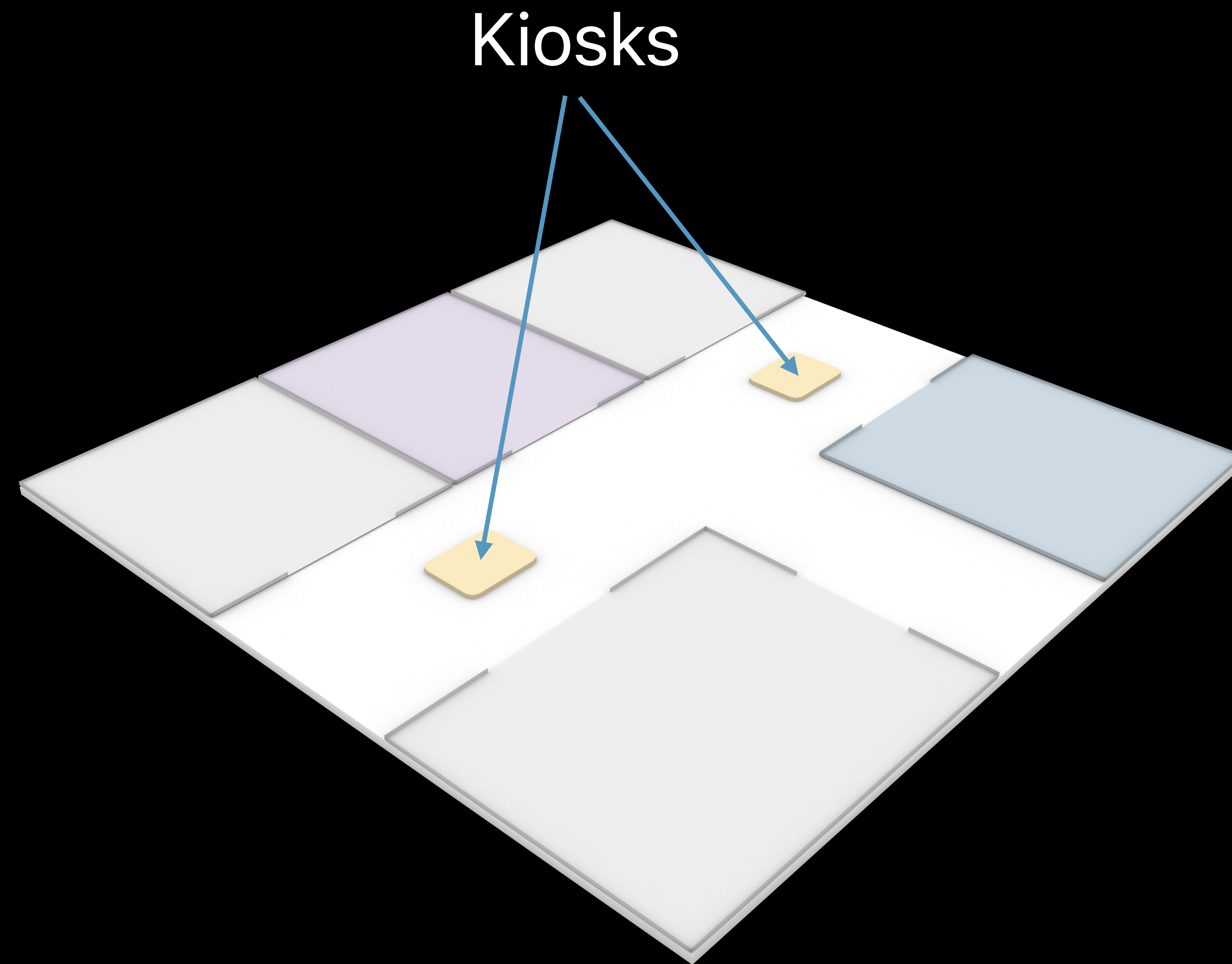


# Indoor Map





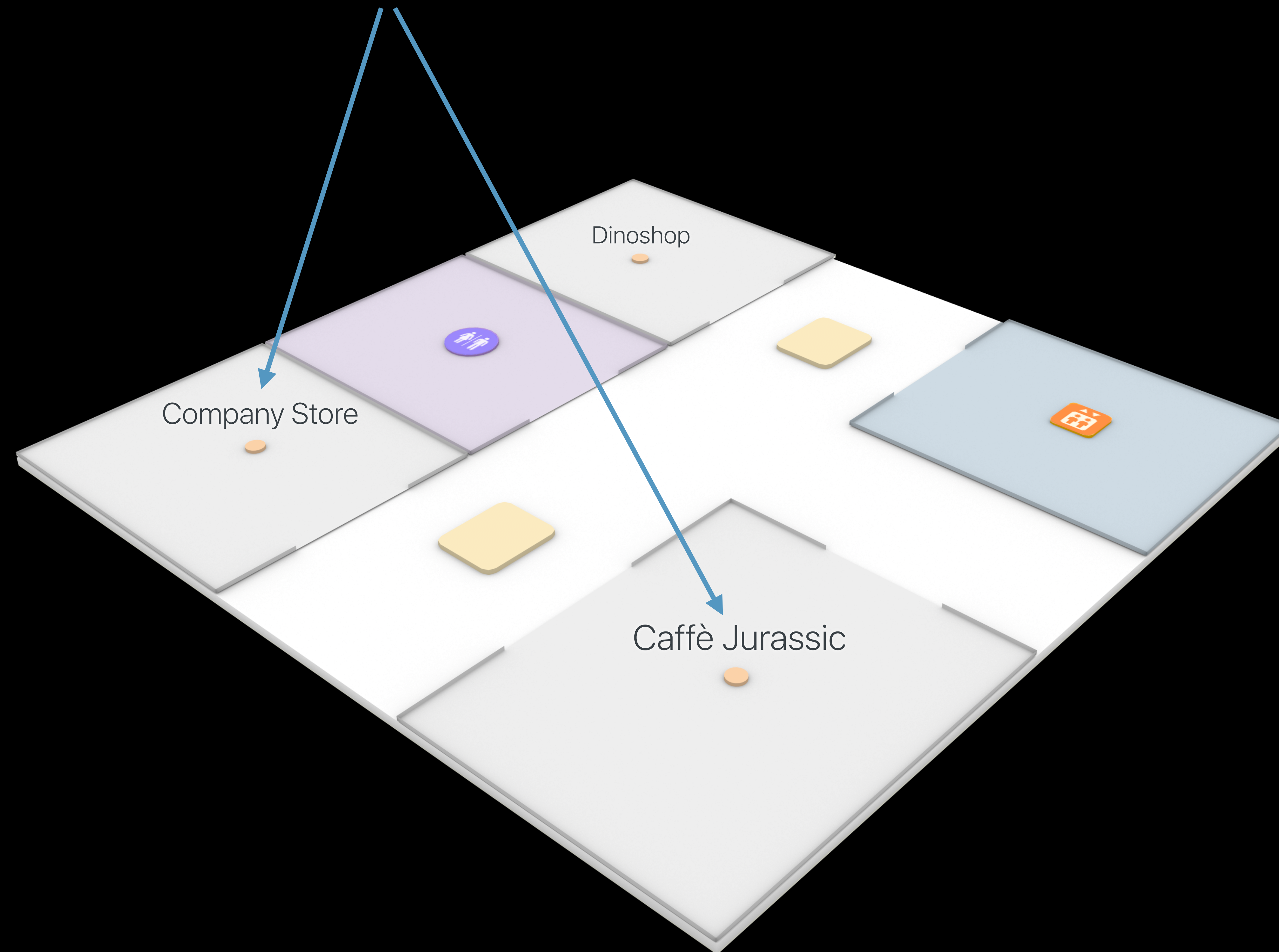
# Indoor Map





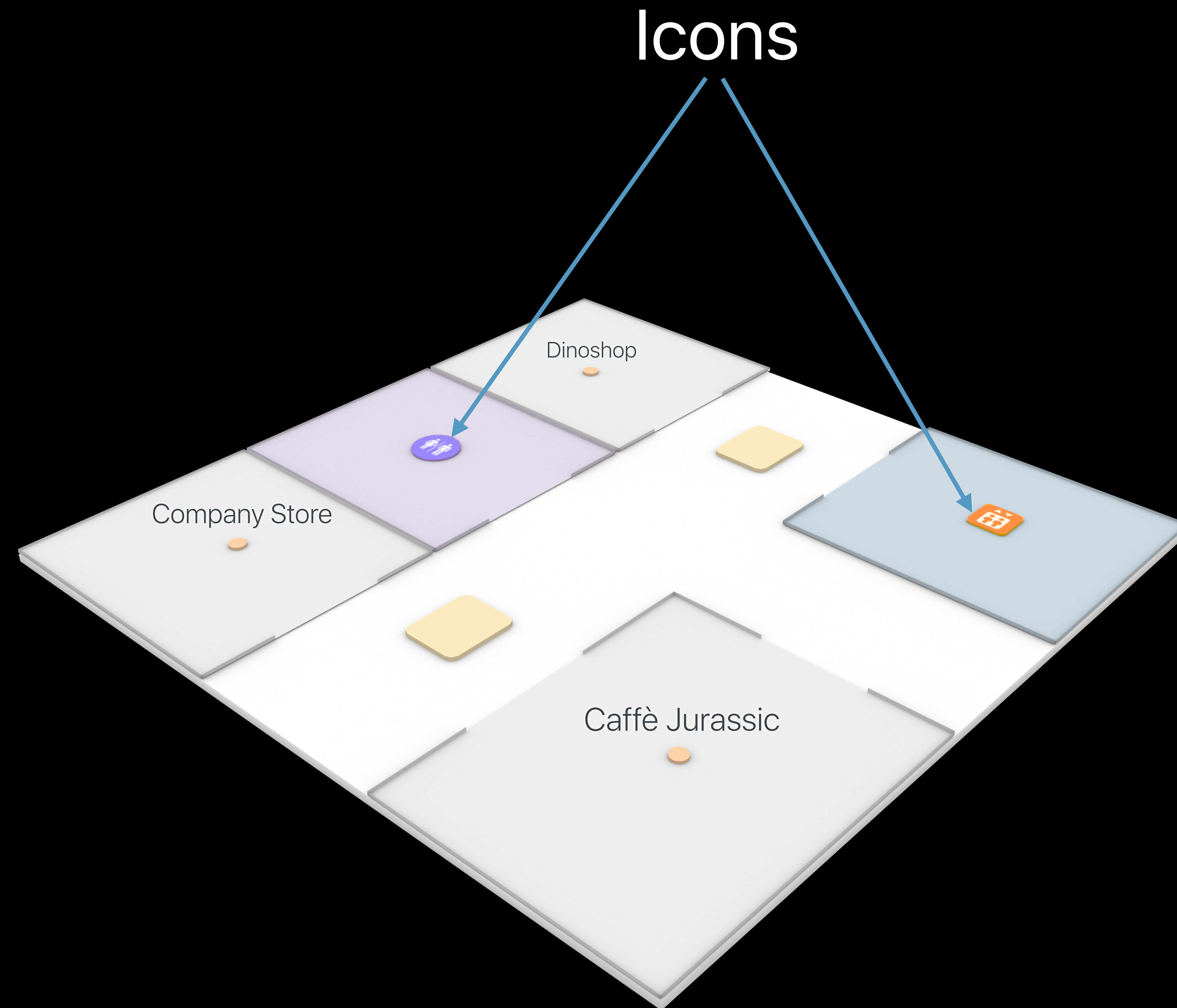
# Indoor Map

Labels



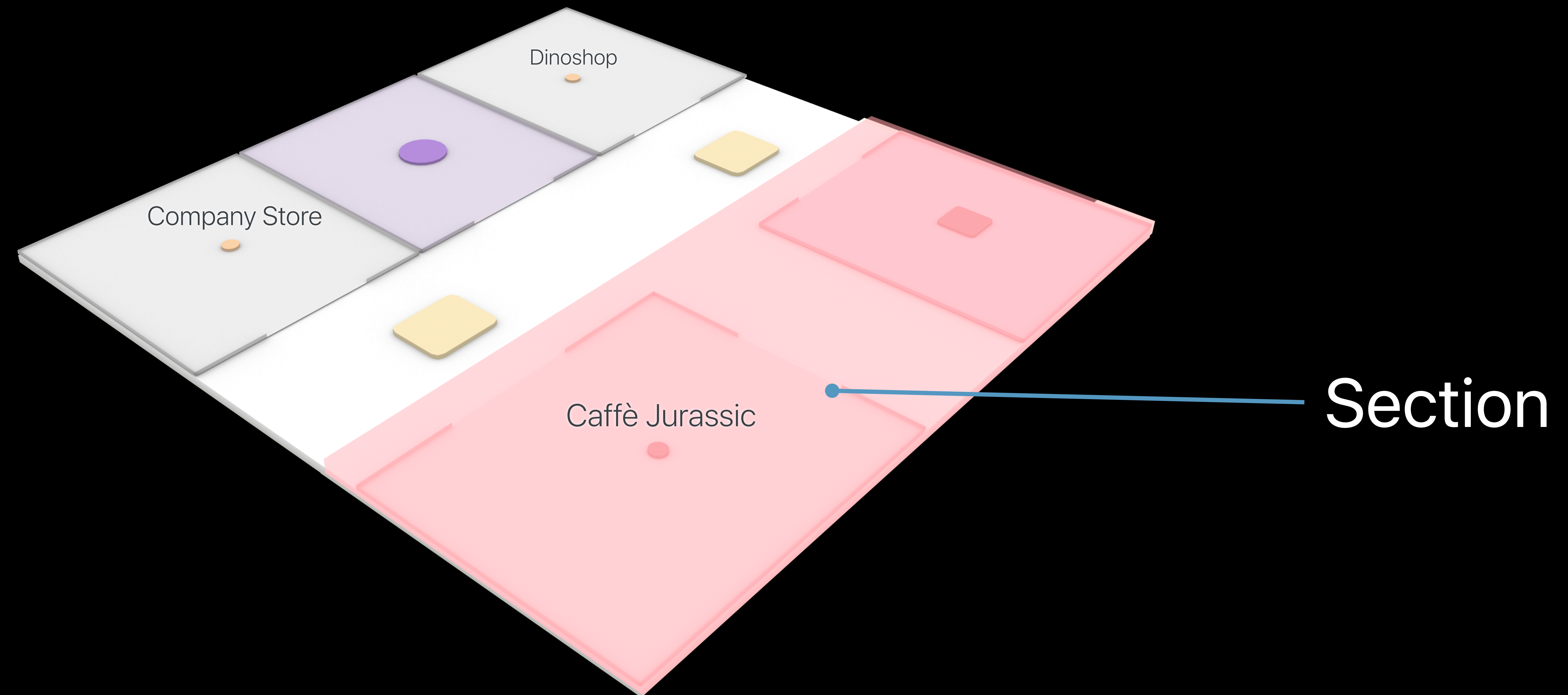


# Indoor Map





# Indoor Map

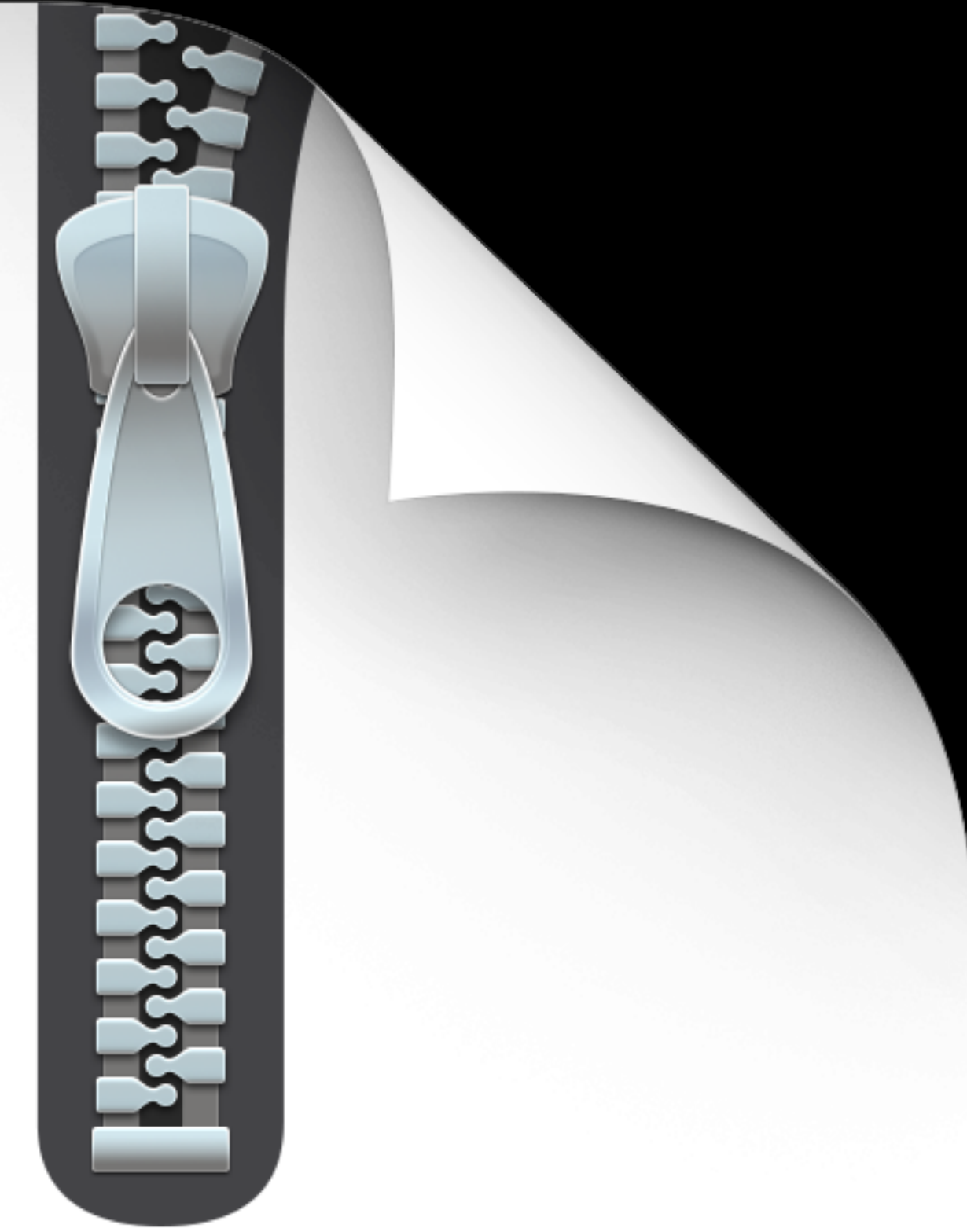




# Indoor Data Mapping Format (IMDF)

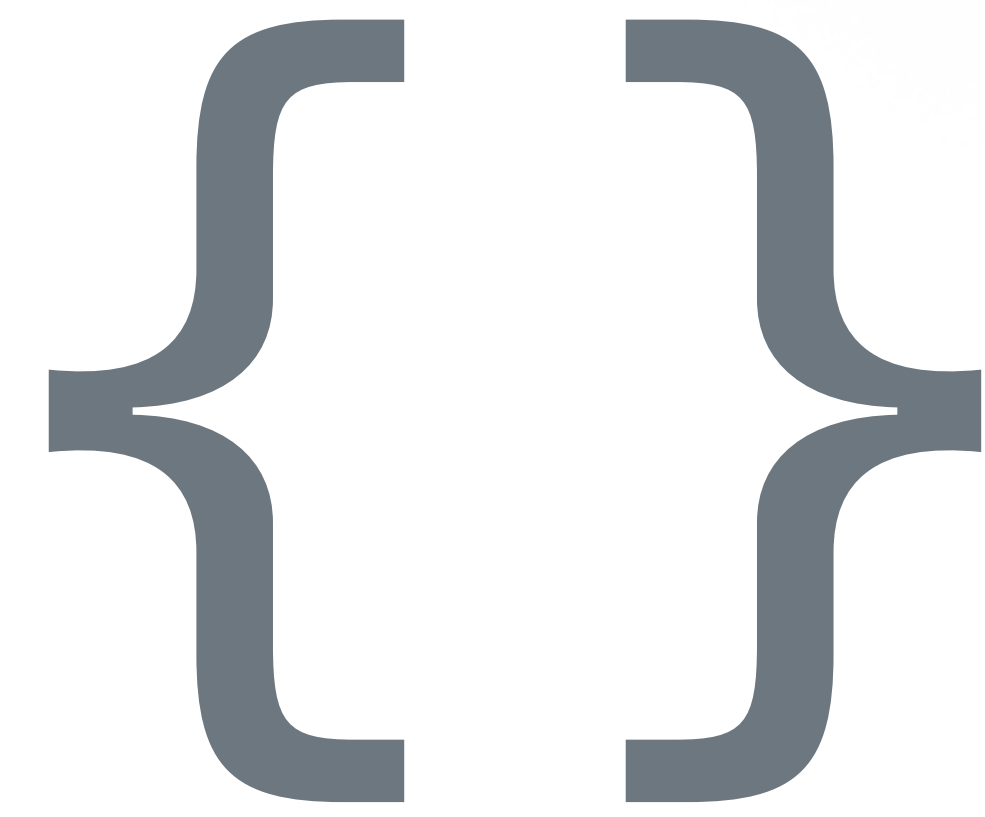
Data model for indoor spaces

2D format based on GeoJSON (RFC-7946)



**IMDF**





**manifest.json**

**address.geojson**

**footprint.geojson**

**anchor.geojson**

**level.geojson**

**building.geojson**

**opening.geojson**

**detail.geojson**

**unit.geojson**

**fixture.geojson**

**venue.geojson**

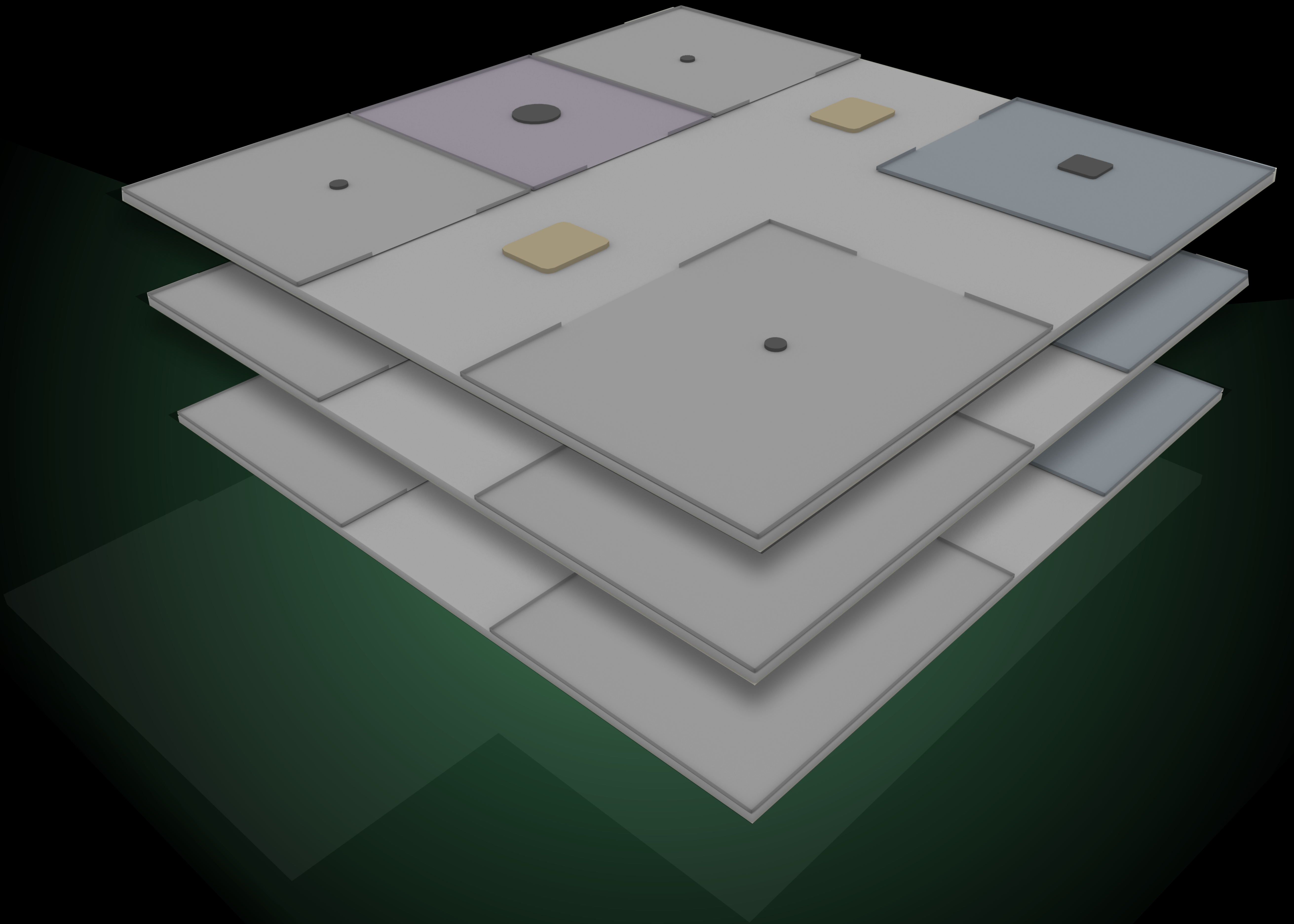




# IMDF

## Level

```
"feature_type": "level",  
"geometry": { "type": "Polygon", ... }  
"properties": {  
  "category": "parking",  
  "ordinal": -1,  
  "name": { "en": "Parking Level 1" },  
  "short_name": { "en": "P1" },  
  "building_ids": []  
  ...  
}
```

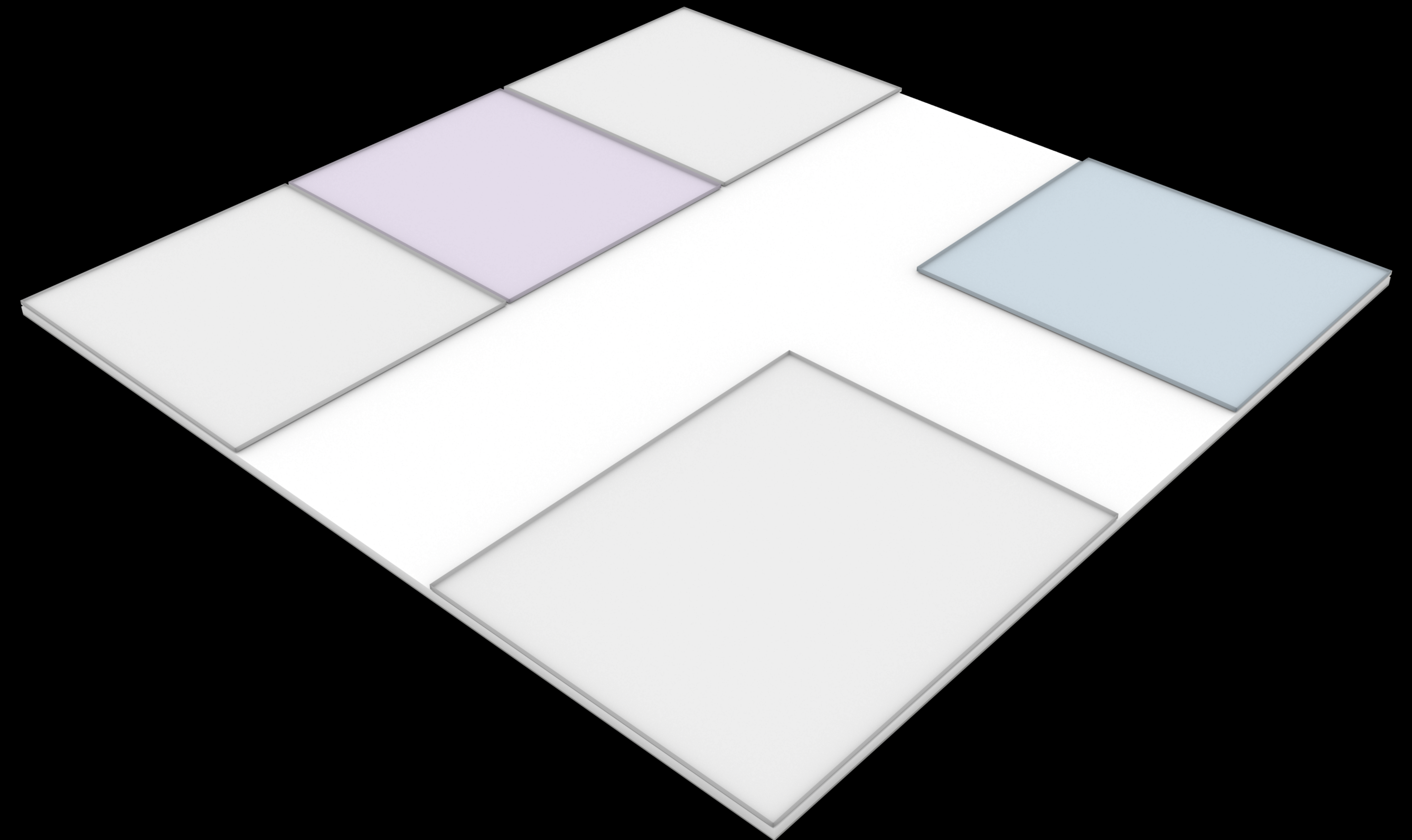




# IMDF

## Unit

```
"feature_type": "unit",  
"geometry": { "type": "Polygon", ... }  
"properties": {  
  "category": "room",  
  "level_id": "...",  
  "name": { "en": "Storage Room 2" },  
  "short_name": { "en": "SR2" }  
  ...  
}
```

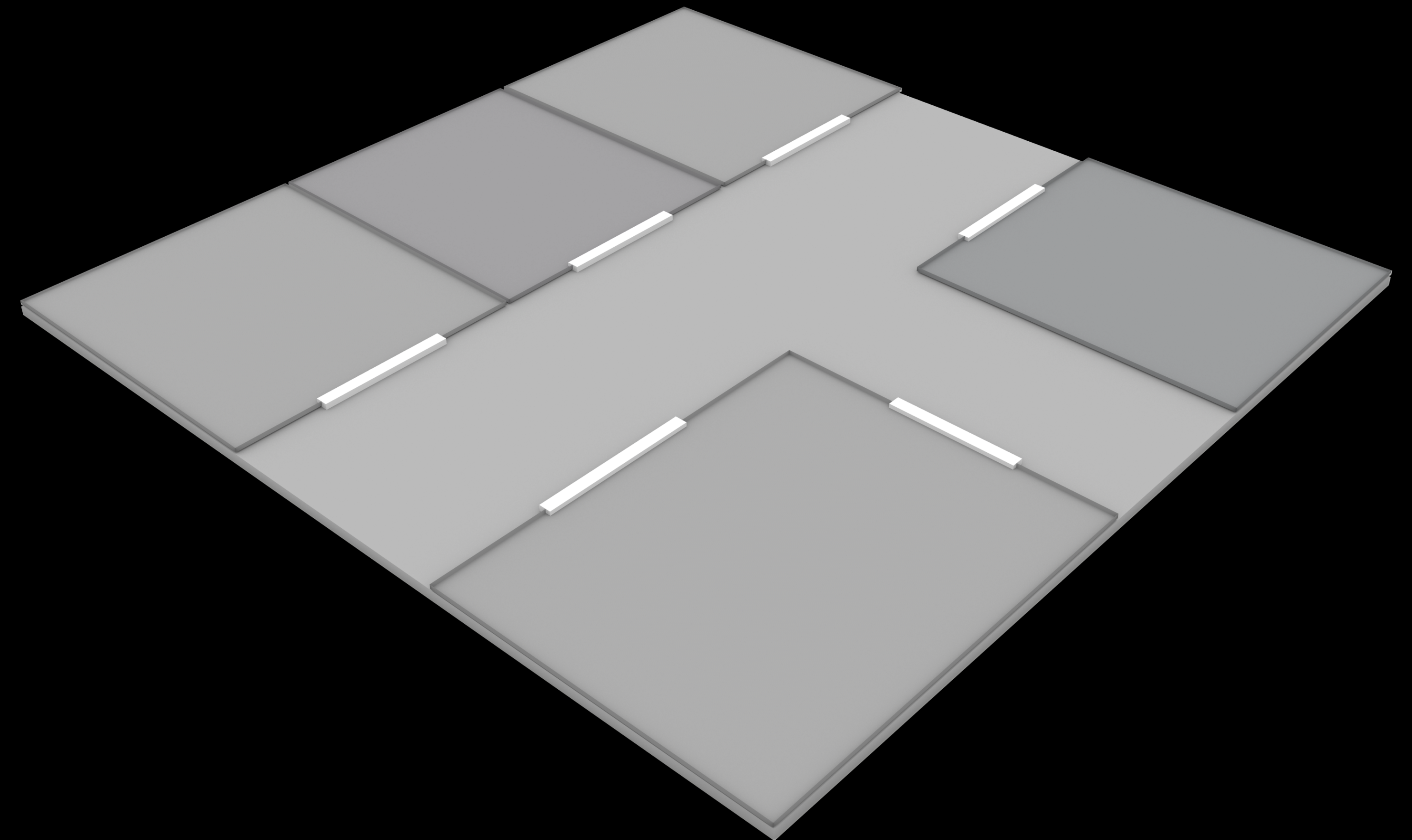




# IMDF

## Opening

```
"feature_type": "opening",  
"geometry": {  
  "type": "LineString", ...  
}  
"properties": {  
  "category": "emergencyexit",  
  "level_id": "...",  
  "accessibility": "wheelchair",  
  "access_control": "badgereader",  
  ...  
  "door": {"automatic": false },  
}
```

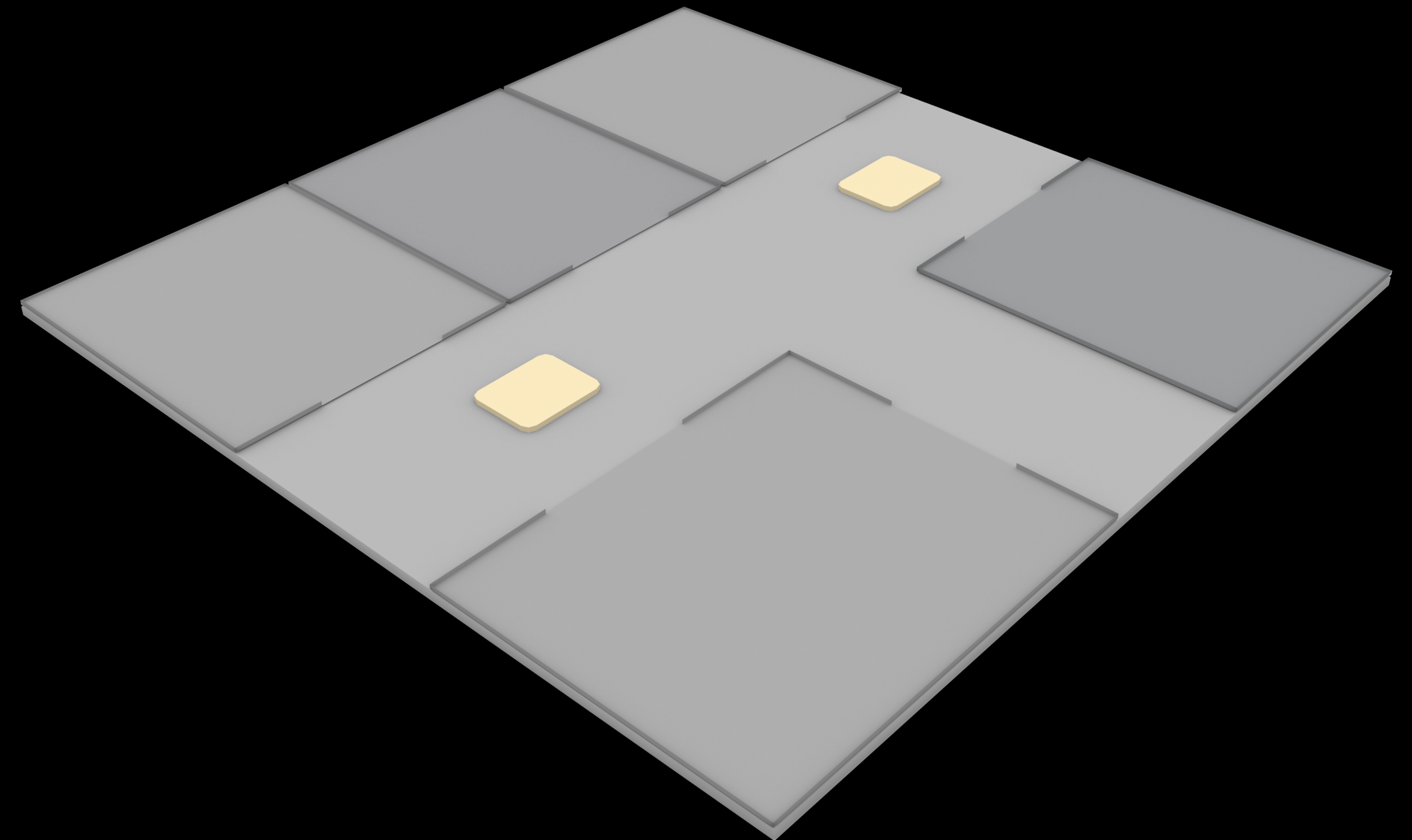




# IMDF

## Kiosk

```
"feature_type": "kiosk",  
"geometry": {  
  "type": "Polygon", ...  
}  
"properties": {  
  "level_id": "...",  
  "anchor_id": "..."  
  ...  
}
```





# IMDF

## Occupant

```
"feature_type": "occupant",  
"geometry": null  
"properties": {  
  "category": "localservices",  
  "anchor_id": "...",  
  "name": { "en": "Caffè Jurassic" },  
  "hours": "Mo-Fr 10:00-21:30",  
  "phone": "+14085512150",  
  "website": "https://apple.com"  
}
```

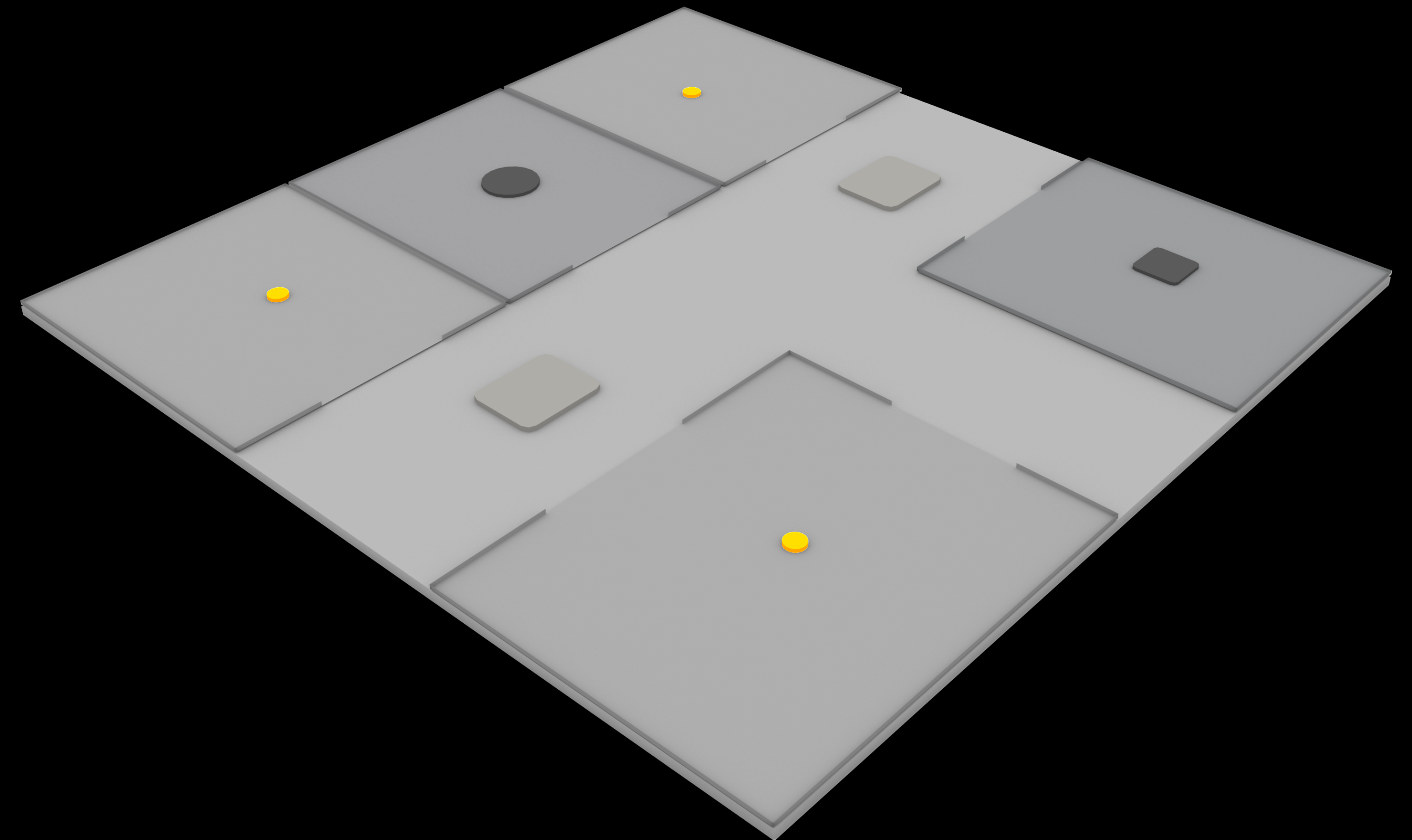




# IMDF

## Anchor

```
"feature_type": "anchor",  
"geometry": {  
  "type": "Point", ...  
}  
"properties": {  
  "address_id": "...",  
  "unit_id": "..."  
}
```

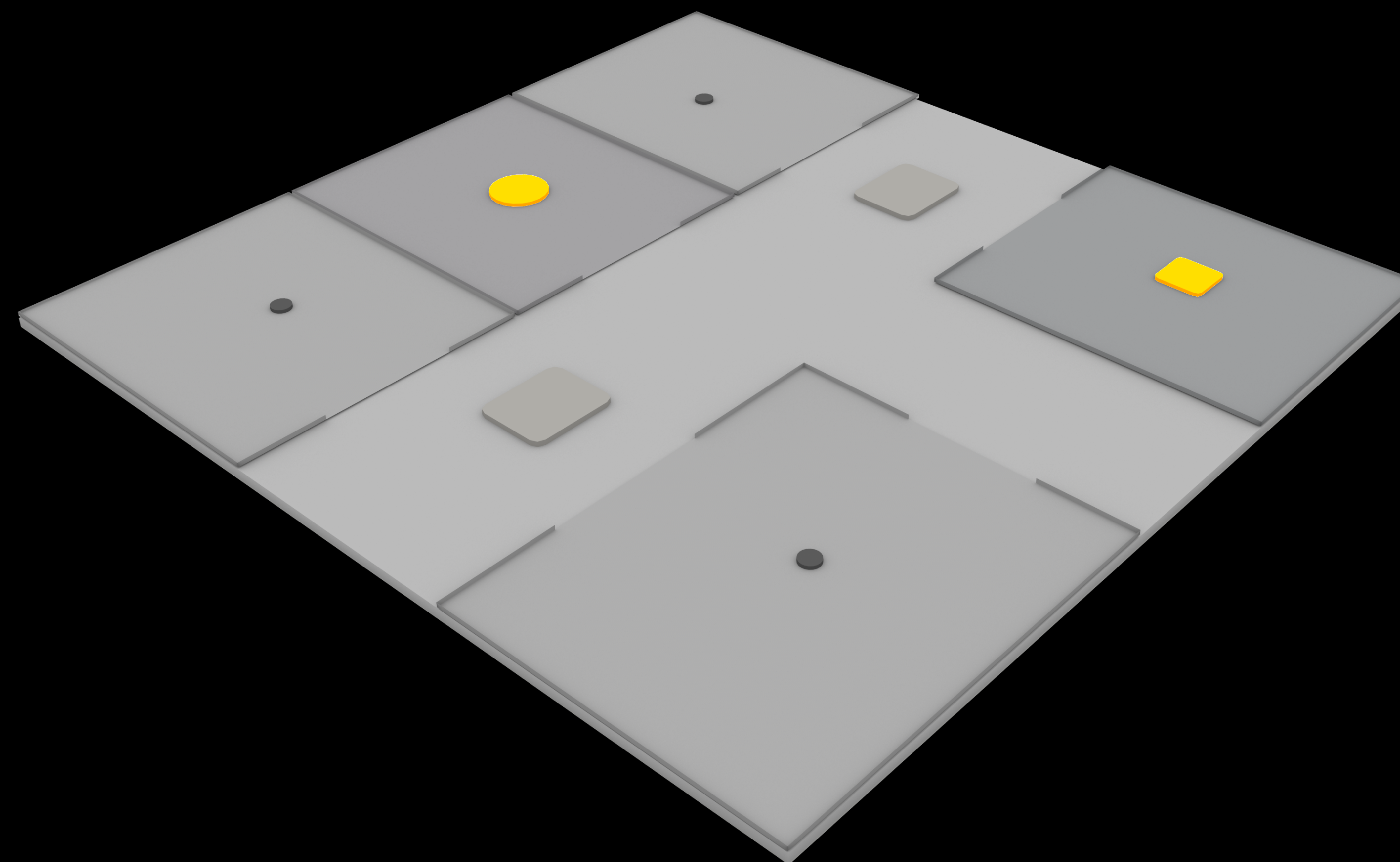




# IMDF

## Amenity

```
"feature_type": "amenity",  
"geometry": {  
  "type": "Point", ...  
}  
"properties": {  
  "category": "exhibit",  
  "unit_ids": ["..."],  
  "name": { "en": "Exhibit Hall" },  
  "hours": "Tu-Su 09:30-17:15",  
  "accessibility": "wheelchair"  
}
```

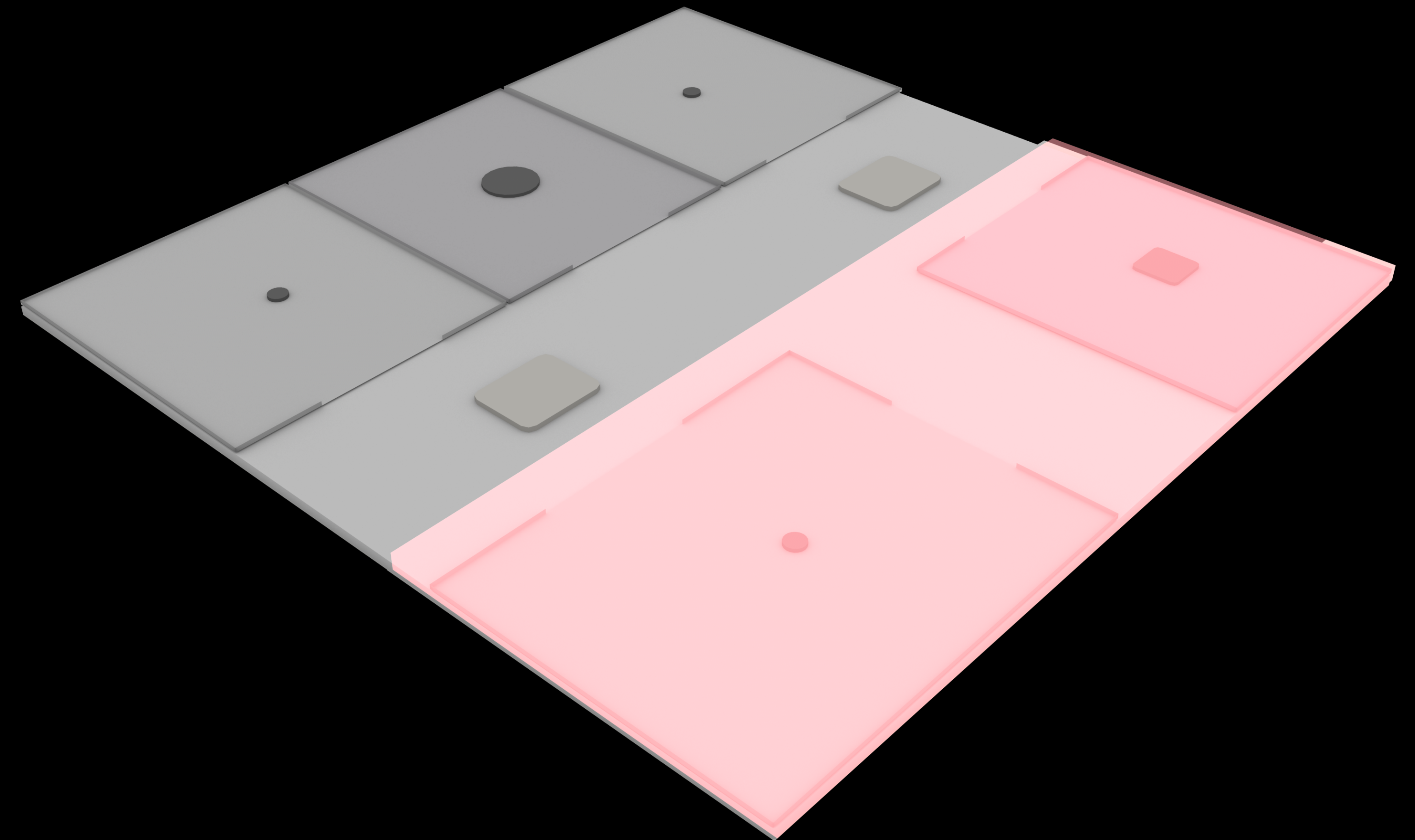




# IMDF

## Section

```
"feature_type": "section",  
"geometry": {  
  "type": "Polygon", ...  
}  
"properties": {  
  "category": "postsecurity",  
  "level_id": "...",  
  "name": { "en": "Post Security 0" },  
  "accessibility": "wheelchair"  
}
```





# IMDF

## Building

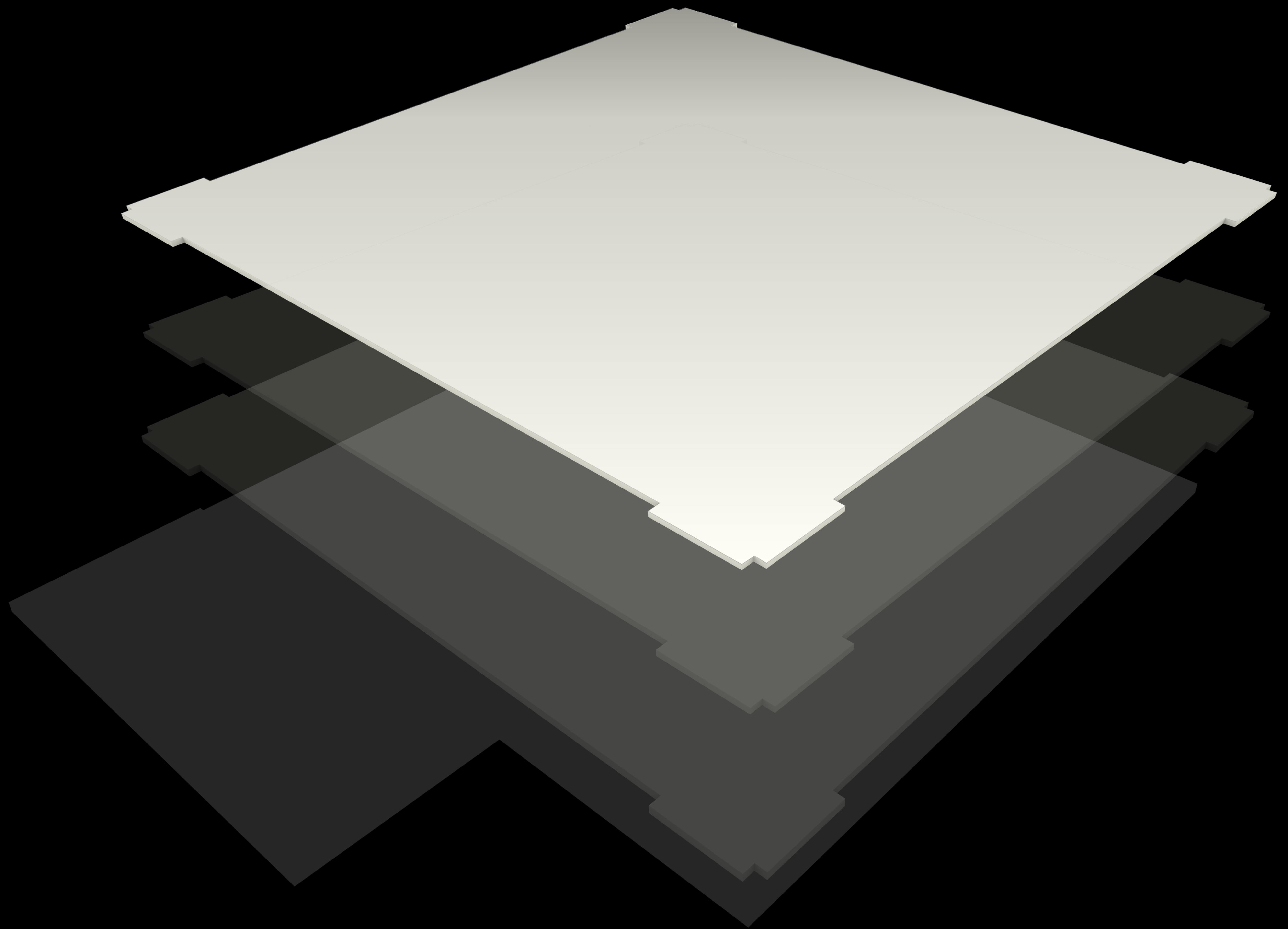
```
"feature_type": "building",  
"geometry": null  
"properties": {  
  "category": "parking",  
  "address_id": "...",  
  "name": { "en": "Parking A" },  
  "accessibility": "wheelchair"  
}
```



# IMDF

## Footprint

```
"feature_type": "footprint",  
"geometry": {  
  "type": "Polygon", ...  
}  
"properties": {  
  "category": "aerial",  
  "building_ids": [...]  
}
```

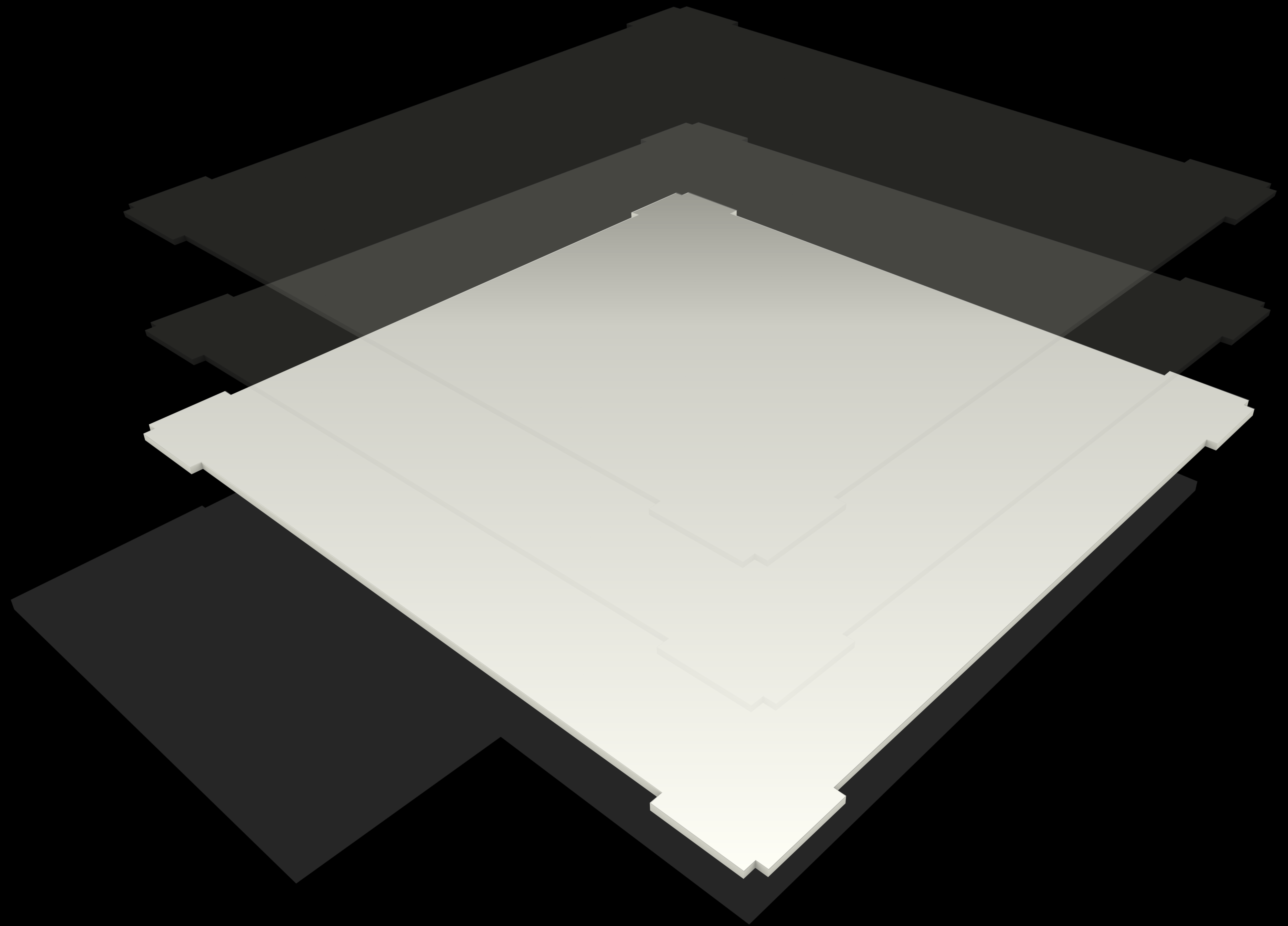




# IMDF

## Footprint

```
"feature_type": "footprint",  
"geometry": {  
  "type": "Polygon", ...  
}  
"properties": {  
  "category": "ground",  
  "building_ids": [...]  
}
```

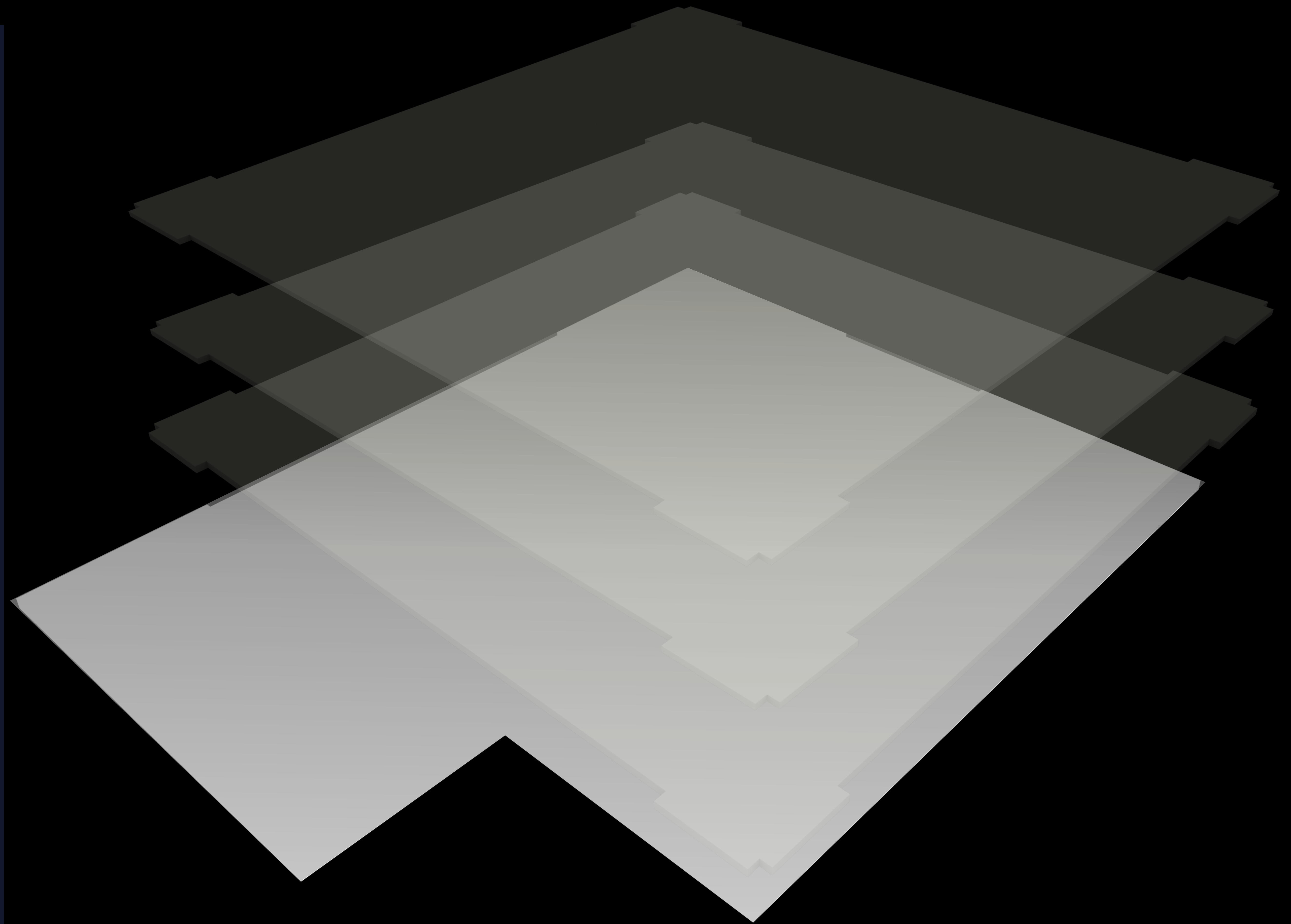




# IMDF

## Footprint

```
"feature_type": "footprint",  
"geometry": {  
  "type": "Polygon", ...  
}  
"properties": {  
  "category": "subterranean",  
  "building_ids": [...]  
}
```





register.apple.com/resources/imdf

Indoor Mapping Data Format (1.0.0.beta.2)

Search docs

Home

- Indoor Mapping Data Format
- Overview
- Specification

License

Reference

Manifest

Feature Types

- Address
- Amenity
- Anchor
- Building
- Detail
- Fixture
- Footprint
- Geofence

Documentation > Home

# Indoor Mapping Data Format

- Version: [1.0.0.beta.2](#)
- Last update: March 04, 2019

## Overview

Indoor Mapping Data Format (referenced throughout this document as "IMDF") provides a generalized, yet comprehensive model for any indoor location, providing a basis for orientation, navigation and discovery. In this release there are also detailed instructions for modeling the spaces of an airport and a shopping mall.

Developers can access both text and visual examples of all features, along with clear explanations of all terms. IMDF conforms to [RFC 7946](#), ensuring compatibility and transferability of the data. IMDF is lightweight, mobile friendly, and can be rendered on any device, OS, or browser.

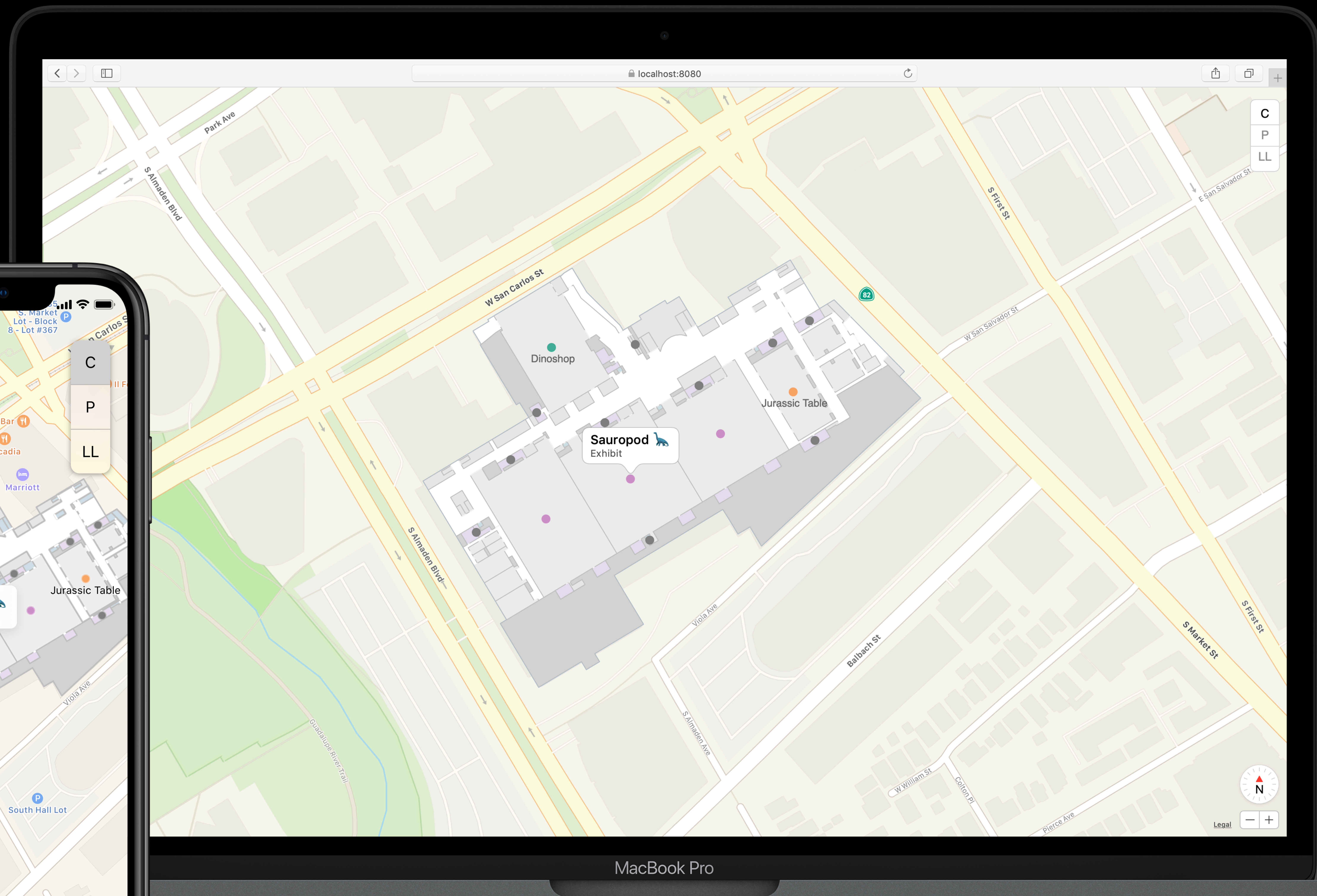
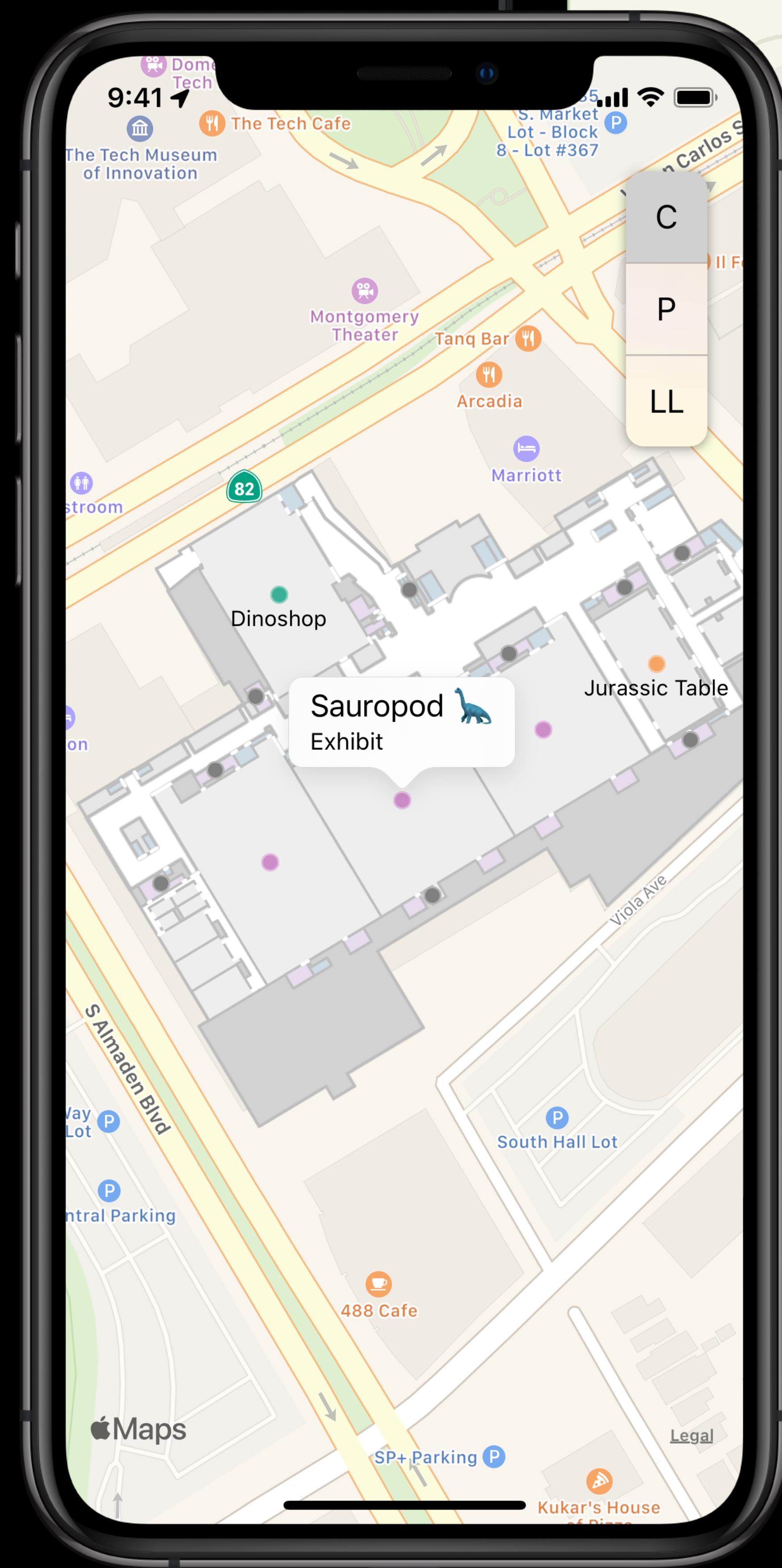
For GIS and BIM specialists, there is support for IMDF in many of your favorite tools.

IMDF is used by Apple to provide [Indoor Positioning support with Core Location](#). Indoor maps integrated with indoor positioning can establish the foundation for a wide range of consumer and enterprise location-based apps and websites.

MacBook Pro

register.apple.com/resources/imdf







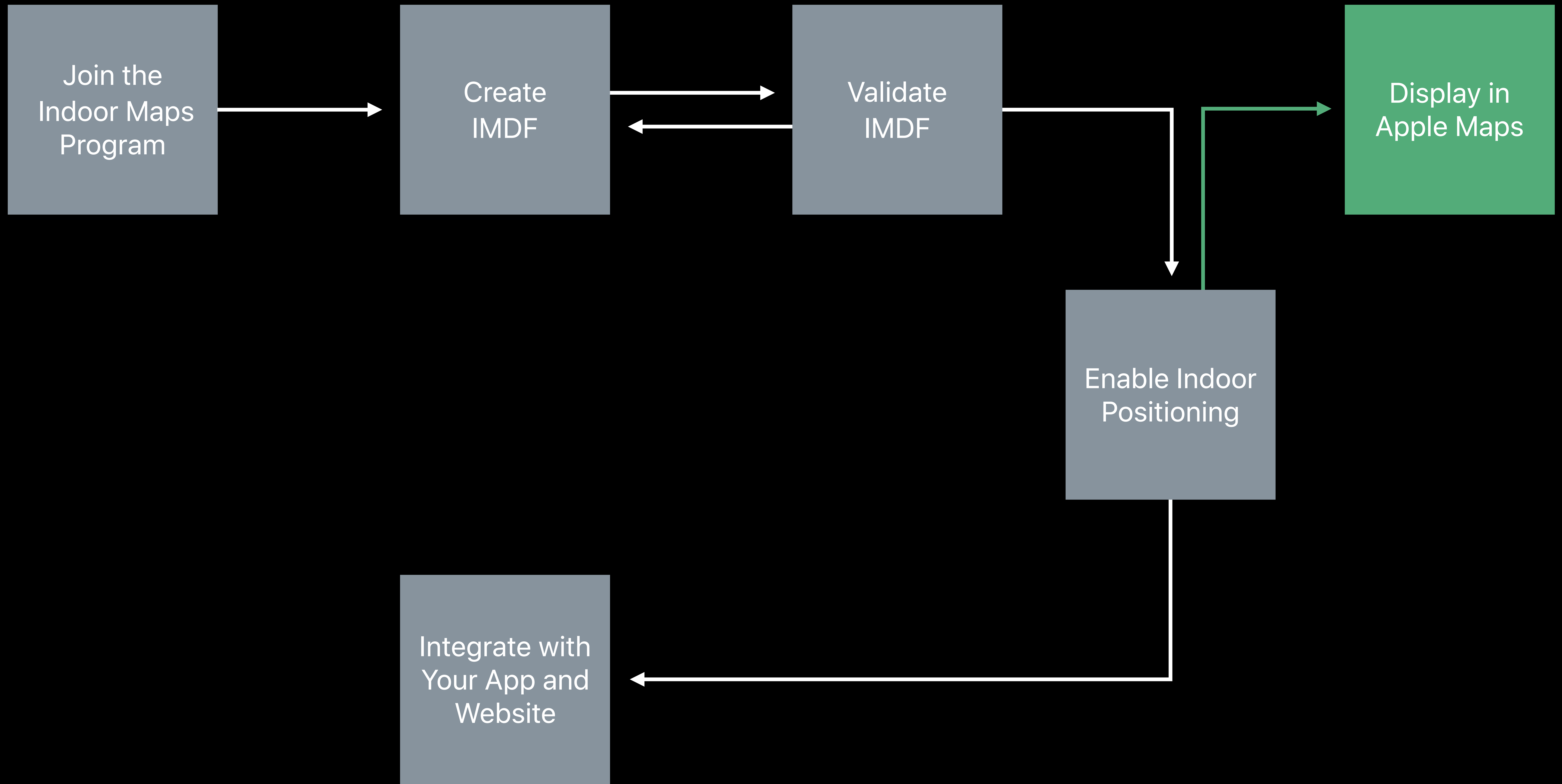
Indoor Mapping Data Format (IMDF)

**Indoor Maps Program**

Displaying Indoor Maps in an iOS App

Displaying Indoor Maps in a Web App





# Indoor Maps

Easily create detailed maps of your indoor spaces and let visitors see where they are right in your app. Organizations with large public and private spaces like airports, shopping centers, arenas, hospitals, universities, and private office buildings can register for the Indoor Maps Program. Indoor maps are built using industry standard tools and require only your existing Wi-Fi network to enable GPS-level location accuracy so visitors can navigate your spaces with ease.

Get Started



## Create standards-based indoor maps

Indoor Mapping Data Format (IMDF) provides a generalized, yet comprehensive data model for any indoor location, creating a basis for orientation, navigation and



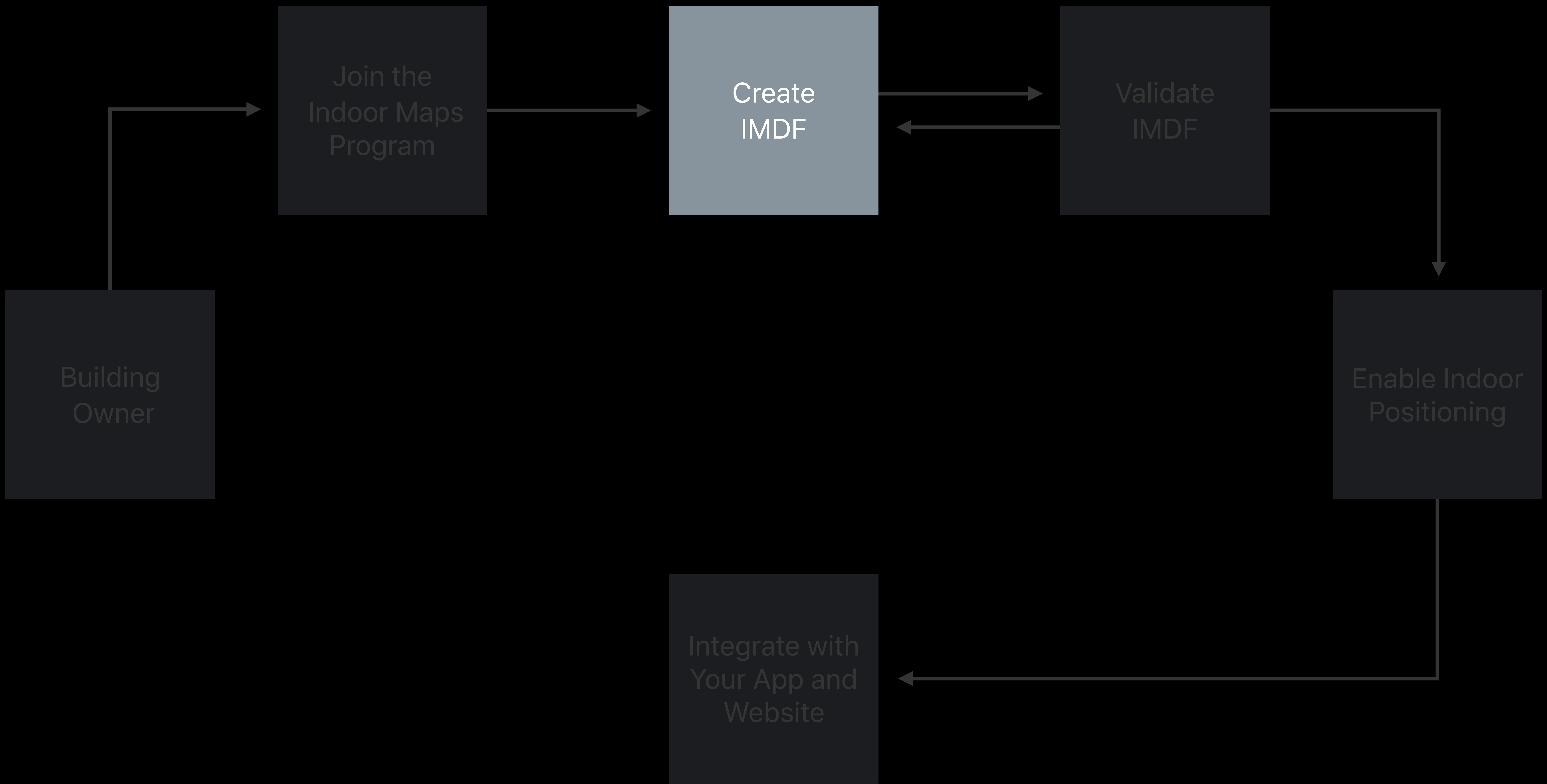
## Enable indoor positioning

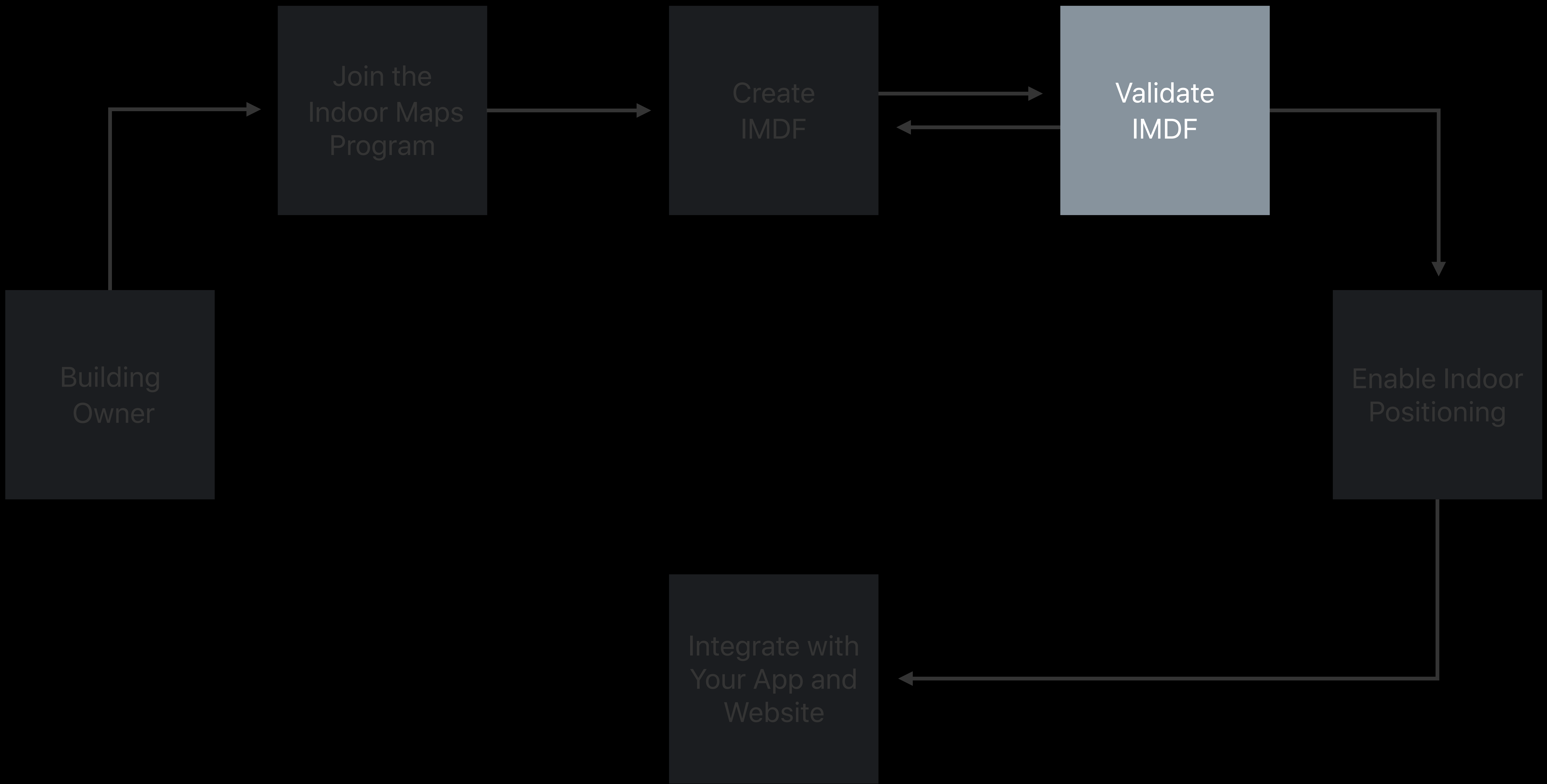
Apple's indoor positioning technology passively uses your existing Wi-Fi infrastructure and requires no additional hardware deployment. This technology achieves GPS-level

MacBook Pro

register.apple.com/indoor









Browser address bar: <https://register.apple.com/indoor/imdf-sandbox>

Page Header: IMDF Sandbox | Victoria International Airport

Page Actions: Open New IMDF | Download IMDF

Left Sidebar:

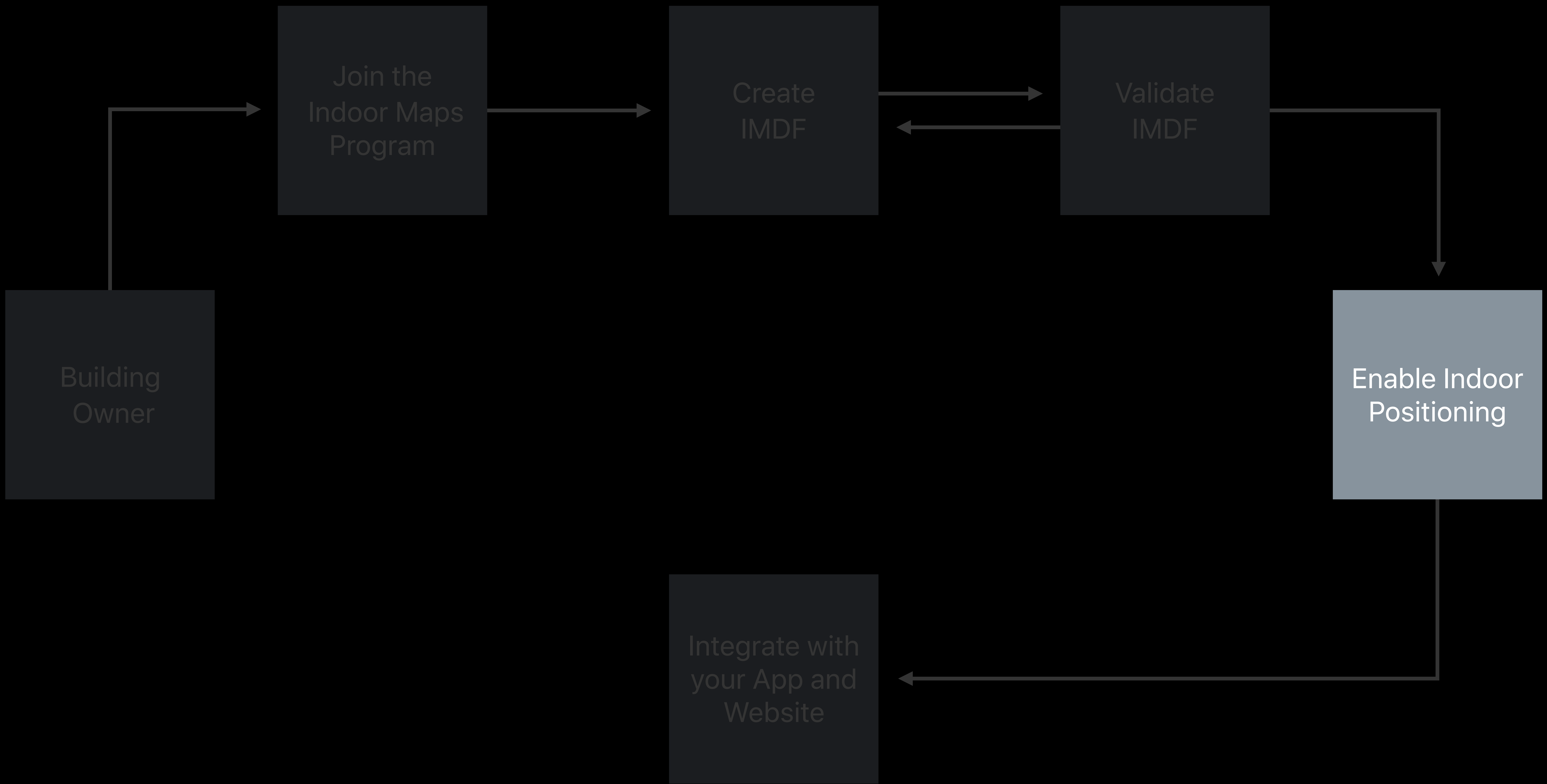
- Navigation: Home, Search
- Location: **Victoria International Airport**
- Warnings: Warnings found 20
- Directory
  - General Information
  - Building Information
  - Businesses 14
  - Services 110
- Levels
  - Level 3
  - Level 2
  - Level 1

Main Map Area:

- Level Selection: L3, L2, L1 (L1 is selected)
- Map Labels: 4, 6, 8, Business Center, Harbour Walk Shops, Automated External Defibrillator, Automated Teller Machine, Yellow Cab, Taxi Information, Pay Phone, Air Canada Jazz C, Fan
- Map Features: Blue route line, North arrow, Zoom in (+) and Zoom out (-) buttons

MacBook Pro







# Indoor Survey

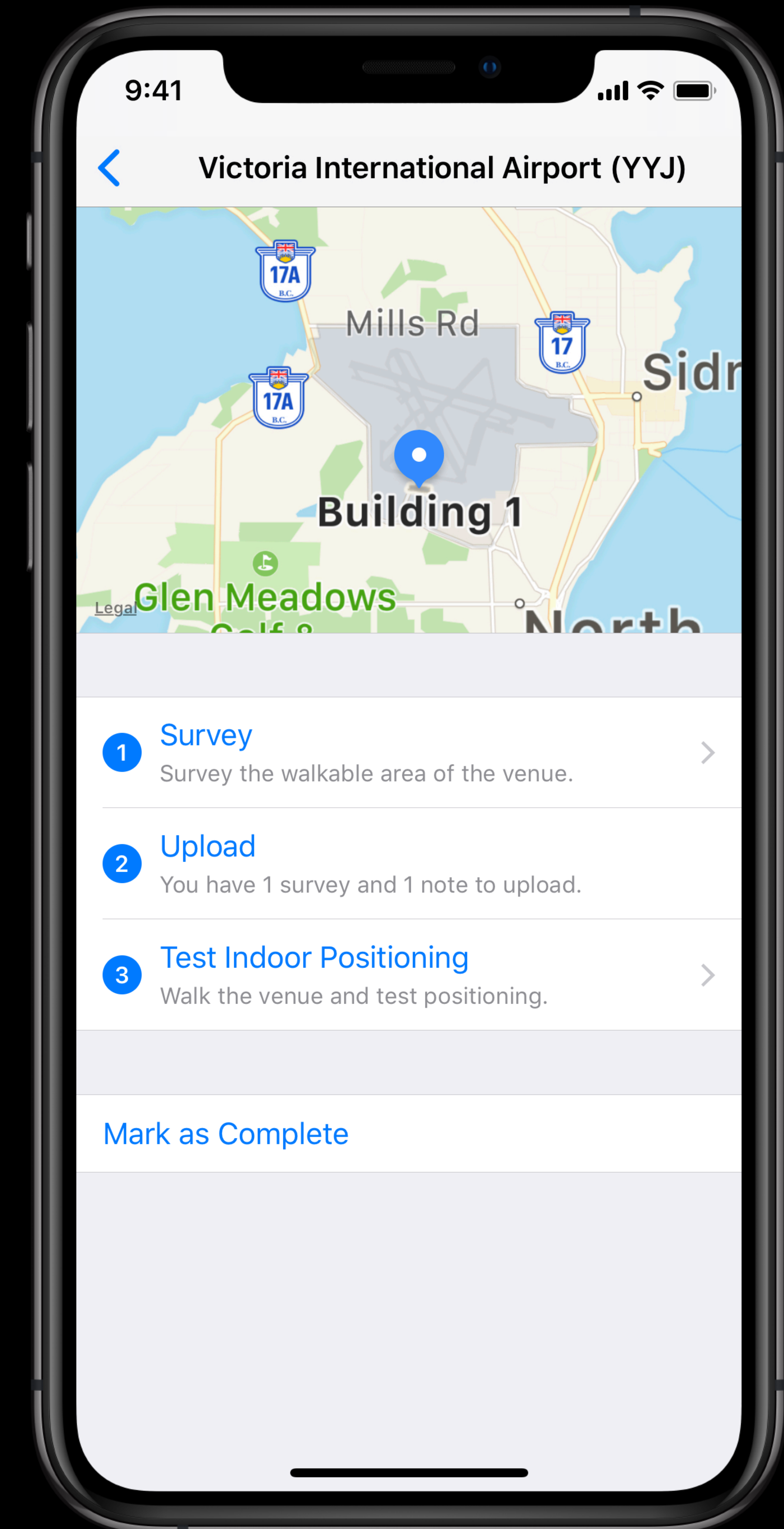
Collect RF fingerprints



# Indoor Survey

Collect RF fingerprints

Upload data for processing



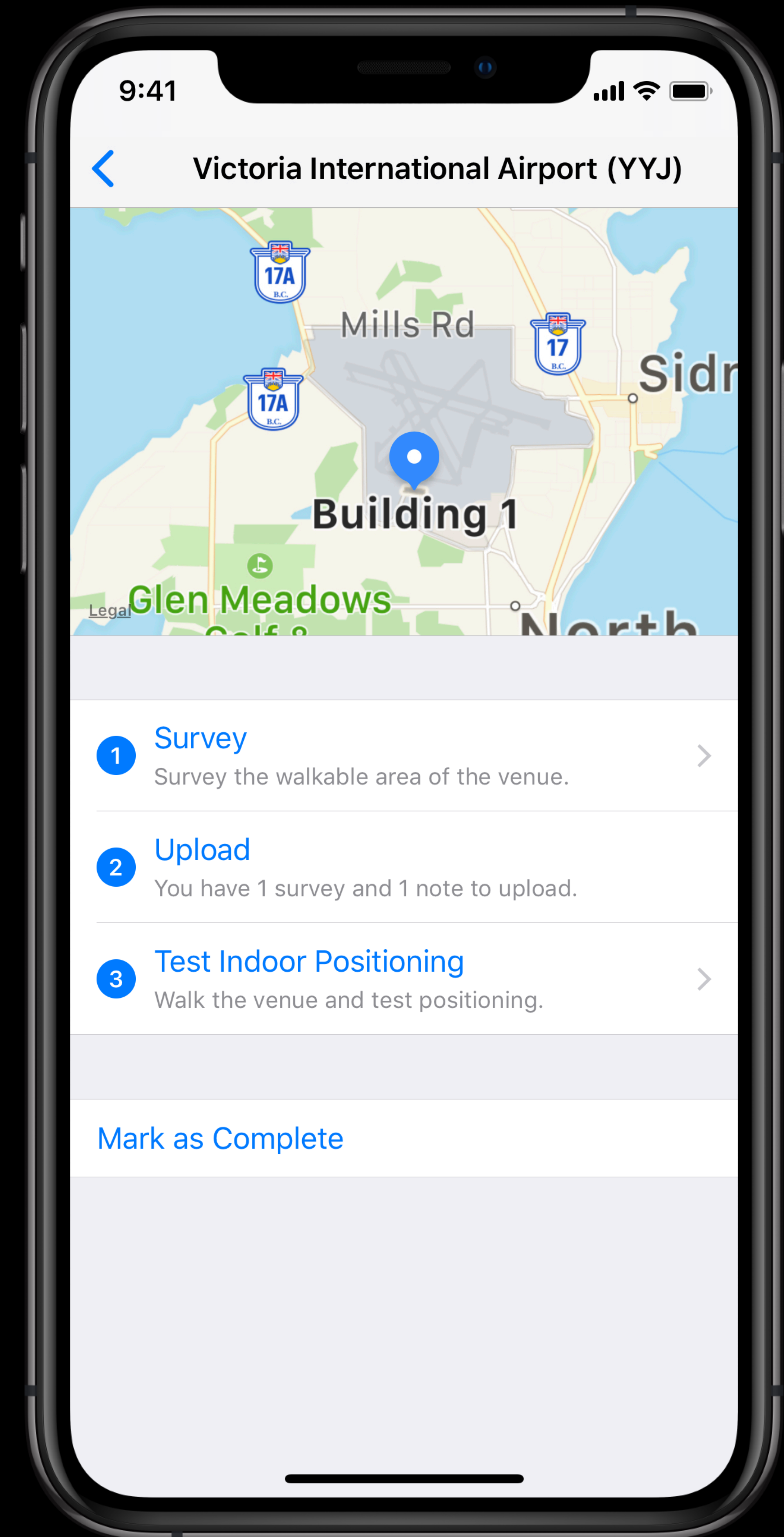


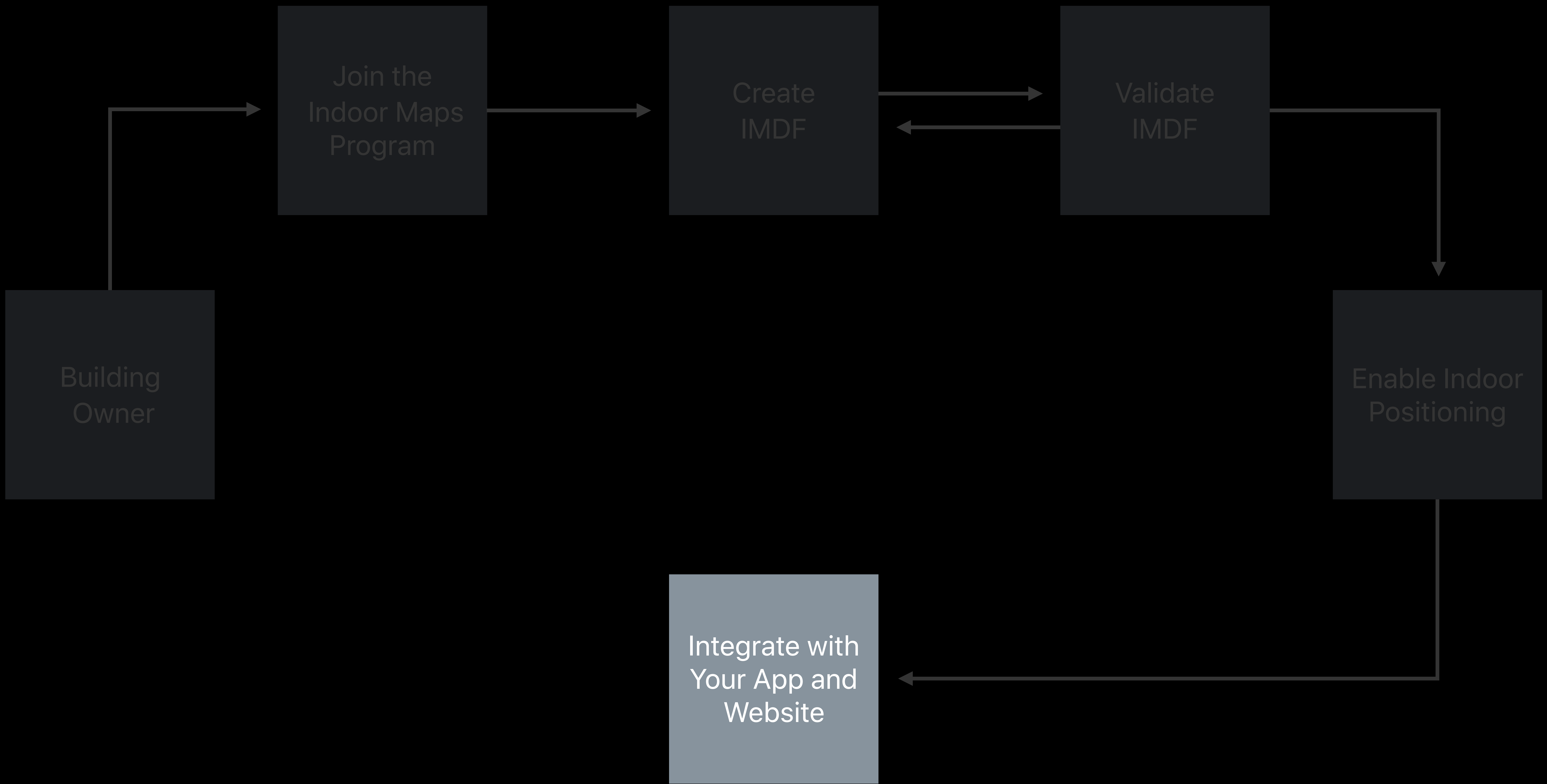
# Indoor Survey

Collect RF fingerprints

Upload data for processing

Test indoor positioning





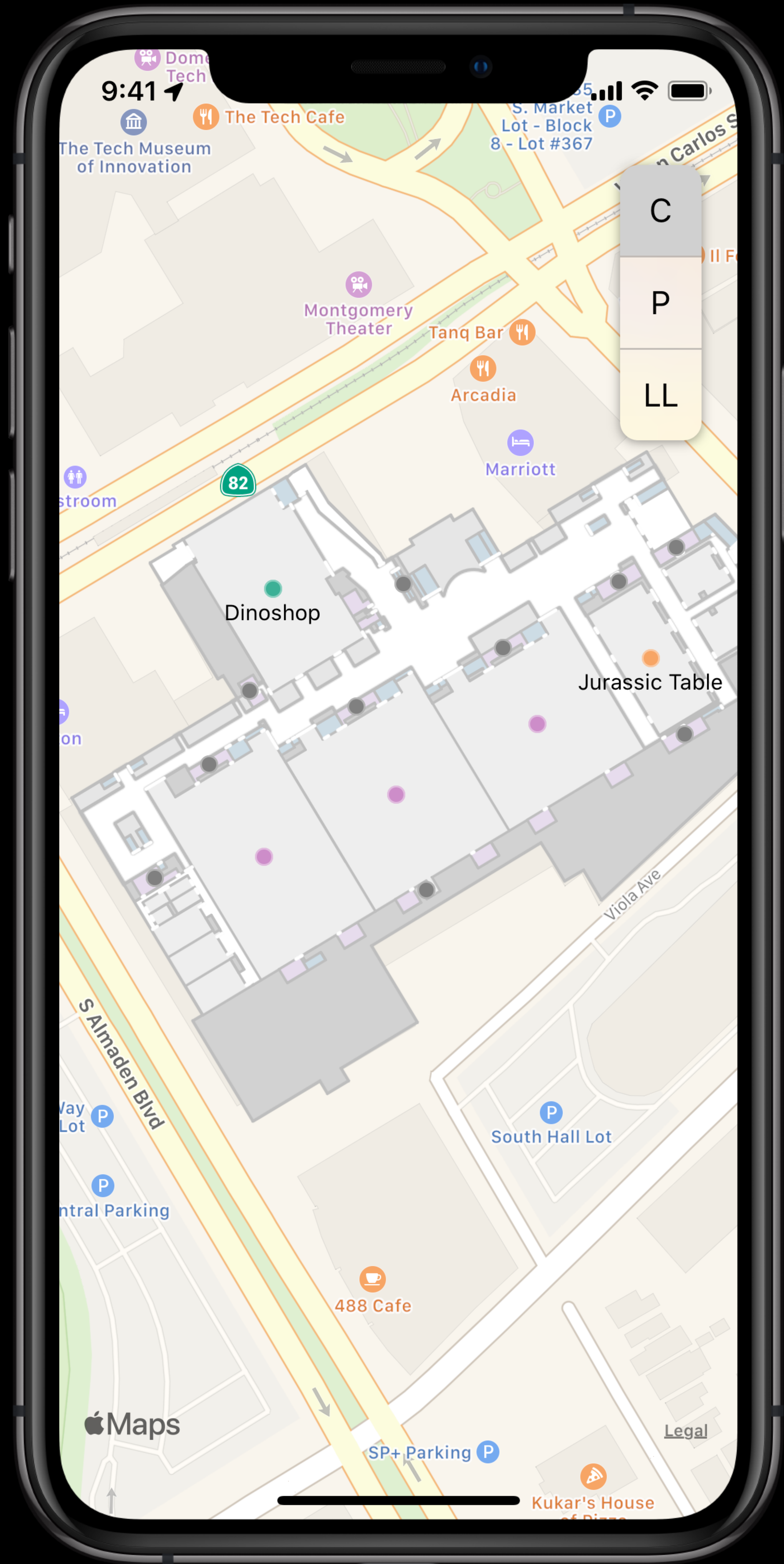


Indoor Mapping Data Format (IMDF)

Indoor Maps Program

Displaying Indoor Maps in an iOS App

Displaying Indoor Maps in a Web App



9:41

The Tech Museum of Innovation

The Tech Cafe

Montgomery Theater

Tanq Bar

Arcadia

Marriott

Dinoshop

Jurassic Table

S Almaden Blvd

South Hall Lot

488 Cafe

Apple Maps

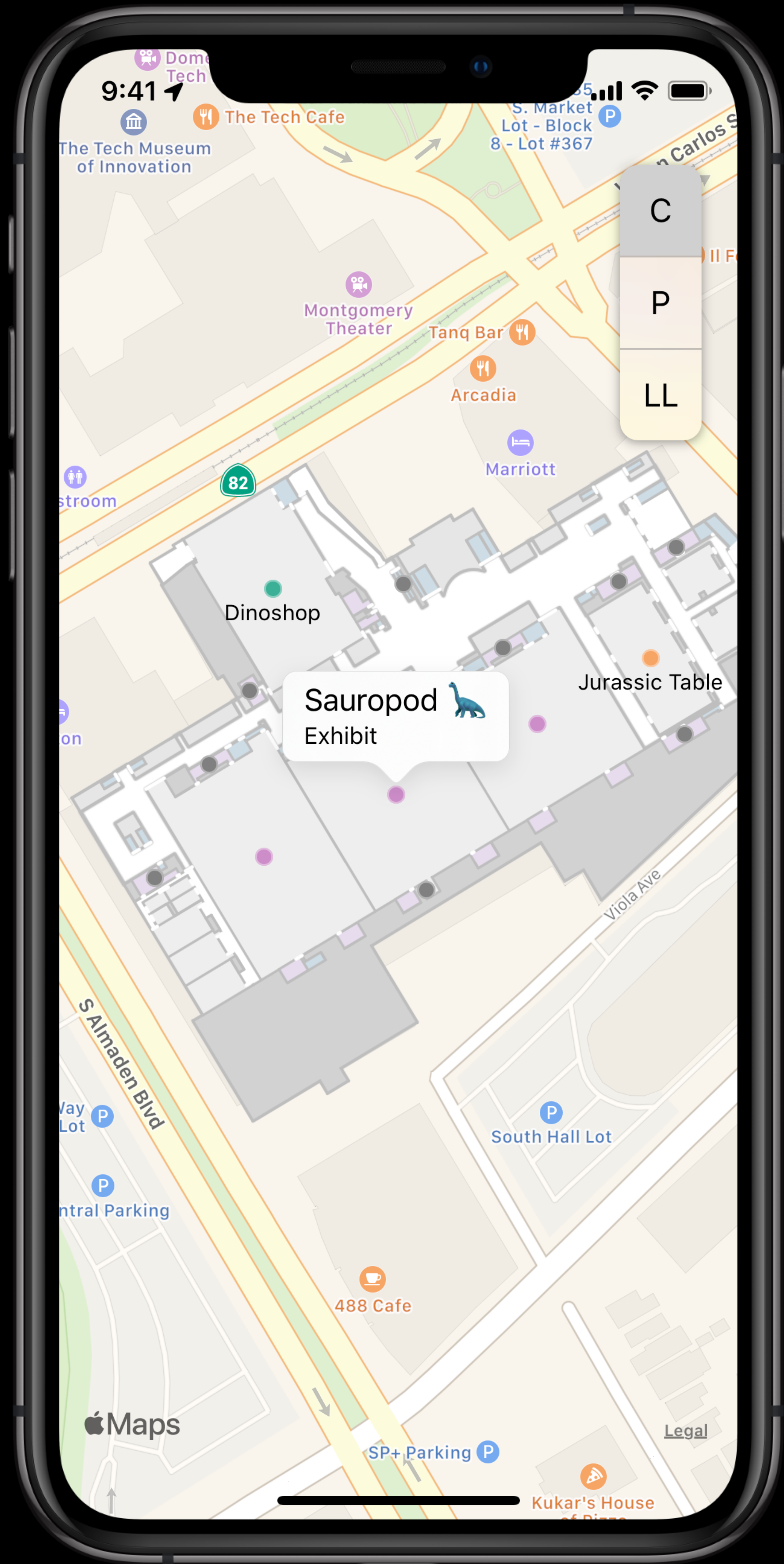
SP+ Parking

Legal

Kukar's House

- C
- P
- LL





9:41

The Tech Museum of Innovation

The Tech Cafe

Montgomery Theater

Tanq Bar

Arcadia

Marriott

Dinoshop

Sauropod Exhibit

Jurassic Table

S Almaden Blvd

South Hall Lot

488 Cafe

Apple Maps

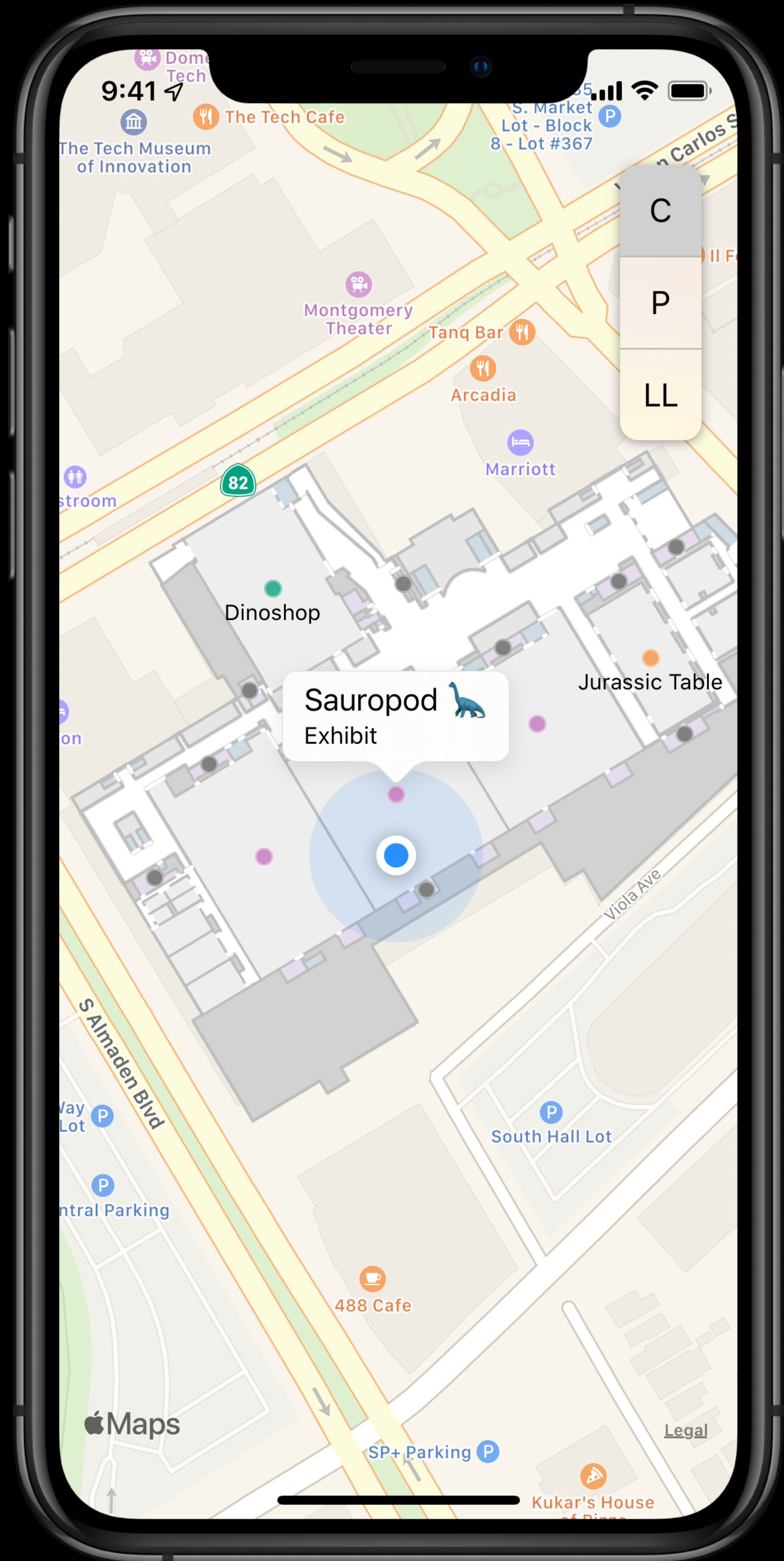
SP+ Parking

Kukar's House

Legal

- C
- P
- LL





9:41

The Tech Museum of Innovation

The Tech Cafe

S. Market Lot - Block 8 - Lot #367

C

P

LL

Montgomery Theater

Tanq Bar

Arcadia

Marriott

82

stroom

Dinoshop

Sauropod Exhibit



Jurassic Table

S Almaden Blvd

Viola Ave

Way Lot

South Hall Lot

entral Parking

488 Cafe

Apple Maps

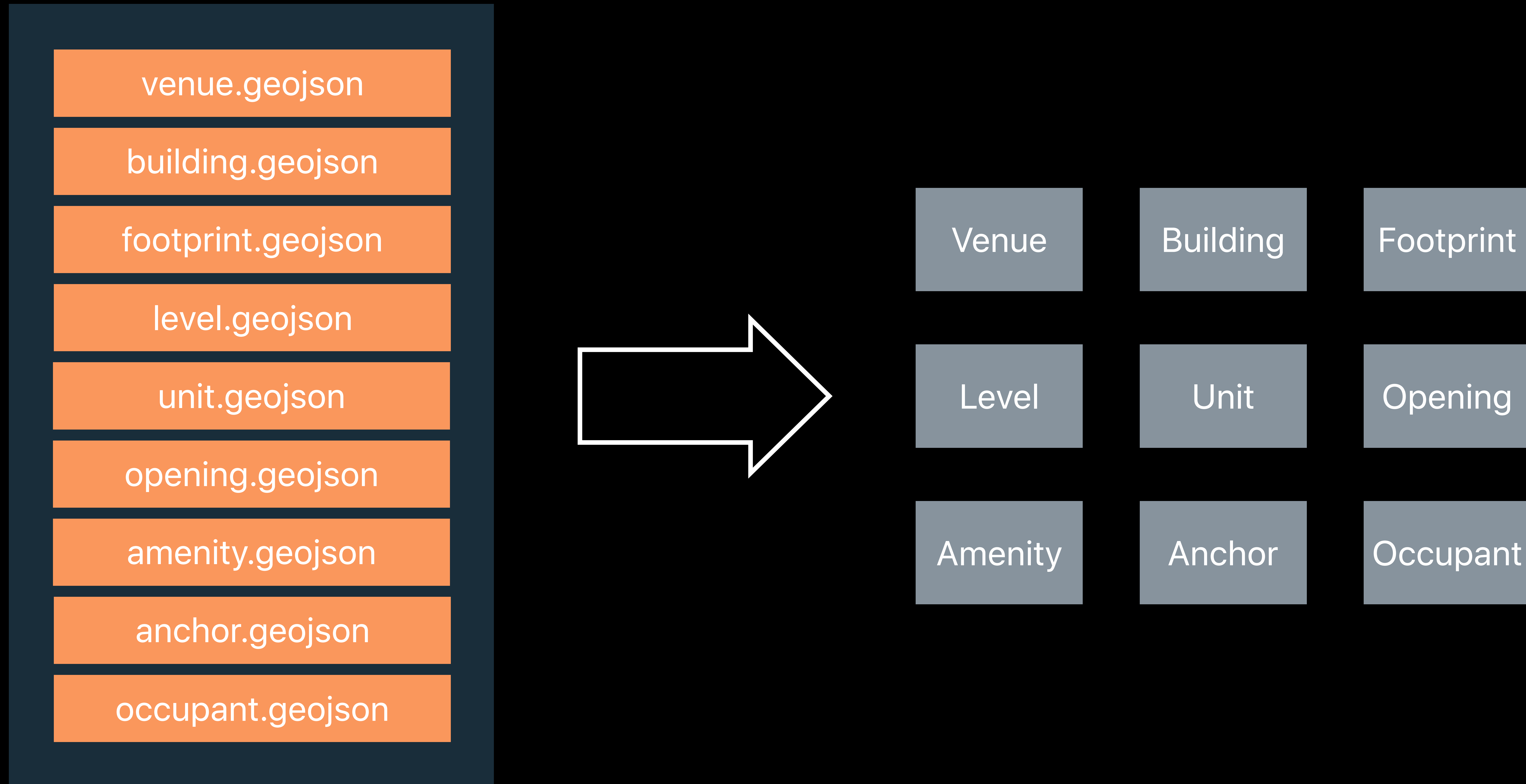
SP+ Parking

Legal

Kukar's House

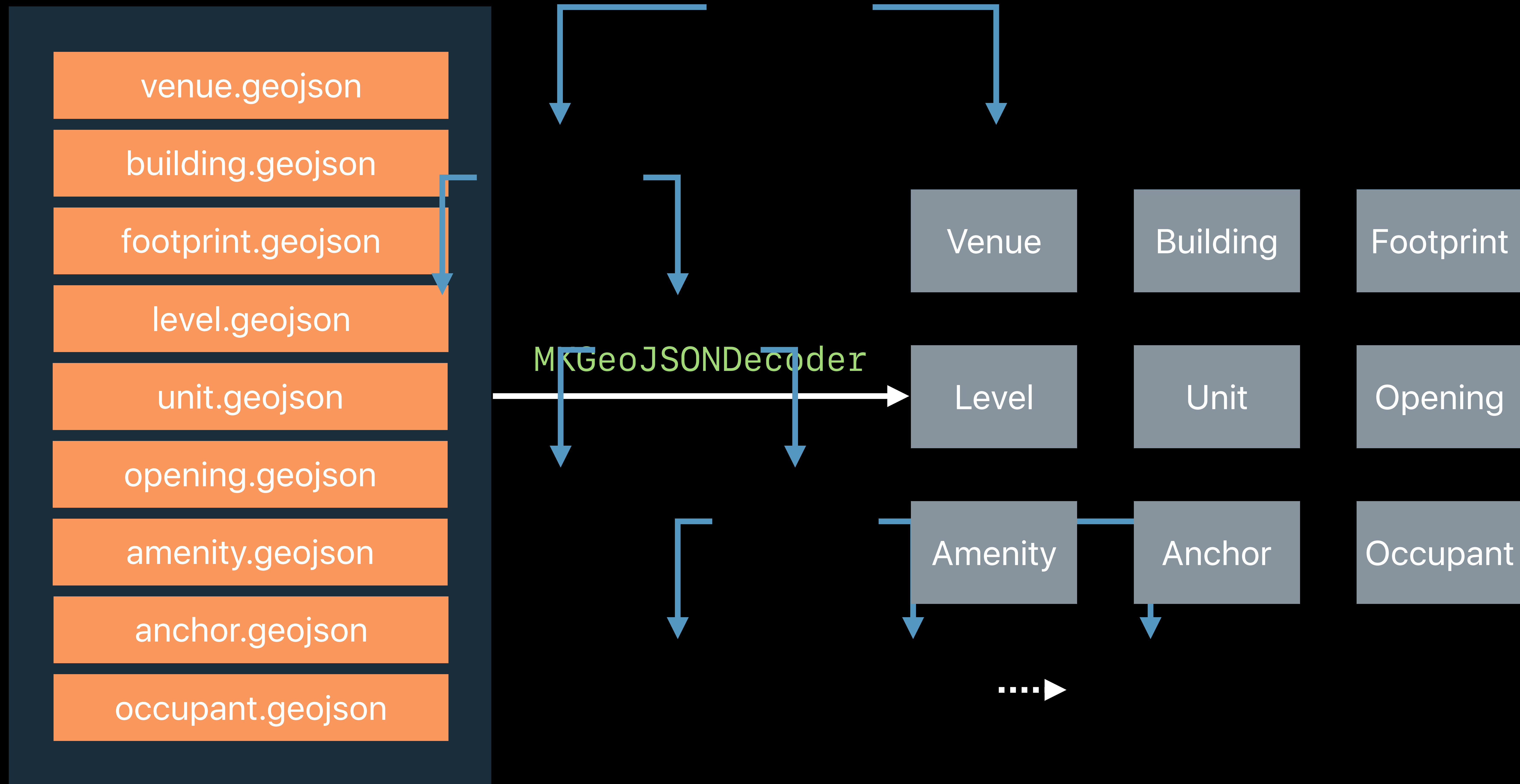


# Step 1: IMDF Model Classes



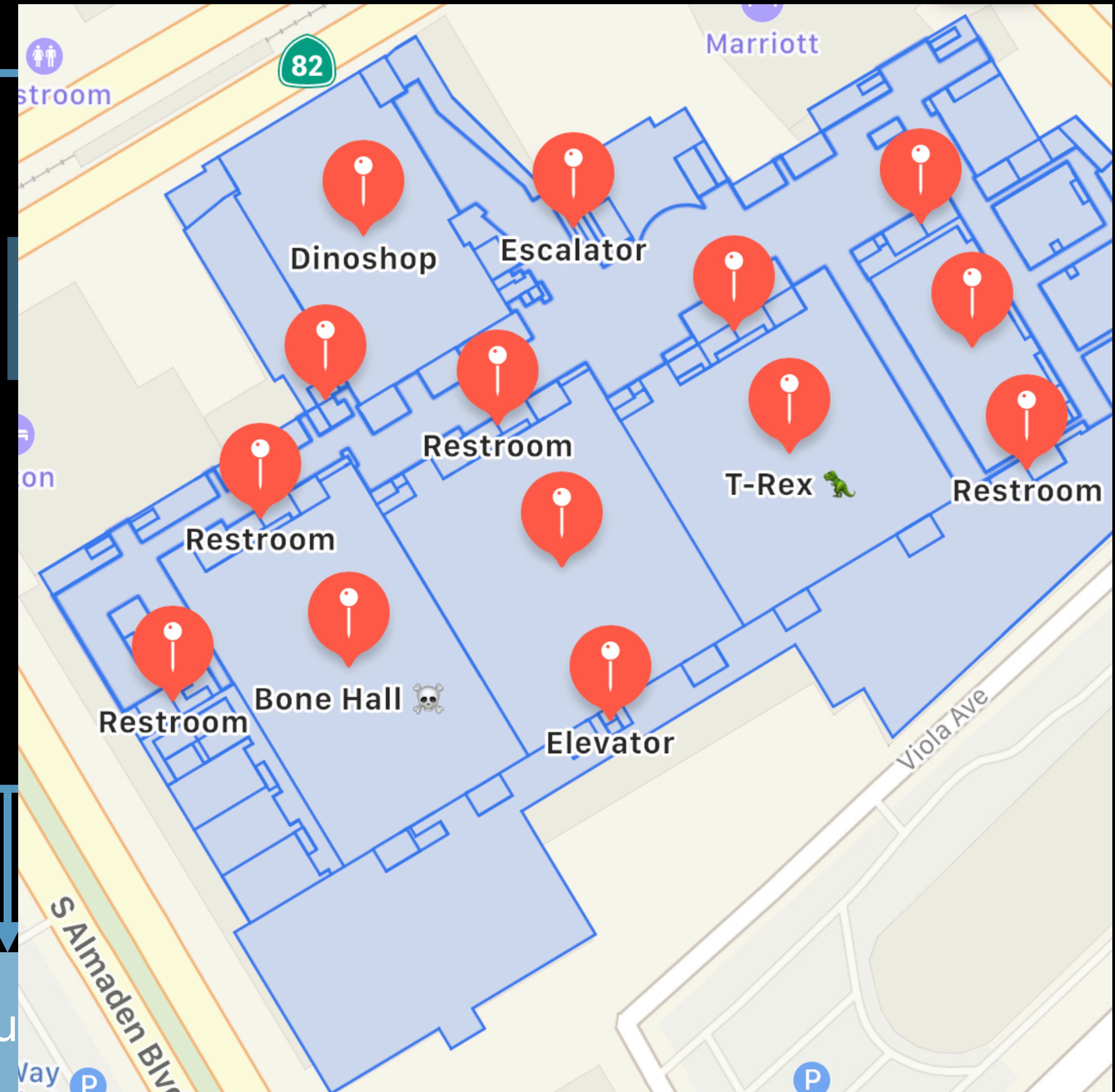
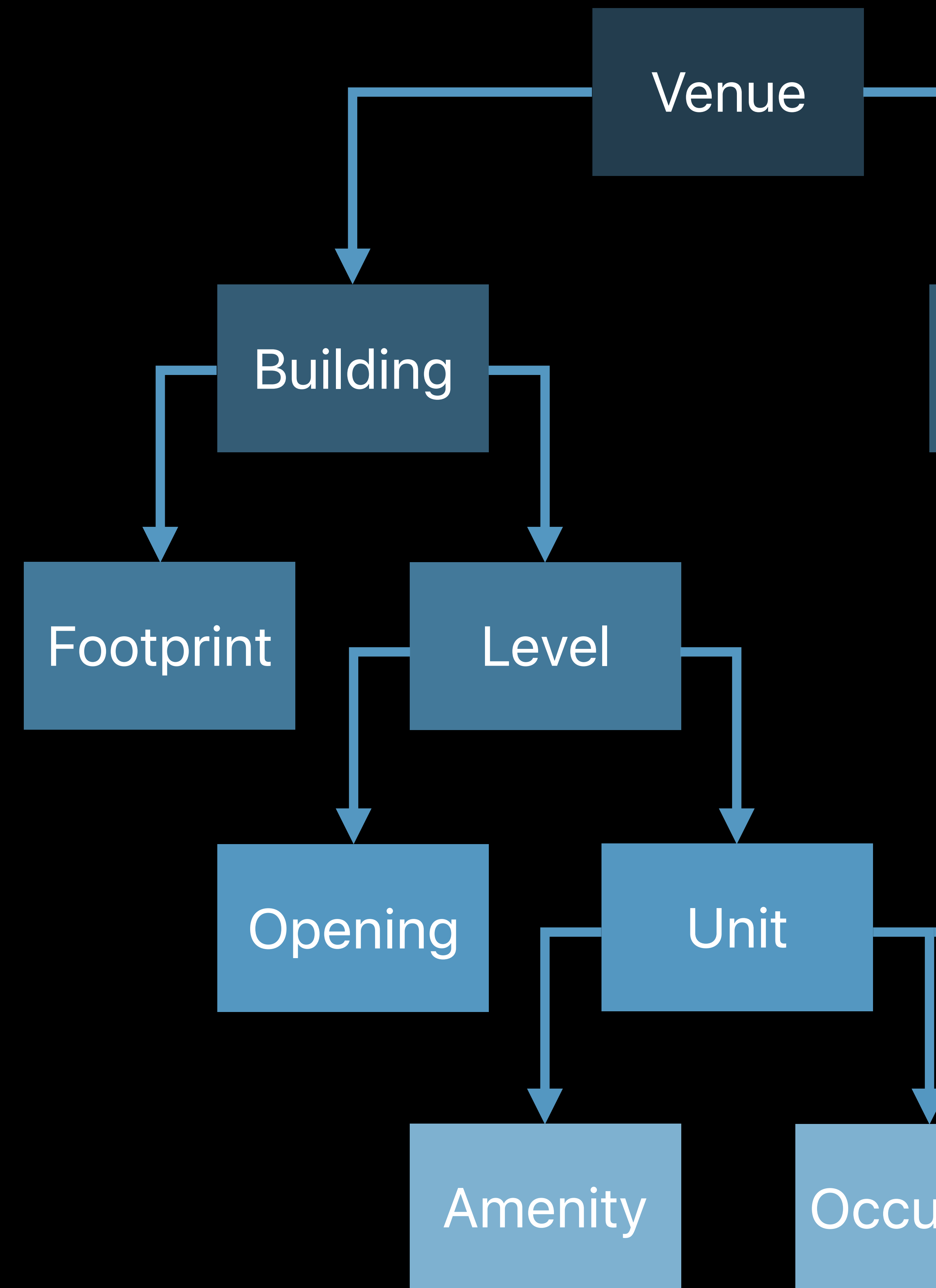


# Step 2: Decode IMDF





# Step 3: Render IMDF





```
// Step 1: IMDF Model Classes  
// Feature.swift
```

```
class Feature<Properties: Decodable>: NSObject, IMDFDecodableFeature {  
    let identifier: UUID  
    let properties: Properties  
    var geometry: [MKShape & MKGeoJSONObject]  
}
```

```
class Unit: Feature<Unit.Properties> {  
    . . .  
}
```





NEW

```
// Step 2: Decode IMDF
// Decoding IMDF file for Level feature

func decodeLevel(fileURL: URL) throws -> [Level] {
    let data = try Data(contentsOf: fileURL)
    let geoJSONDecoder = MKGeoJSONDecoder()
    let geoJSONFeatures = try geoJSONDecoder.decode(data)
    guard let features = geoJSONFeatures as? [MKGeoJSONFeature] else {
        throw IMDFError.invalidType
    }

    let levelFeatures = try features.map { try Level(feature: $0) }
    return levelFeatures
}
```



```
// Step 3: Render IMDF
// MapKit APIs for displaying overlays and annotations

// Add IMDF features as overlays and annotations to map
func addOverlays(_ overlays: [MKOverlay])
func addAnnotations(_ annotations: [MKAnnotation])

// MKMapViewDelegate protocol to customize appearance
func mapView(_ mapView: MKMapView, rendererFor overlay: MKOverlay) -> MKOverlayRenderer
func mapView(_ mapView: MKMapView, viewFor annotation: MKAnnotation) -> MKAnnotationView?
```



***Demo***

# Displaying Indoor Map in an iOS Application

Create model classes

Venue

Building

Footprint

Level

Unit

Opening

Amenity

Anchor

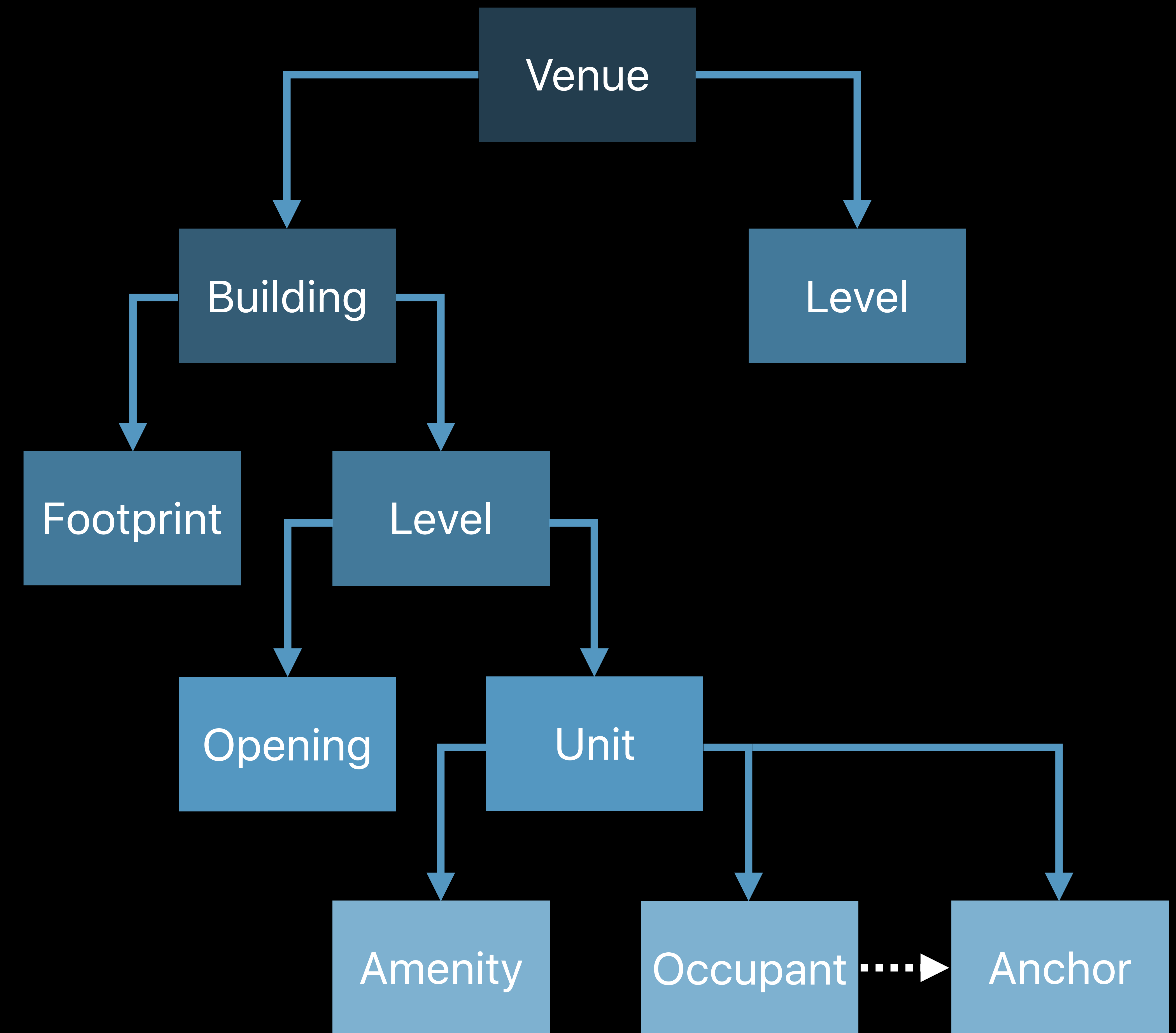
Occupant



# Displaying Indoor Map in an iOS Application

Create model classes

Decode IMDF



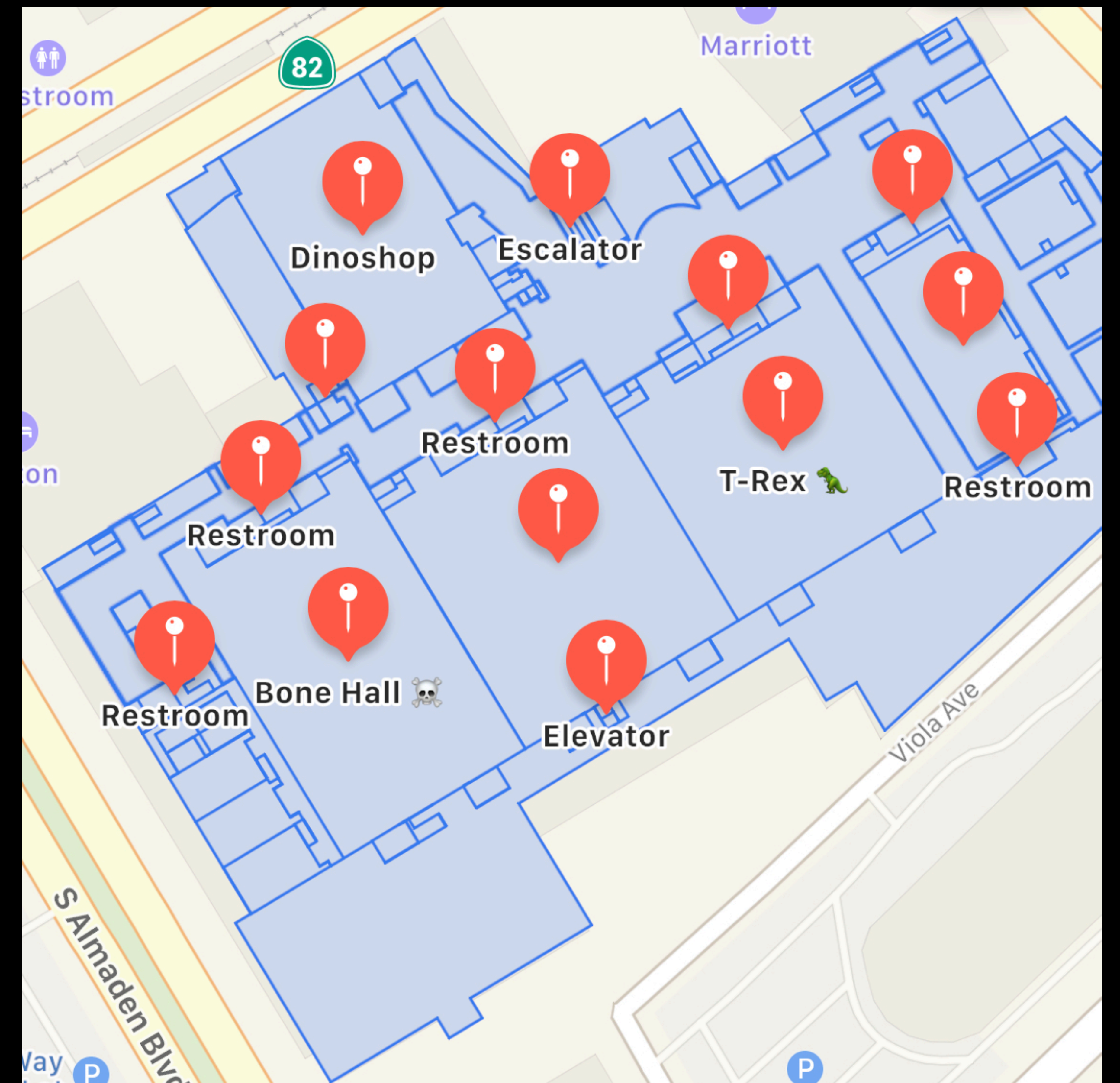


# Displaying Indoor Map in an iOS Application

Create model classes

Decode IMDF

Render IMDF





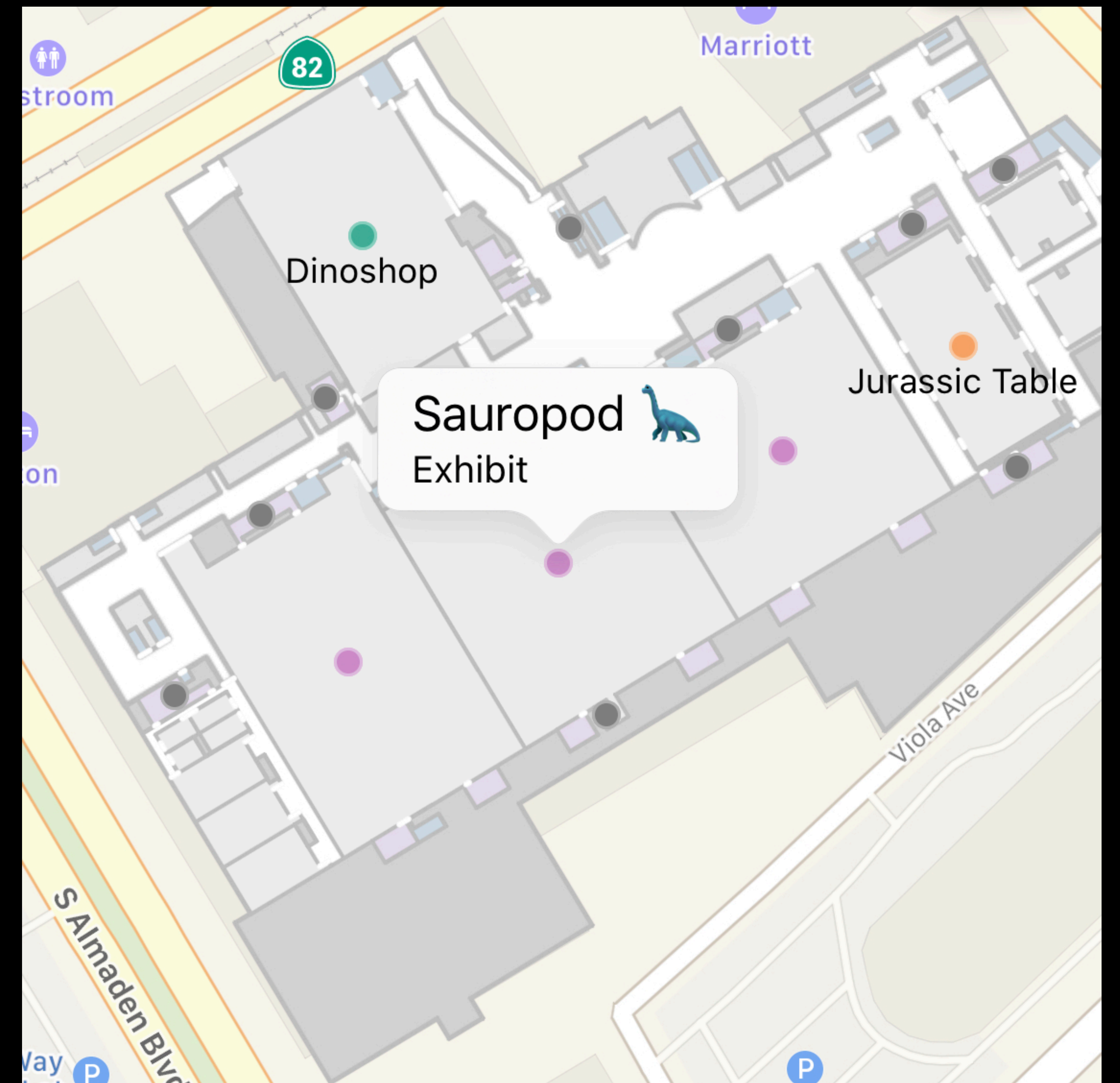
# Displaying Indoor Map in an iOS Application

Create model classes

Decode IMDF

Render IMDF

Apply style





# Displaying Indoor Map in an iOS Application

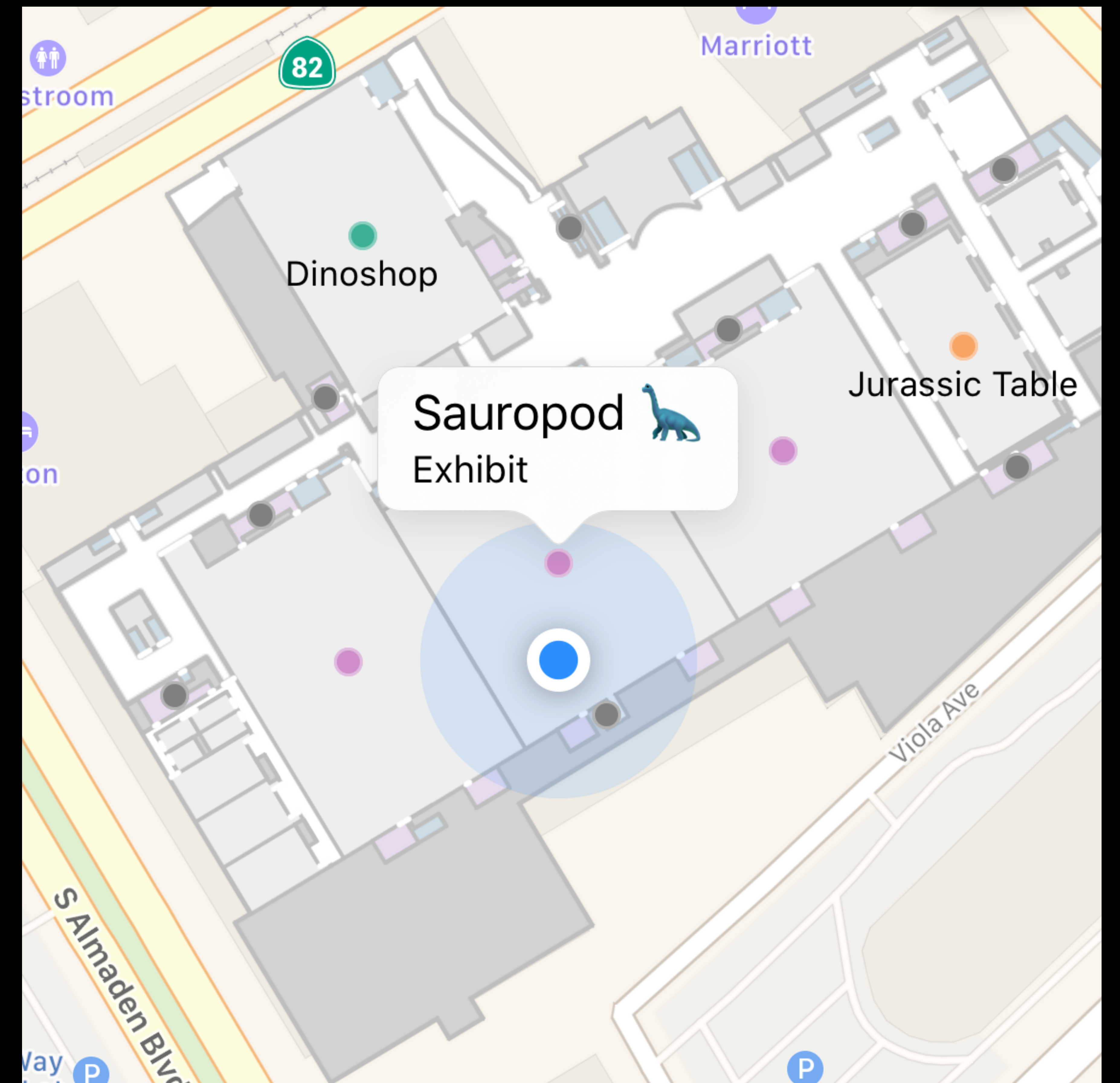
Create model classes

Decode IMDF

Render IMDF

Apply style

Display indoor location





```
// Step 4: Style
```

```
protocol StylableFeature {  
    var geometry: [MKShape & MKGeoJSONObject] { get }  
    func configure(overlayRenderer: MKOverlayPathRenderer)  
    func configure(annotationView: MKAnnotationView)  
}
```

***Demo***

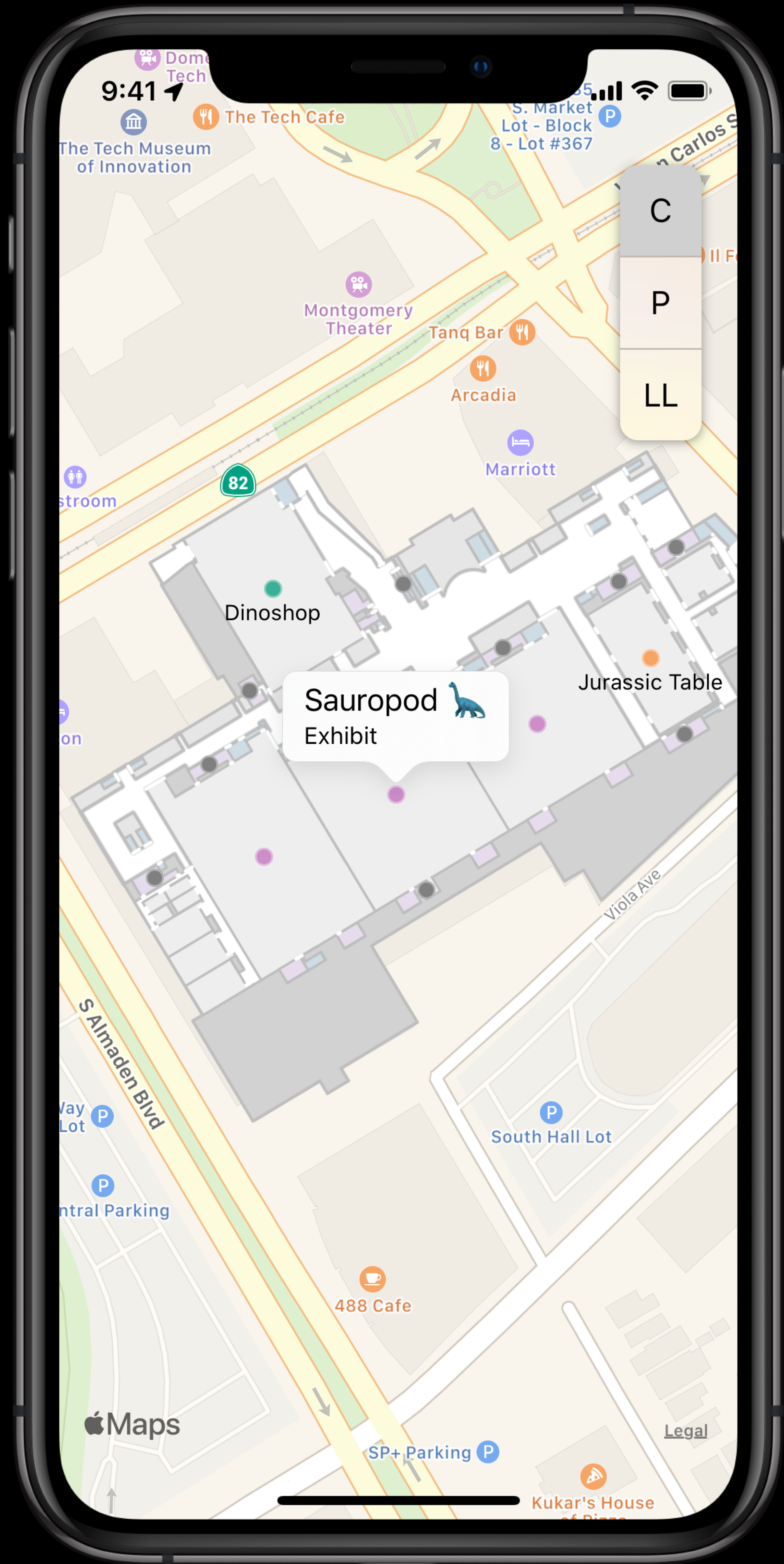


Indoor Mapping Data Format (IMDF)

Indoor Maps Program

Displaying Indoor Maps in an iOS App

Displaying Indoor Maps in a Web App



9:41

The Tech Museum of Innovation

The Tech Cafe

Montgomery Theater

Tanq Bar

Arcadia

Marriott

Dinoshop

Sauropod Exhibit

Jurassic Table

488 Cafe

Apple Maps

SP+ Parking

Kukar's House

Legal

C

P

LL

82

S Almaden Blvd

Viola Ave

South Hall Lot

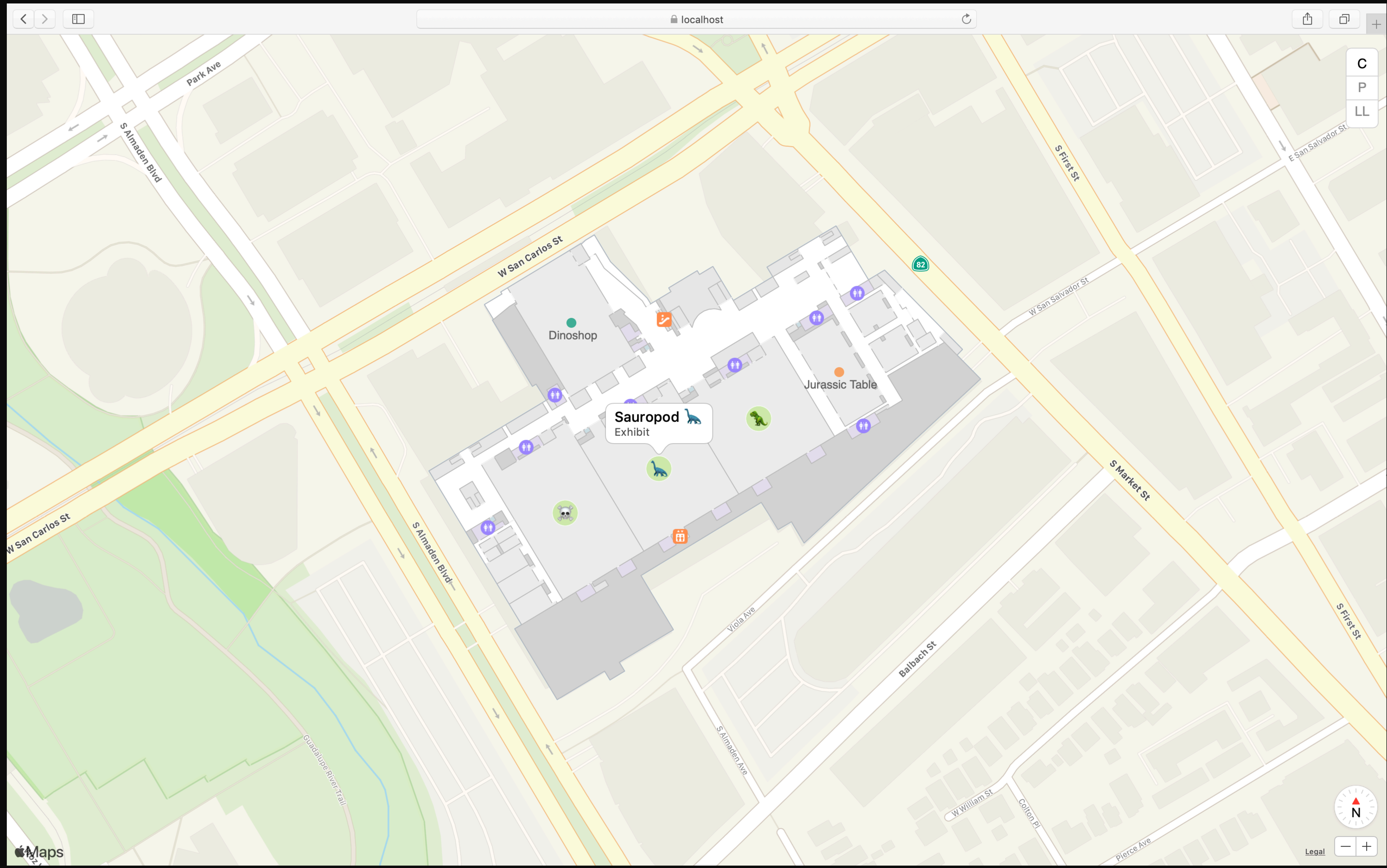
Central Parking

S. Market Lot - Block 8 - Lot #367

Carlos S

Way Lot





MacBook Pro



```
// mapkit.importGeoJSON
```

```
let delegate = {
```

```
  geoJSONDidError(error, geJSON) {},
```

```
  geoJSONDidComplete(items, geoJSON) {}
```

```
};
```

```
mapkit.importGeoJSON(geoJSON, delegate);
```



```
// mapkit.importGeoJSON
```

```
let delegate = {
```

```
  geoJSONDidError(error, geoJSON) {},
```

```
  geoJSONDidComplete(items, geoJSON) {
```

```
    map.showItems(items); // or addItem() if no need to center the map on these items
```

```
  }
```

```
};
```

```
mapkit.importGeoJSON(geoJSON, delegate);
```

```
// styleForOverlay

let delegate = {
  geoJSONDidError(error, geoJSON) { ... },
  geoJSONDidComplete(items, geoJSON) { ... },
  styleForOverlay(overlay, geoJSON) {
    overlay.style.strokeColor = "white";
    overlay.style.lineDash = [3, 3];
    return overlay.style;
  }
};

mapkit.importGeoJSON(geoJSON, delegate);
```



```
// itemForPoint

let delegate = {
  geoJSONDidError(error, geoJSON) { ... },
  geoJSONDidComplete(items, geoJSON) { ... },
  styleForOverlay(overlay, geoJSON) { ... },
  itemForPoint(coordinate, geoJSON) {
    const options = {
      title: "Title of the annotation",
      displayPriority: 500,
      color: "green"
    };
    return new mapkit.MarkerAnnotation(coordinate, options);
  }
};

mapkit.importGeoJSON(geoJSON, delegate);
```

***Demo***



# Displaying Indoor Map in a Web App

Organize features by type and filter by level

# Displaying Indoor Map in a Web App

Organize features by type and filter by level

Add a level picker and display features on level



# Displaying Indoor Map in a Web App

Organize features by type and filter by level

Add a level picker and display features on level

Style units and openings

# Displaying Indoor Map in a Web App

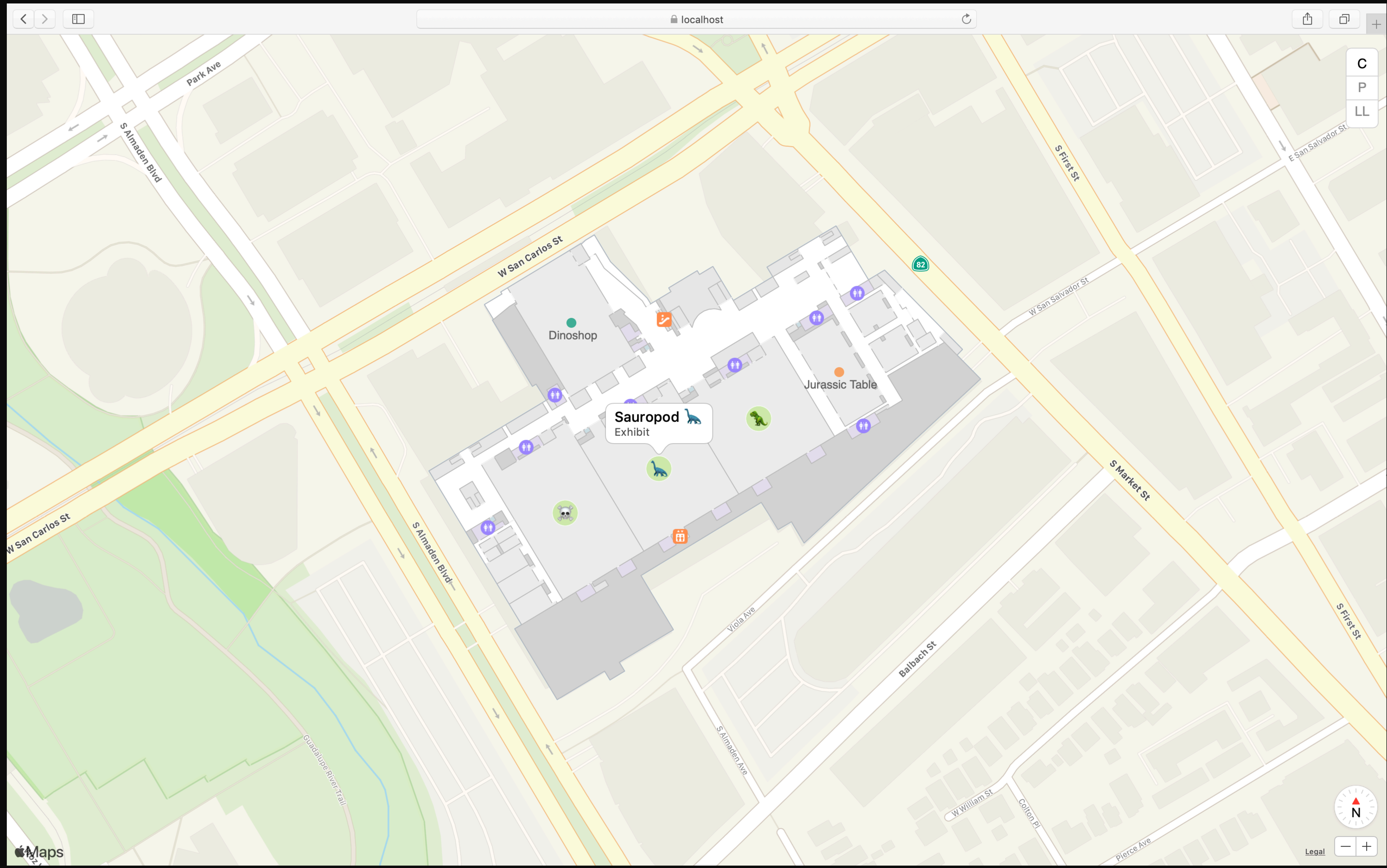
Organize features by type and filter by level

Add a level picker and display features on level

Style units and openings

Customize annotations





MacBook Pro



Indoor Mapping Data Format (IMDF)

Indoor Maps Program

Displaying Indoor Maps in an iOS App

Displaying Indoor Maps in a Web App



# Summary

Indoor Maps — Best practices

Apply styling to match the visual style of your app or brand

# Summary

Indoor Maps — Best practices

Apply styling to match the visual style of your app or brand

Select distinct styles based on category



# Summary

Indoor Maps — Best practices

Apply styling to match the visual style of your app or brand

Select distinct styles based on category

Use easily recognizable icons

# Summary

Indoor Maps — Best practices

Apply styling to match the visual style of your app or brand

Select distinct styles based on category

Use easily recognizable icons

Adjust map detail based on the zoom level



# Summary

Indoor Maps — Best practices

Apply styling to match the visual style of your app or brand

Select distinct styles based on category

Use easily recognizable icons

Adjust map detail based on the zoom level

Provide context

# Summary

## Indoor Maps — Best practices

Apply styling to match the visual style of your app or brand

Select distinct styles based on category

Use easily recognizable icons

Adjust map detail based on the zoom level

Provide context

Enable indoor user location



# More Information

[developer.apple.com/wwdc19/241](https://developer.apple.com/wwdc19/241)

---

MapKit, MapKit JS, and Indoor Location Lab

Friday, 3:00

---

What's New in MapKit and MapKit JS

WWDC 2019

---

Introducing the Apple Maps Program

WWDC 2019

---

 WWDC19