Knowing Where via Geo-Referencing through Google's S2-Cell System

Zhangyu Wang
University of California Santa Barbara

KnowWhereGraph (KWG) is a knowledge base composed of triples.

- ➔ A triple is a **fact** expressed as (*s*, *r*, *t*), where *s* is the source, *r* is the relation, and *t* is the target. For example, “Hurricane Katrina hits Florida” can be triplified as (*Hurricane*, *influences*, *Florida*).
- ➔ Geospatial relations are just a special kind of relations.
- ➔ To relate entities in the graph with geospatial concepts is called geo-referencing.
- ➔ Useful to answer questions like “Where did X happen?”

How to do geo-referencing in KWG?

There are multiple natural ways of geo-referencing:

- ➔ Raw coordinates
- ➔ Administrative hierarchy
- ➔ Grid system

Major concerns:

- ➔ Appropriate granularity
- ➔ Uniform and coherent indexing

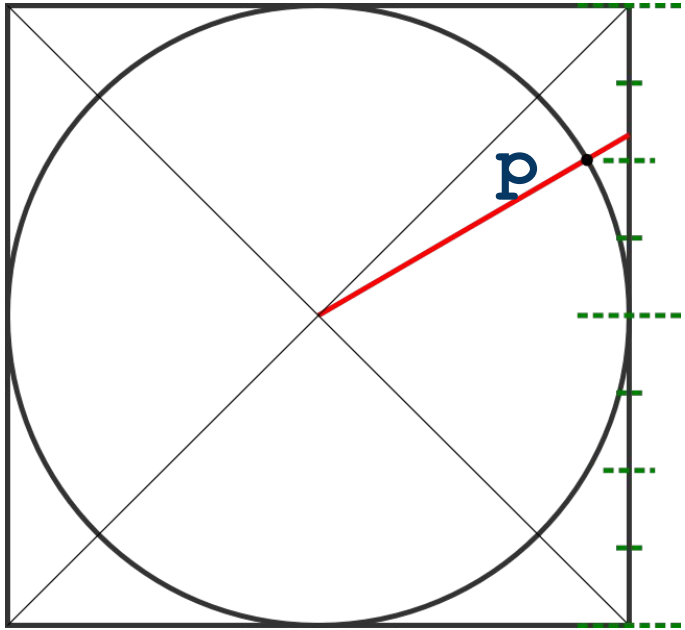
Google's S2-Cell System fits our needs well

- ➔ Hierarchical decomposition of the sphere into cells
- ➔ Uniform areas of cells – easy to approximate region sizes based on cell amount
- ➔ Indexing well organized – 64-bit index, inclusion/bordering relations can be directly referred from the cell indices without computation
- ➔ Multiple granularity: from level 0 (dividing the entire sphere into 6 surfaces) to level 30 (around 1 cm^2). It also provides APIs for computing the smallest set of covering cells.

https://s2sphere.readthedocs.io/en/latest/

Hierarchical Decomposition

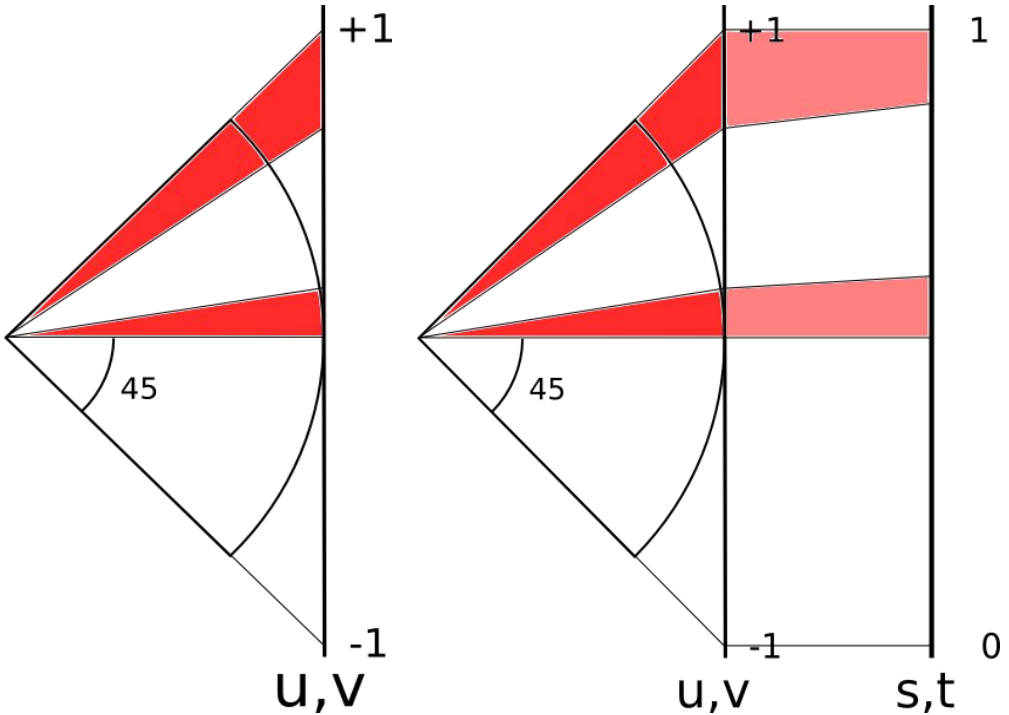
- ➔ Enclose sphere in cube
- ➔ [-1,1] x [-1,1] x [-1,1]
- ➔ Project p on the cube
- ➔ Build a quad-tree on each cube face
- ➔ Find quad-tree cell that contains the projection of p



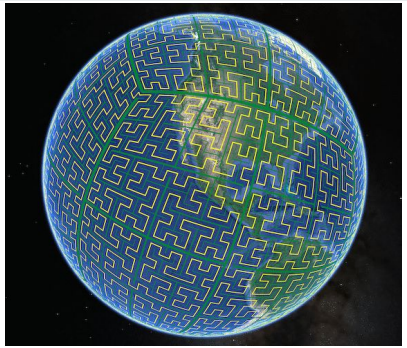
https://www.slideshare.net/SwagataPrateek2/geometry-on-the-sphere-googles-s2-librarypdf

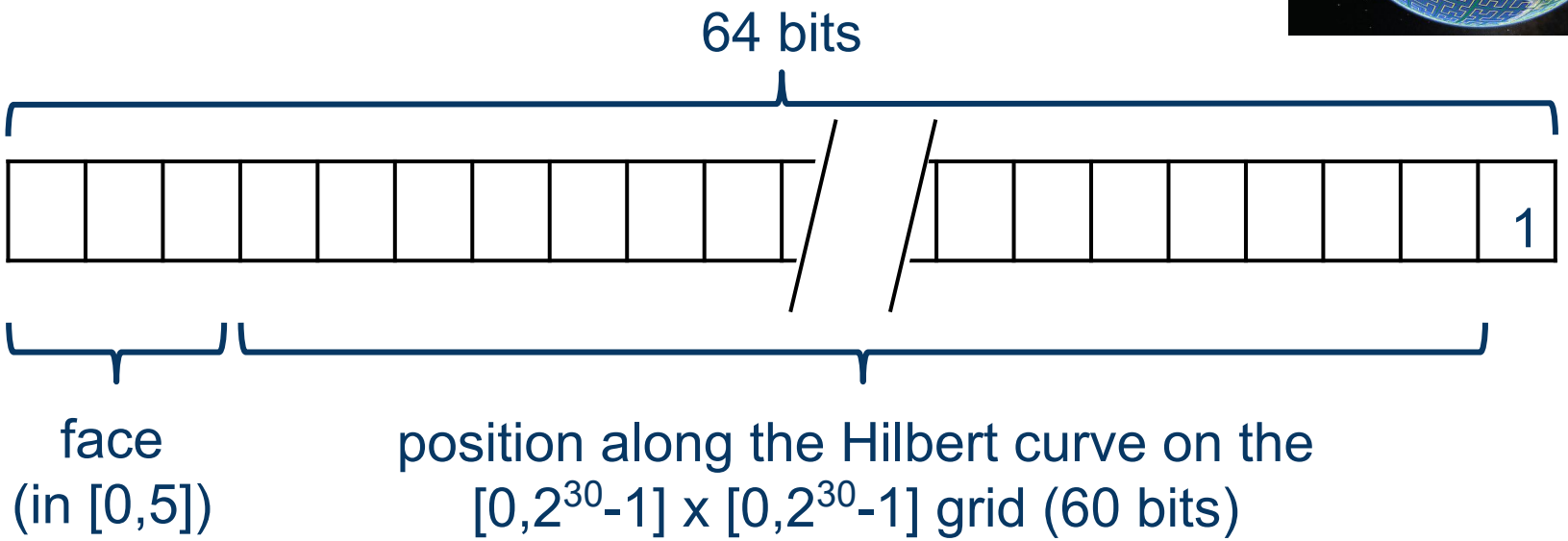
Uniform Area

➔ Non-linear (quadratic) transformation: more accurate than linear, and computationally efficient.



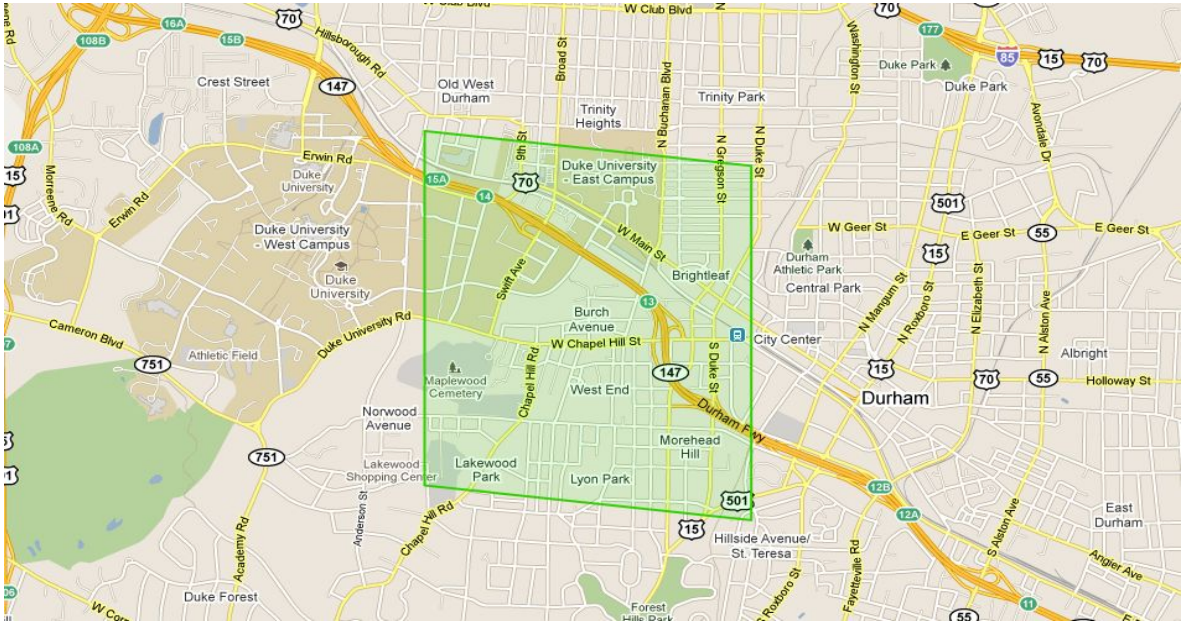
Organized Indexing





https://s2geometry.io/devguide/s2cell_hierarchy.html

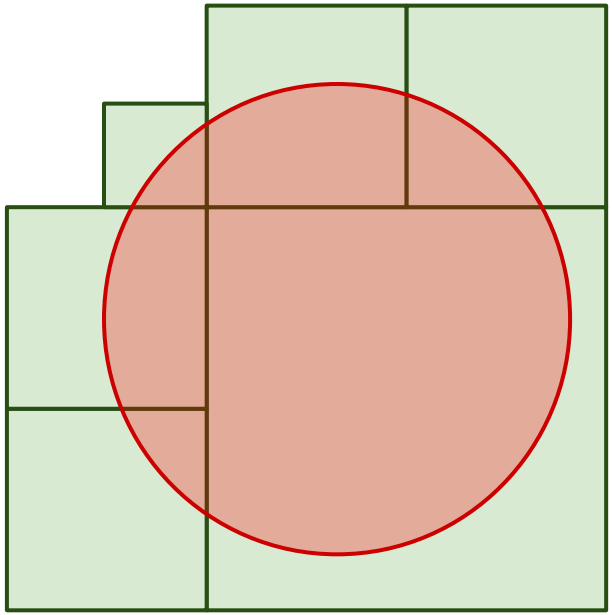
Organized Indexing



Id: 0x89ace41000000000 (0b1000100110101100111001000001000...), Level: 12

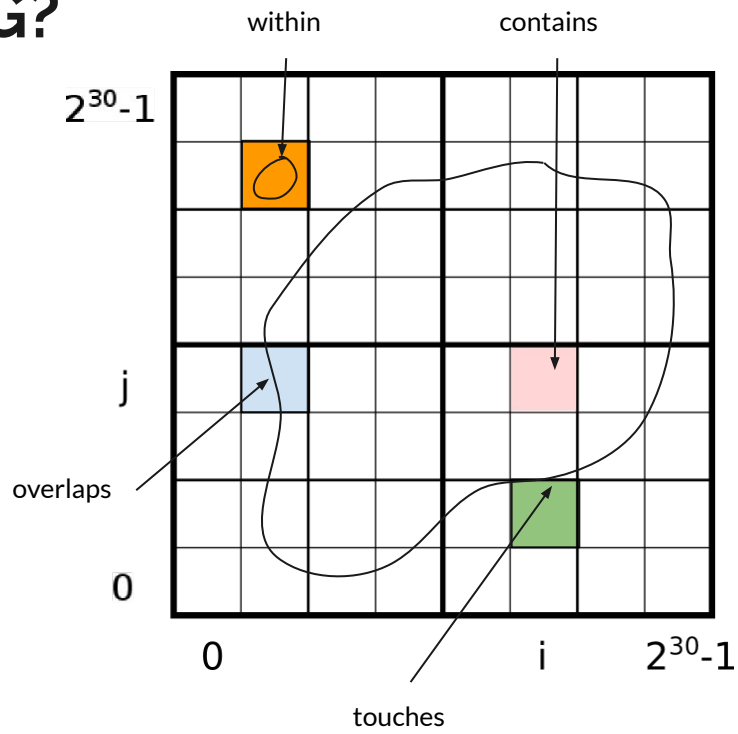
Computing set of covering cells

- ➔ Given a polygon (e.g., red region to the right).
- ➔ Given a min level and a max level of cells.
- ➔ Compute the smallest set of cells that are between the min/max levels and fully cover the polygon.
- ➔ Notice: there may be multiple sets of possible covering cells – that is, the output is not unique. These cells are also not of the same levels.



How do we use S2Cells in KWG?

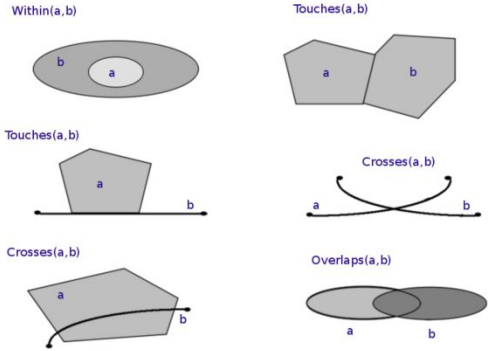
- ➔ Spatial relations: *within*, *contains*, *touches*, *overlaps*.
- ➔ Spatial join the polygons of the entities in the graph with the S2 cell polygons to get the spatial relations.
- ➔ Conduct spatial joins from level 0 to level 13.
- ➔ Then we obtain a triple like (wild_fire_1, overlaps, s2_0x100000)



DE-9IM

- ➔ Dimensionally Extended 9-Intersection Model
- ➔ Based on a 3×3 intersection matrix with the form:

	Interior	Boundary	Exterior
Interior	 $\dim[I(a)\cap I(b)] = 2$	 $\dim[I(a)\cap B(b)] = 1$	 $\dim[I(a)\cap E(b)] = 2$
Boundary	 $\dim[B(a)\cap I(b)] = 1$	 $\dim[B(a)\cap B(b)] = 0$	 $\dim[B(a)\cap E(b)] = 1$
Exterior	 $\dim[E(a)\cap I(b)] = 2$	 $\dim[E(a)\cap B(b)] = 1$	 $\dim[E(a)\cap E(b)] = 2$

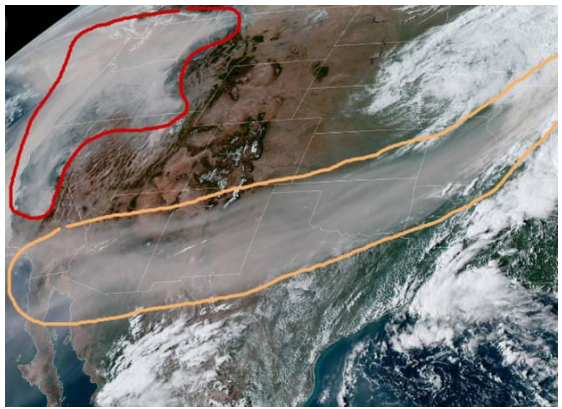


Example of an overlap relation: T*T***T**

Type	Description
T	Intersection exists
F	Intersection does not exist
*	Does not matter
0	Intersection is a point (dimension = 0)
1	Intersection is a line(dimension = 1)
2	Intersection is a polygon (dimension = 2)

Difficulties and solutions

Computing spatial relations for large polygons (e.g., smoke plumes that span over thousands of miles) is very time-consuming.



A satellite image shows smoke from the Western wildfires stretching as far east as Michigan.
THE NATIONAL WEATHER SERVICE WEATHER PREDICTION CENTER

- ➔ Compute the minimum set of covering cells first, and only do spatial join with the covering cells.
- ➔ *within* and *contains* are inverse relations, so we only need to model *contains*.
- ➔ Hierarchical spatial join: *contains* relations are transitive. If A *contains* B, then A also *contains* all child cells of B.