

Realtime Location Queries With Firebase

Geohashing



The Problem

Querying for nearby places is an almost essential feature to many web and mobile applications.

Typically when searching for nearby locations, you would need to create a range of latitude and longitude coordinates that could be used to capture a list of restaurants, stores, etc. However, Firebase Realtime Database queries can only perform range filtering on a single field. This means that we are unable to query on both latitude and longitude as separate fields.



The Solution



Geohashing is a geocoding method used to encode geographic coordinates (latitude and longitude) into a short string of digits and letters. The more characters in the string, the more precise the location.

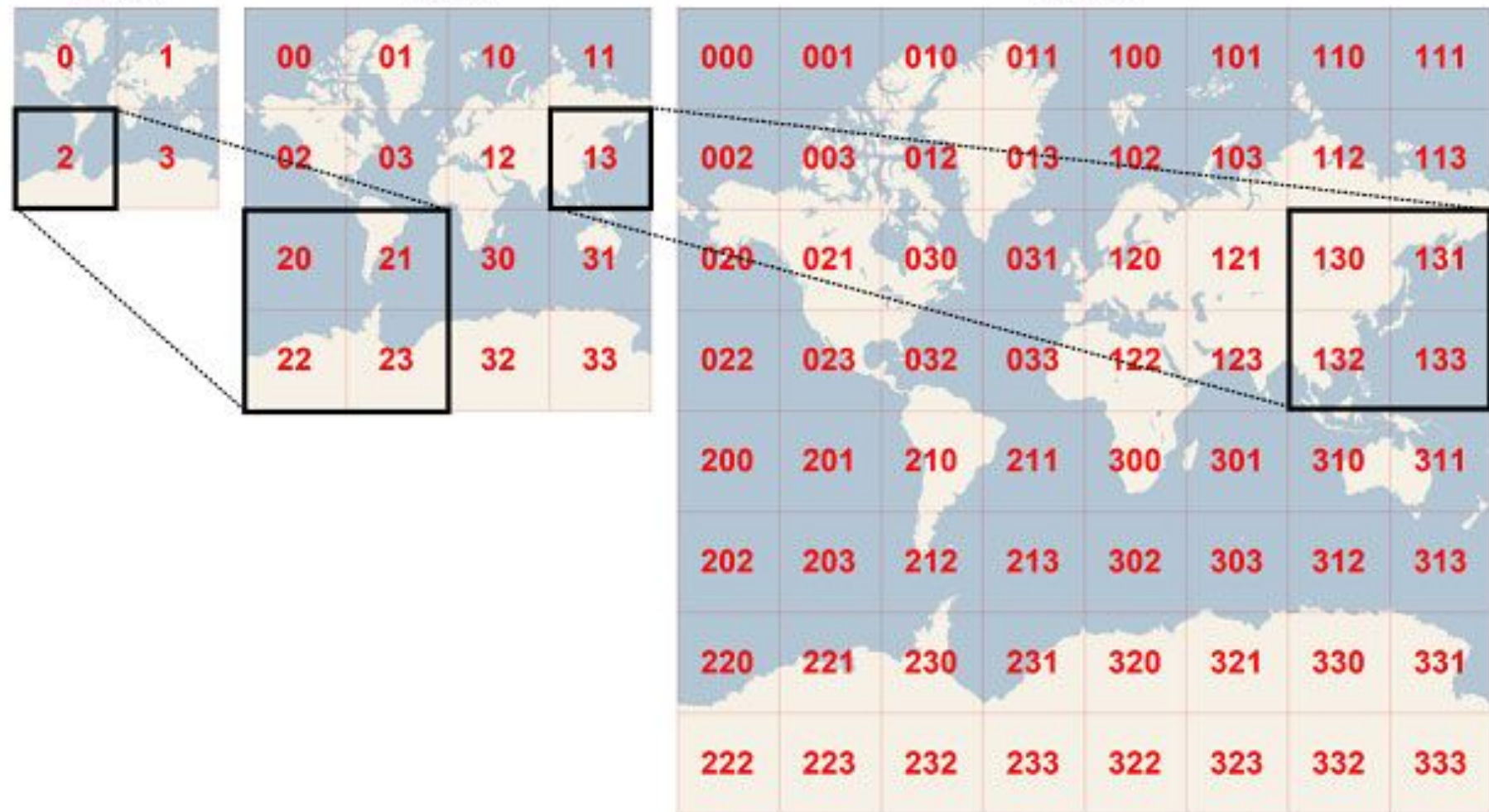
How Geohashing works ?

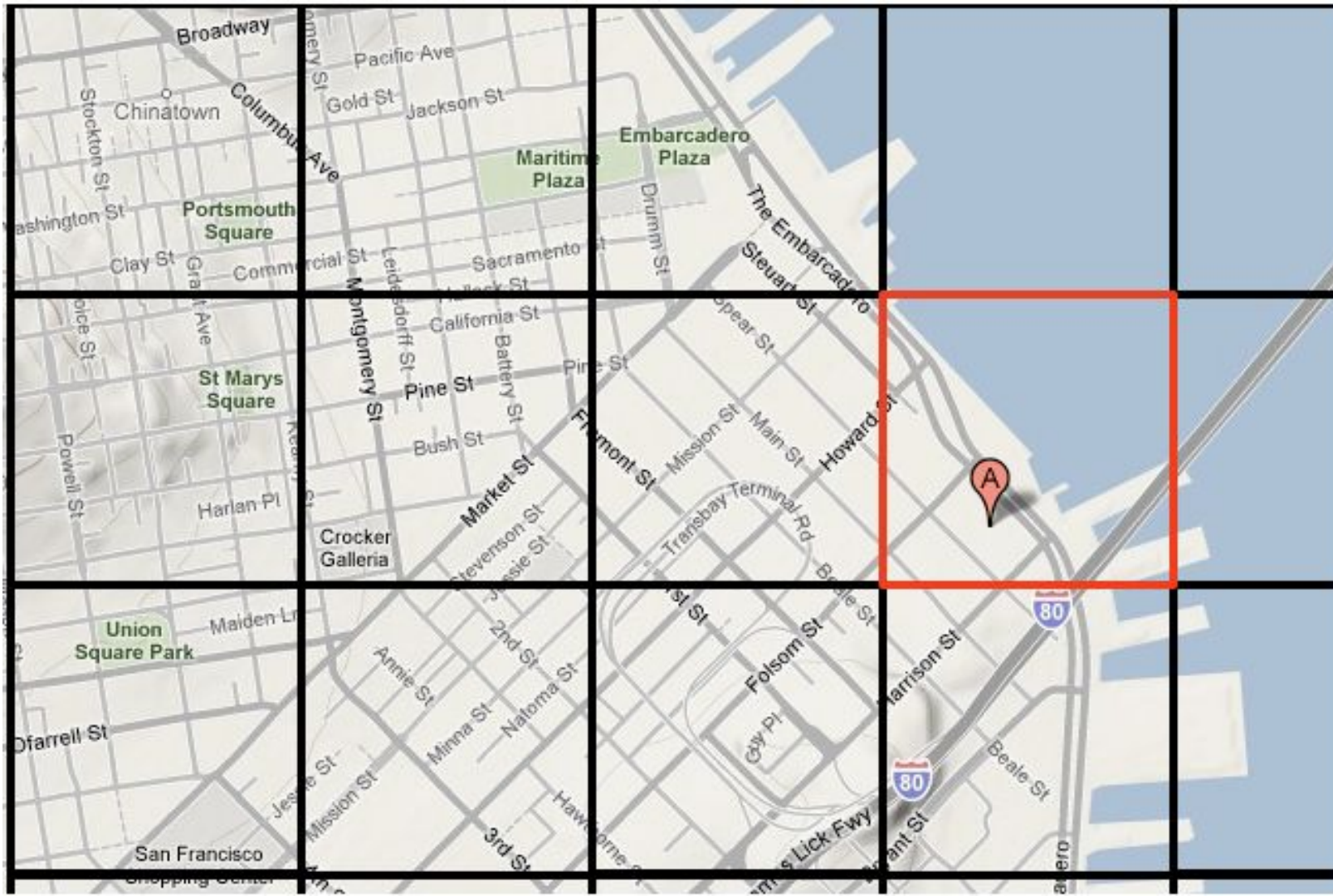
- Geohashes use Base-32 alphabet encoding (characters can be 0 to 9 and A to Z, excl "A", "I", "L" and "O").
- Imagine the world is divided into a grid with 32 cells. The first character in a geohash identifies the initial location as one of the 32 cells.
- This cell will also contain 32 cells, and each one of these will contain 32 cells (and so on repeatedly).
- Adding characters to the geohash sub-divides a cell, effectively zooming in to a more detailed area.
- The precision factor determines the size of the cell.
- A precision factor of one creates a cell 5,000km high and 5,000km wide
- A precision factor of six creates a cell 0.61km high and 1.22km wide
- A precision factor of nine creates a cell 4.77m high and 4.77m wide (cells are not always square).

Level 1

Level 2

Level 3







DEMO

Thank You

Any Questions ?