Scaling Spatio-Temporal Analytics with GeoMesa

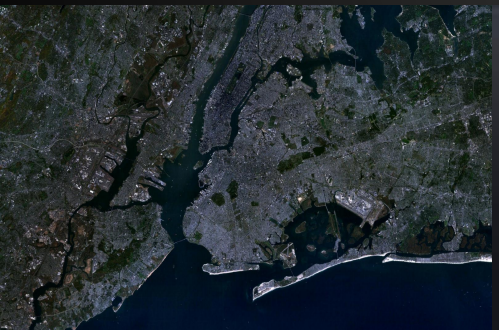
Maryland Data Science Meetup -- August 18, 2014
Andrew Hulbert -- ahulbert at ccri.com

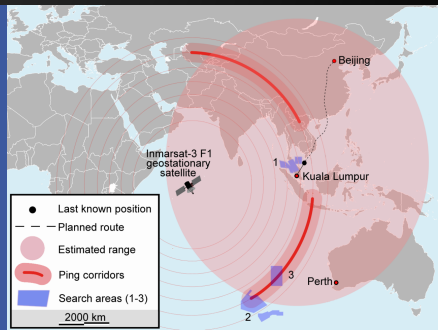
Agenda

- Spatio-Temporal Data
- Distributed Indexing Strategies
- GeoMesa Architecture
- SpatioTemporal Analytics
- Roadmap & Future Work

What is Spatio-Temporal Data?

- Satellite Imagery
- FAA Flight Information
- Twitter & Social Media
- GPS-Enabled Apps
- Network Traffic & Clickstreams





What is GeoMesa?

- Distributed Spatio-Temporal Database
  - Apache Accumulo
  - Apache Hadoop
- Standards-based Data Access
  - GeoTools
  - GeoServer
- LocationTech Open Source
  - Eclipse Foundation

Open Source Geo

OGC™
Open Geospatial Consortium, Inc.

GeoTools

GeoServer

LocationTech

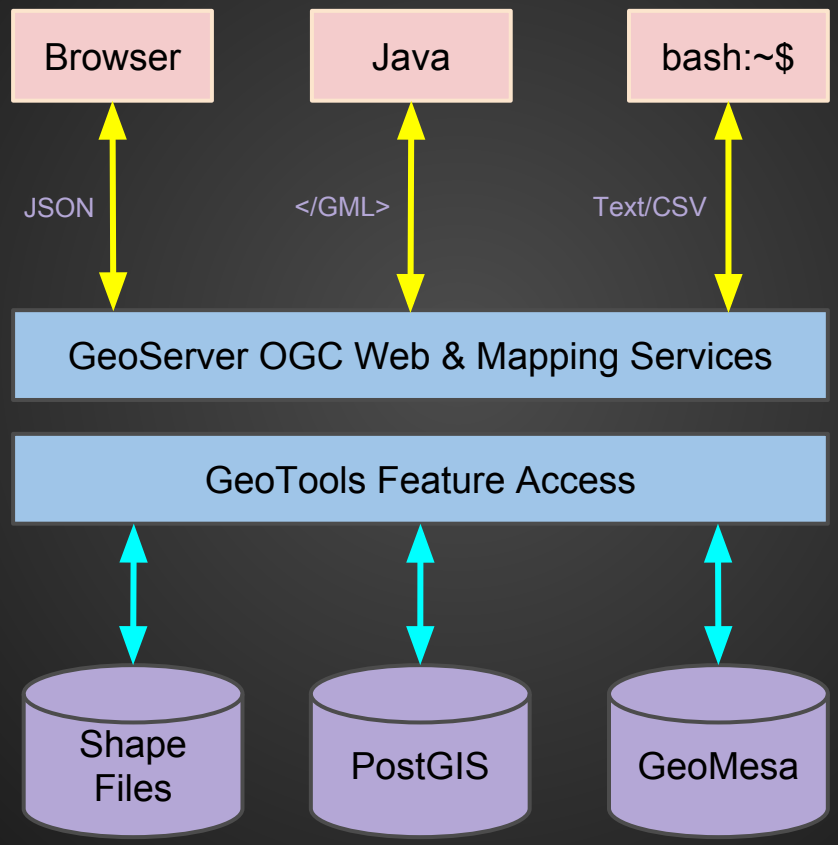
Simple Features

```
{
  "id" : "c11b0bdc-a400-42b3-9177-81f835bff6f5",
  "geometry" : {
    "coordinates" : [-74.189, 40.7728],
    "type" : "Point"
  },
  "type" : "Feature",
  "properties" : {
    "dtg" : "2014-07-08T04:08:43.000+0000",
    "text" : "\"This is our city!\" -David Ortiz",
    "tweet_id" : 010101001010101001010101001,
    "user_id" : 010101001,
    "user_name" : "Fresh Prince of Belair"
  }
}
```

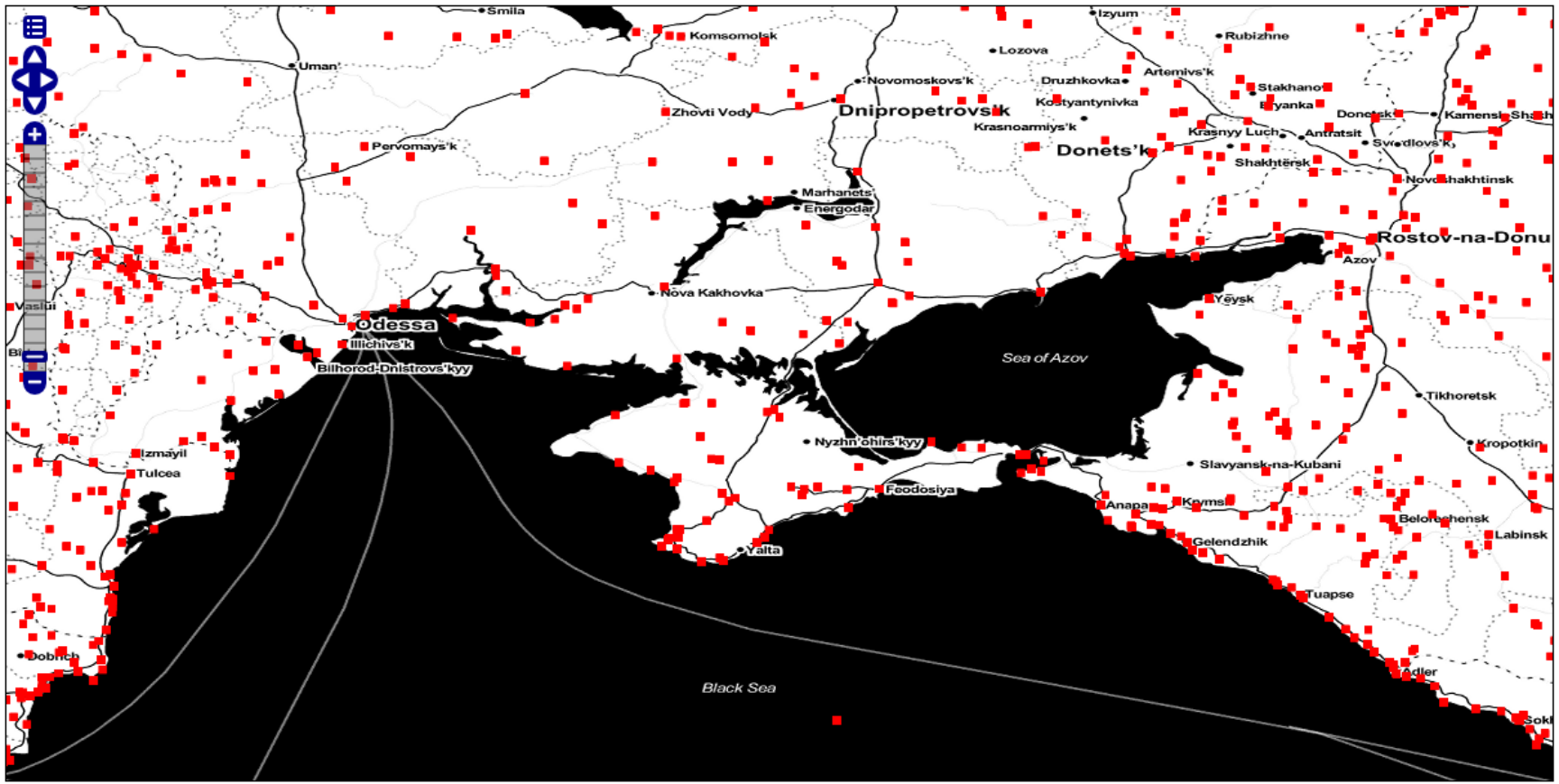
Common Query Language (CQL)

```
BBOX(geom, 40, 30, 45, 35) AND
  name = “Person 1” AND
  age = 30 AND
  hobby <> “texting”
```

Tiered Client Architecture



Full Stack Open Source



Database Evolution

- PostGIS
- R-Trees
- Vertical Scaling

- Hadoop
- Key Value Stores
- Linear Scaling

Software Engineering 101

Hop on over from PostGIS….

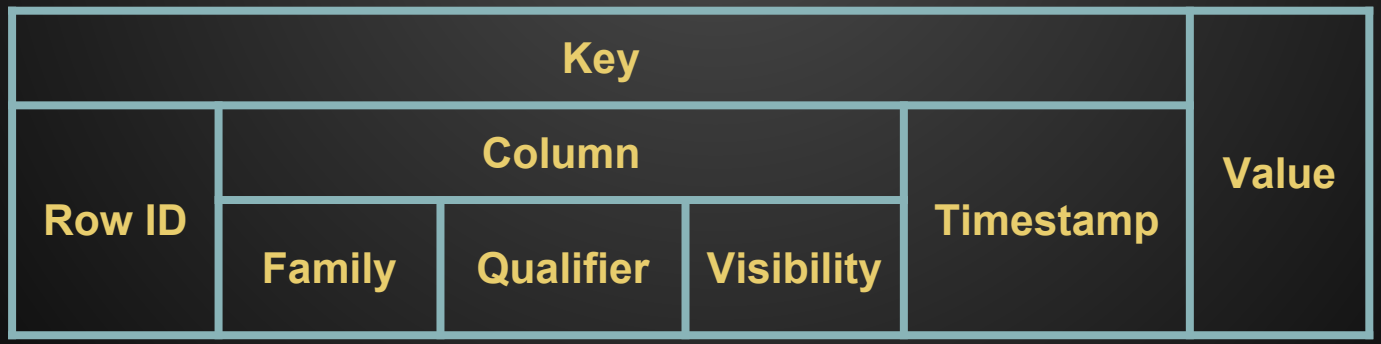
```
val params = Map("dbtype"     -> "postgis",
                 "host"       -> "localhost",
                 "port"       -> 5432,
                 "schema"     -> "public",
                 "database"   -> "gdelt",
                 "user"       -> "postgres",
                 "passwd"     -> "postgres")
val dataStore = DataStoreFinder.getDataStore(params)
```

to GeoMesa

```
val params = Map("instanceId"->"accumulo",
                 "zookeepers"->"zoo1:2181",
                 "user"      ->"geomesa",
                 "password"  ->"supersecret",
                 "auths"     ->"",
                 "tableName" ->"gdelt")
val dataStore = DataStoreFinder.getDataStore(params)
```

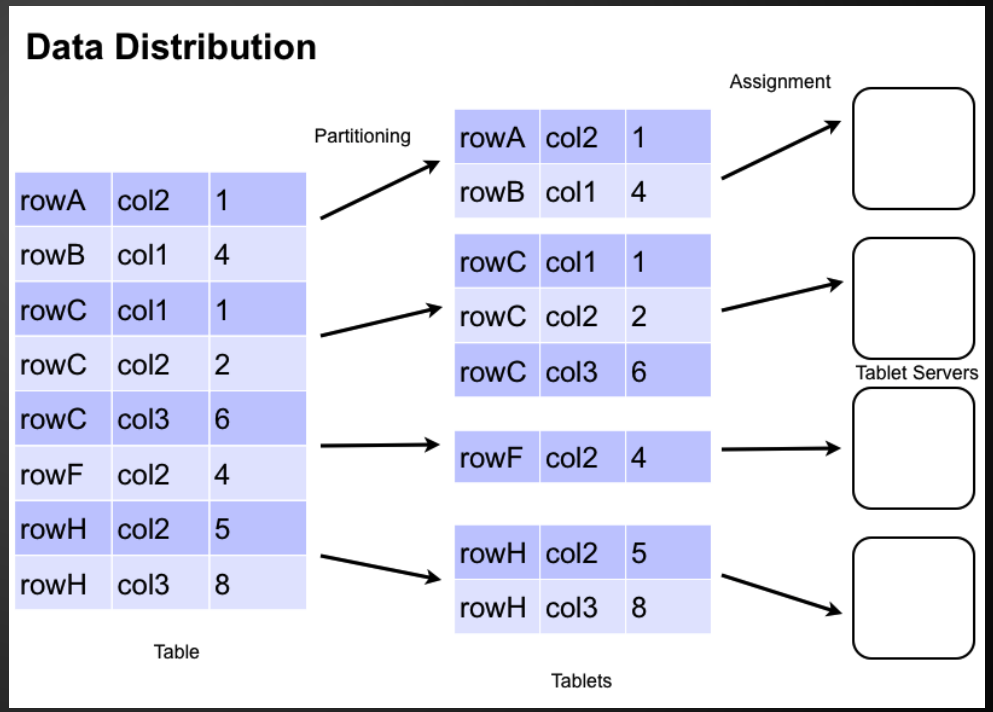
Accumulo: Distributed Database

- Distributed Key/Value Store
  - 5-tuple Key
- Lexicographic Ordering
- Schema-less...



Apache Accumulo

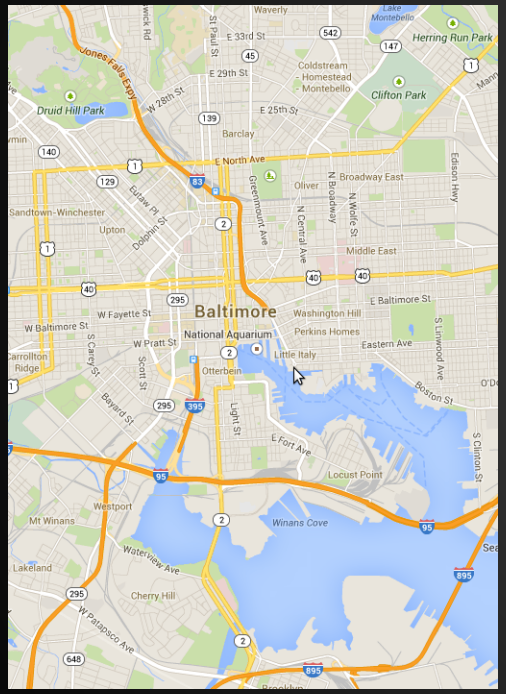
- Tables composed of Tablets
- Tablet Servers
- Partitioning Data
- Parallel Queries

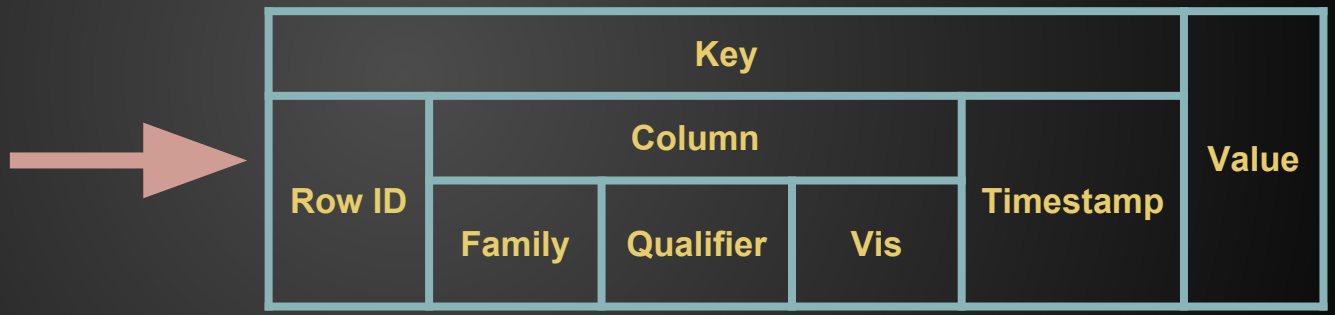


(E)CQL in Accumulo?

- BBOX
  - `bbox(geom, -74.1, 40.7, -74.2, 40.8)`
- INTERSECTS
  - `intersects(geom, Polygon((-74.1 40.7, -74.2 40.7, -74.2 40.8, -74.1 40.8, -74.1 40.7)))`
- DWITHIN
  - `dwithin(geom, Point(-74.00 40.71), 5000, meters)`
- Temporal Predicates
  - `dtg between 2014-01-02 AND 2014-01-06`

How will we do this?



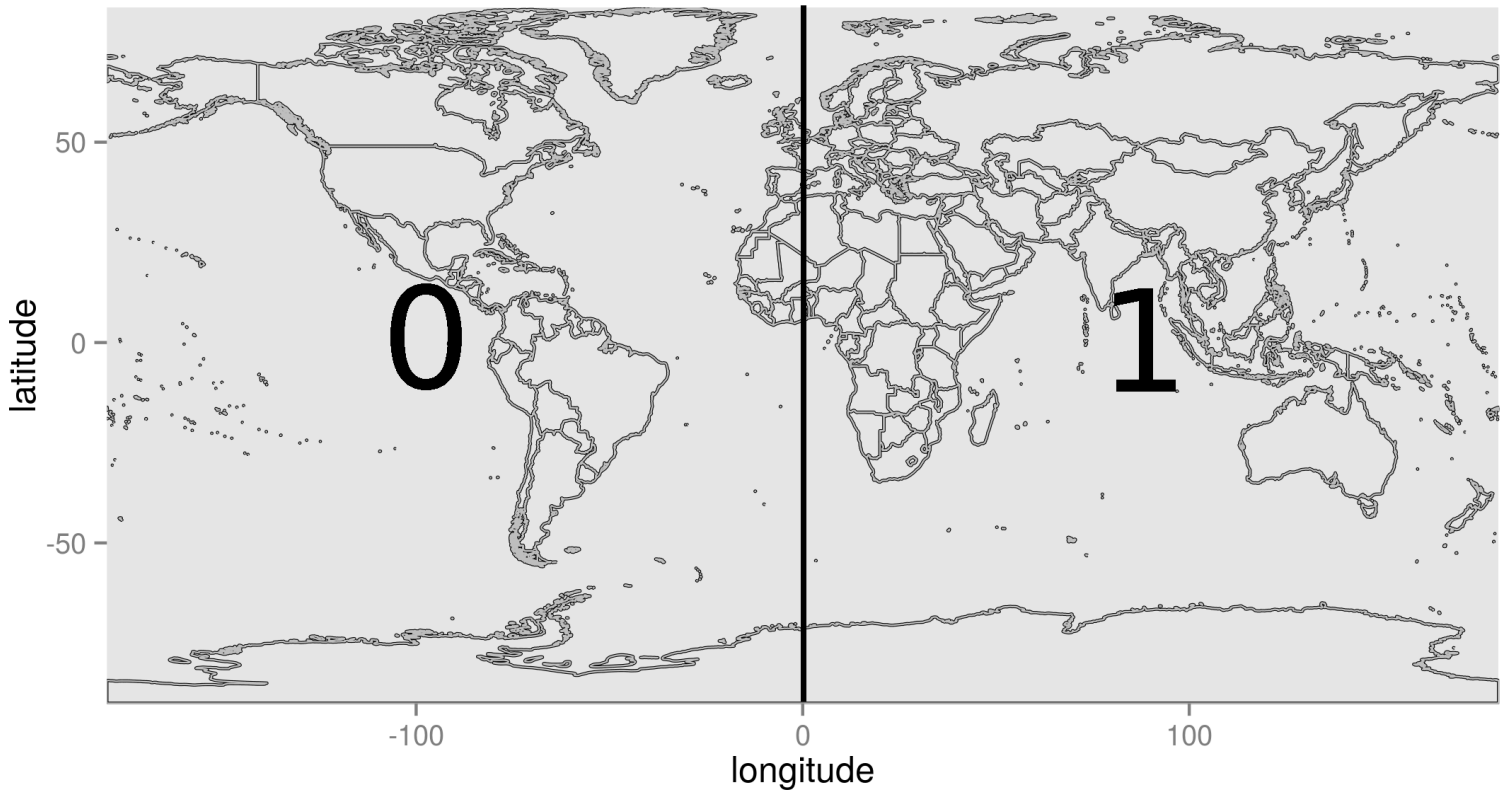


Space Filling Curves

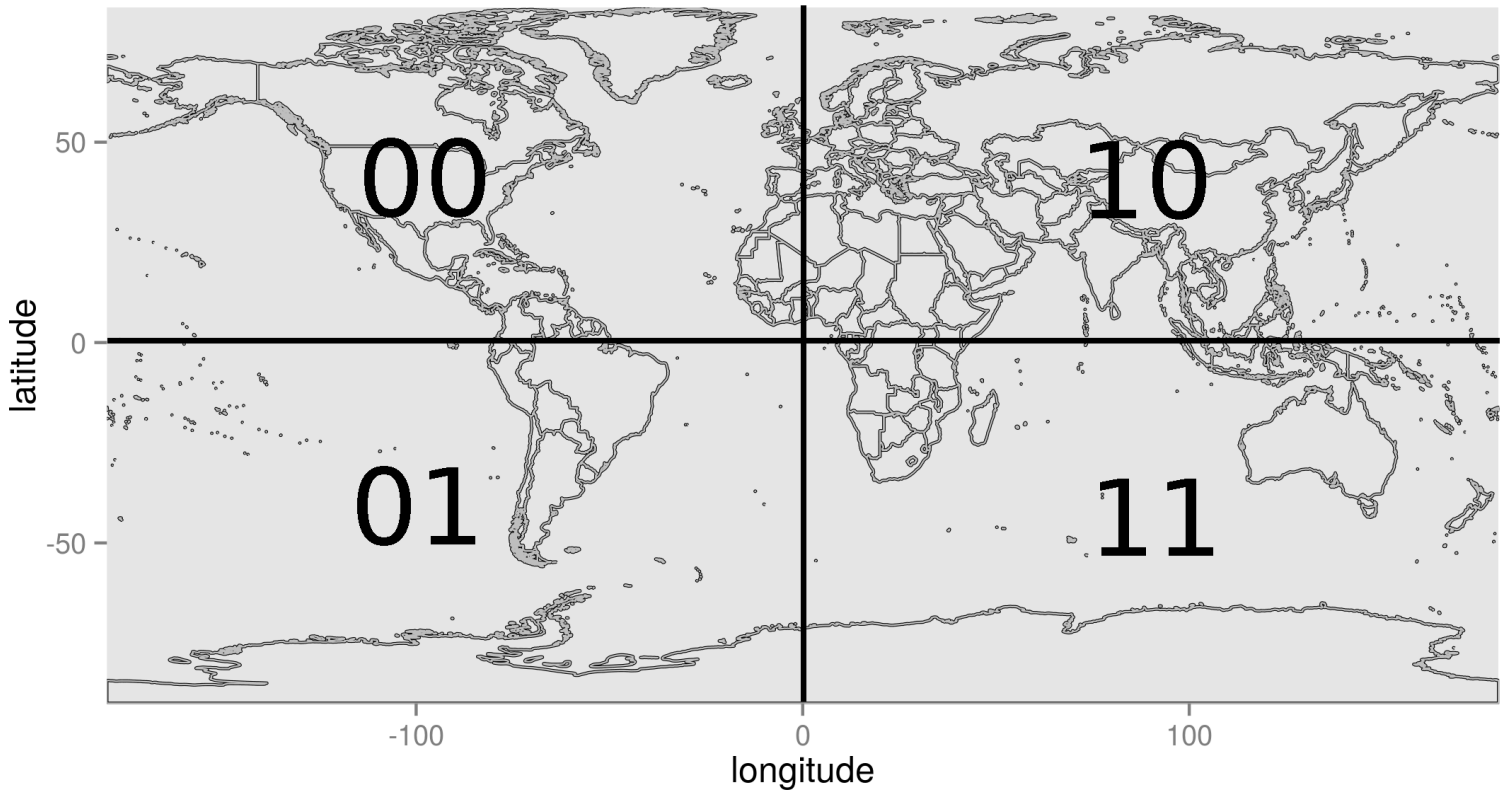
- Z-order Curve / GeoHash Algorithm
  - Encode multiple dimensions
- Interleaved Lon-Lat bits
  - Left to Right binary encoding
  - Alternate Splitting Rectangles

```
Left   -> 0
Right  -> 1
Top    -> 0
Bottom -> 1
```

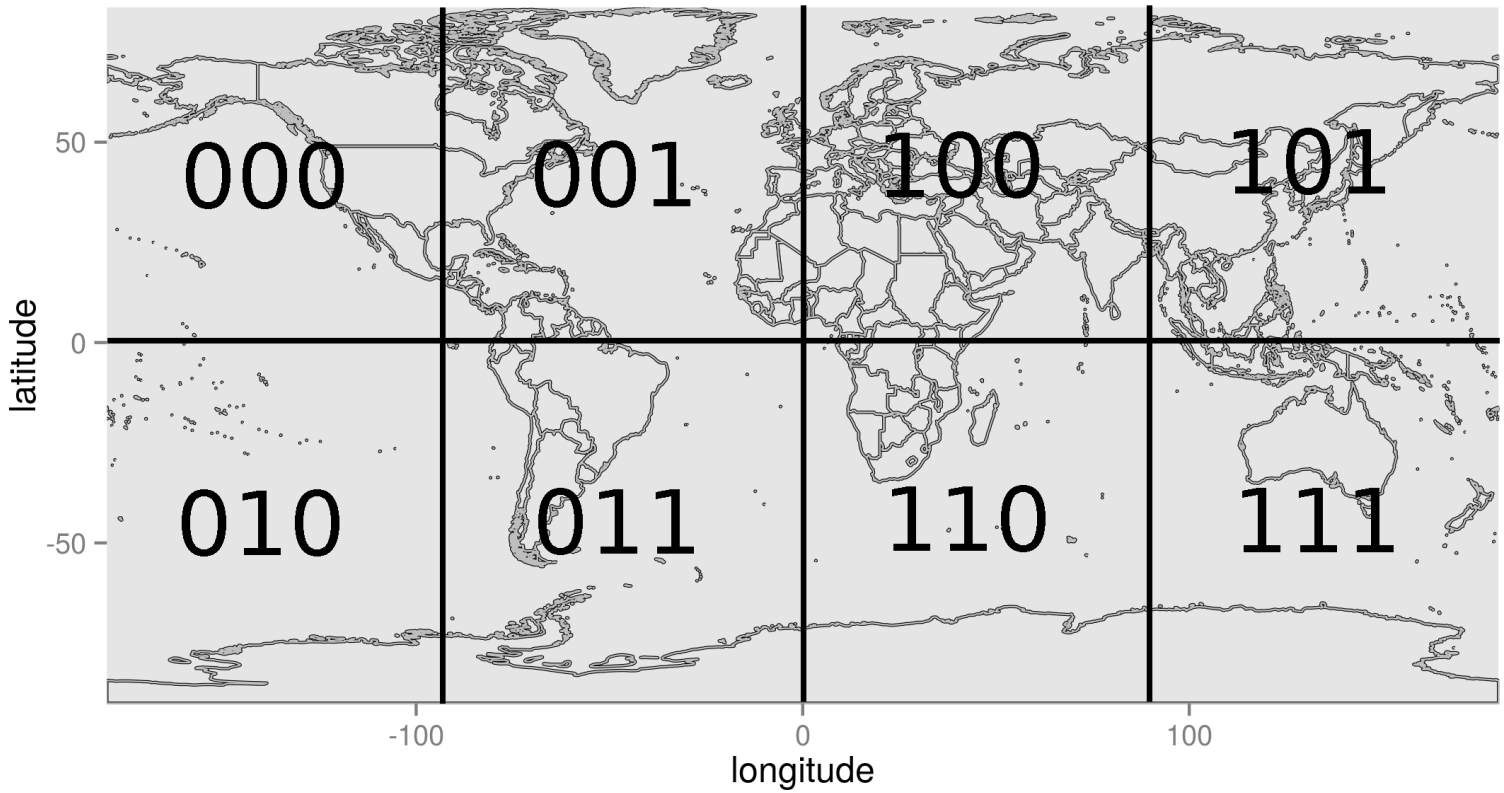
Linearize the Keyspace



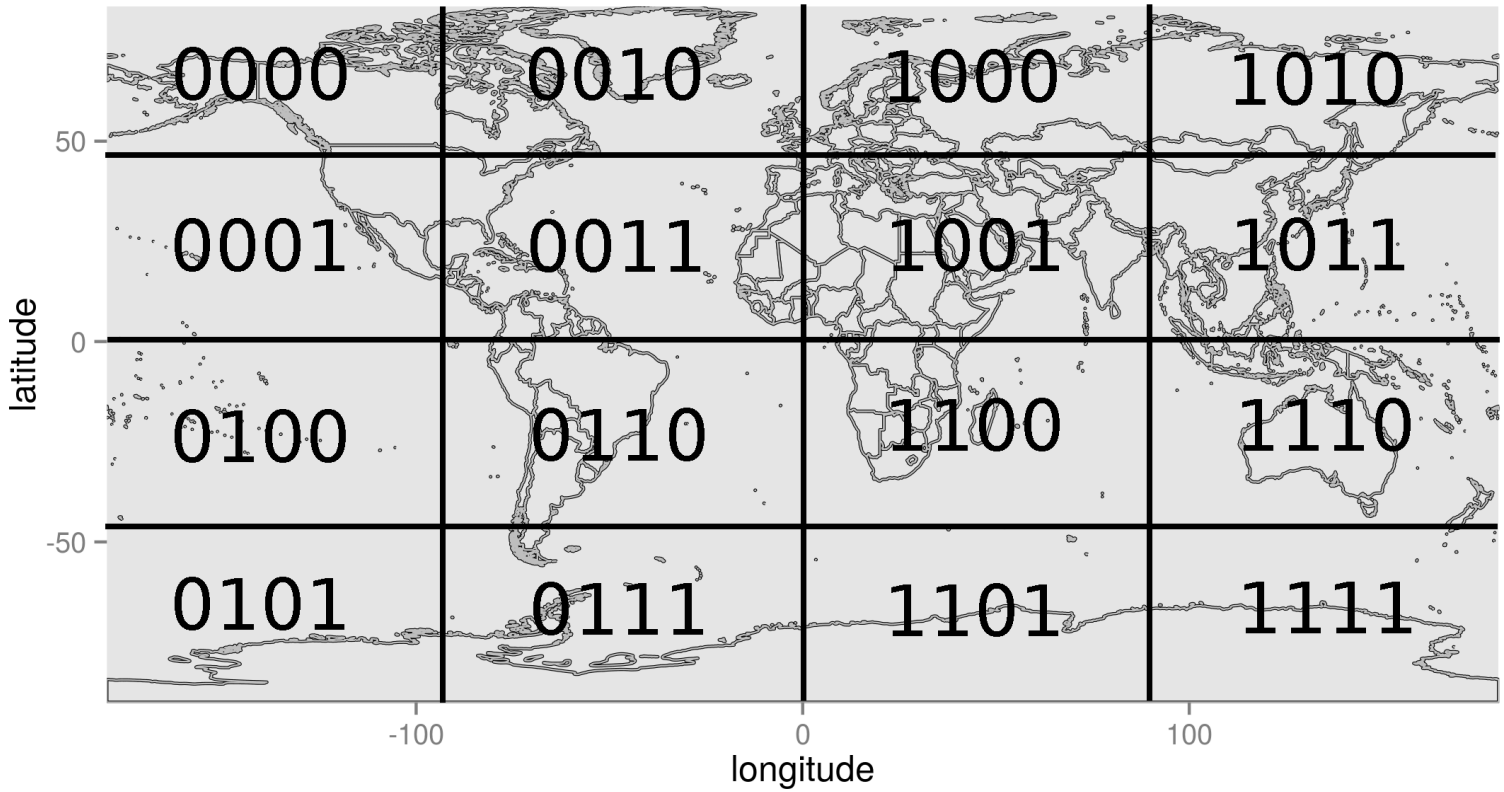
Linearize the Keyspace



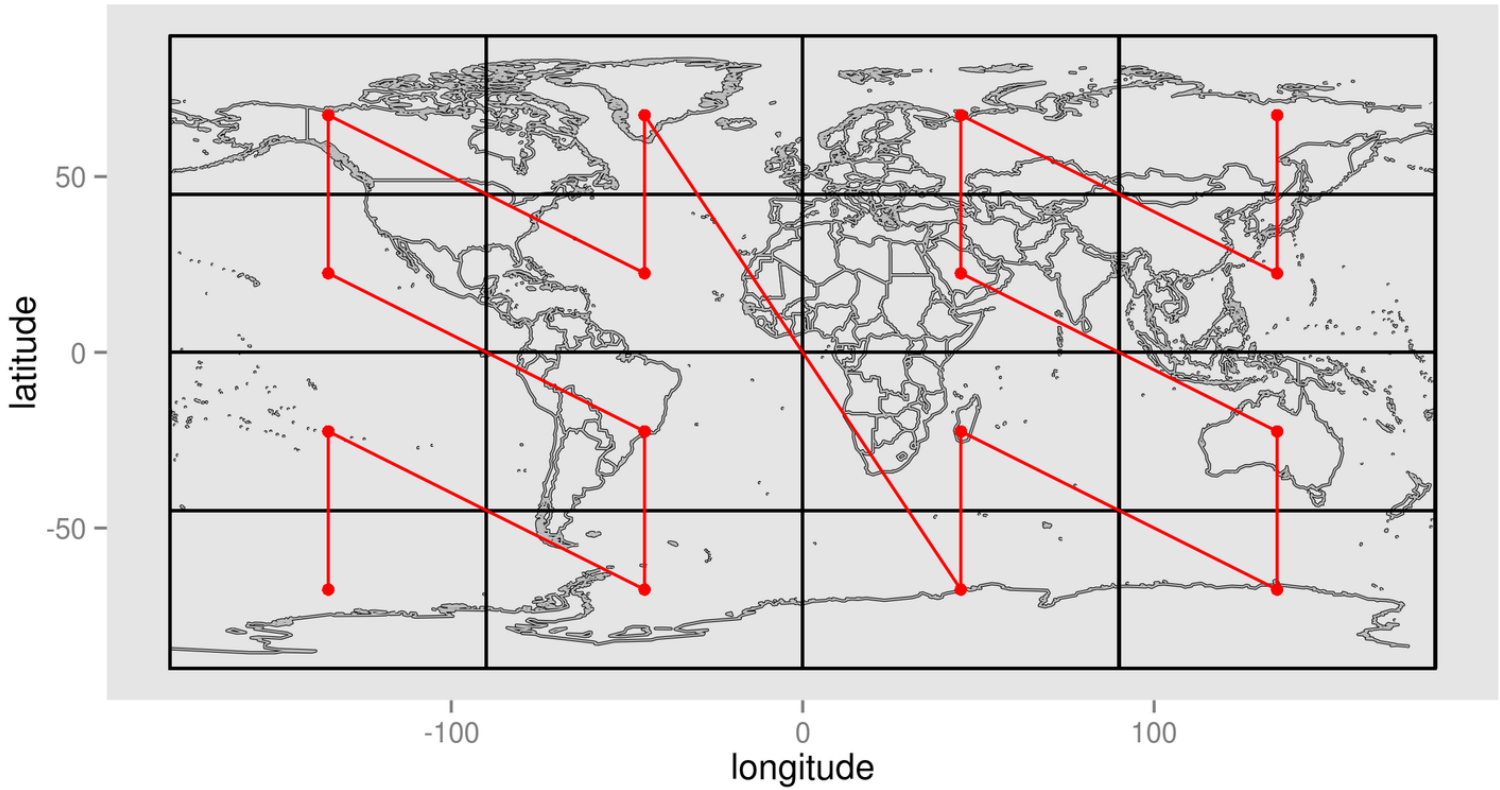
Linearize the Keyspace



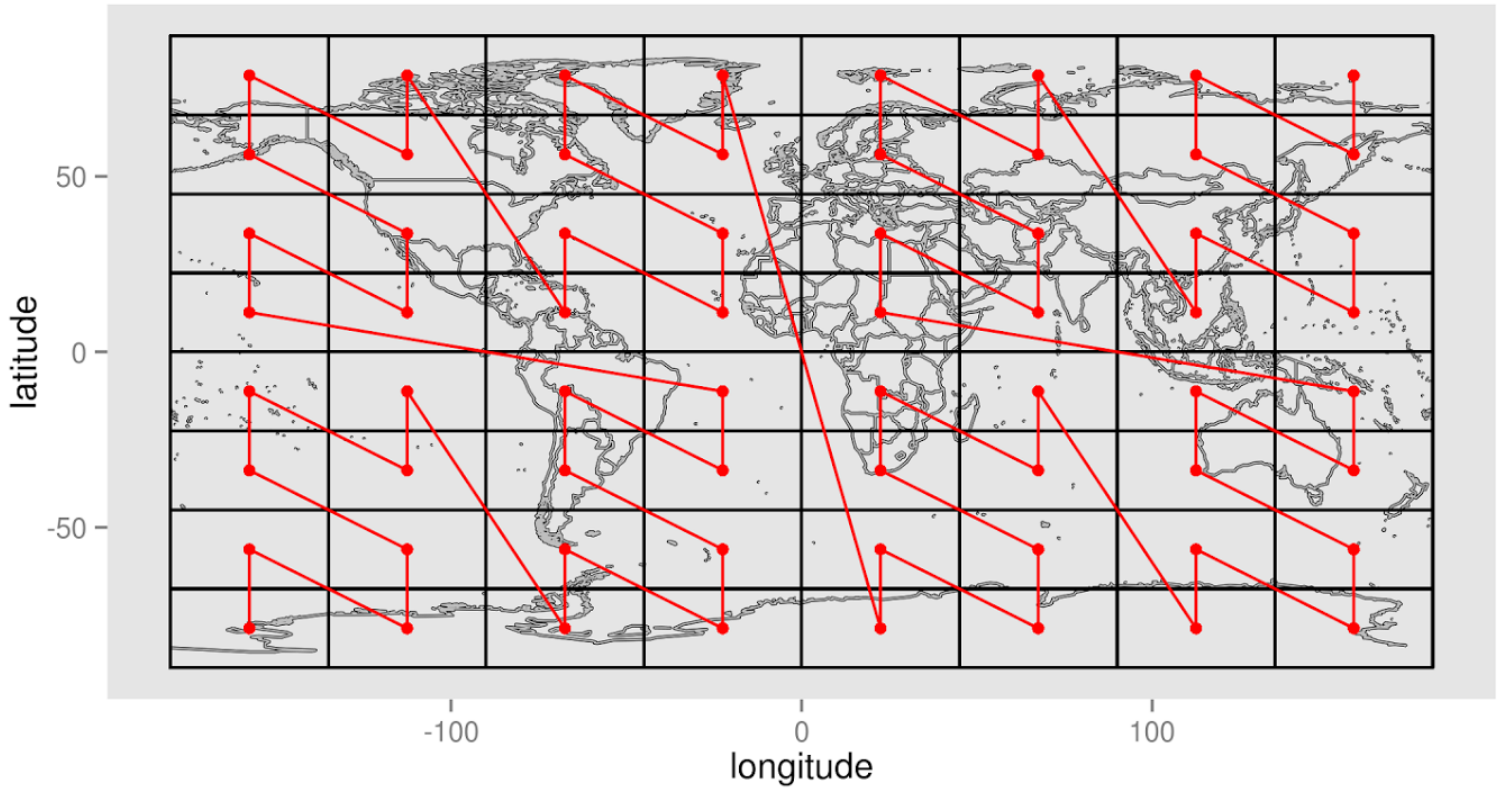
Linearize the Keyspace



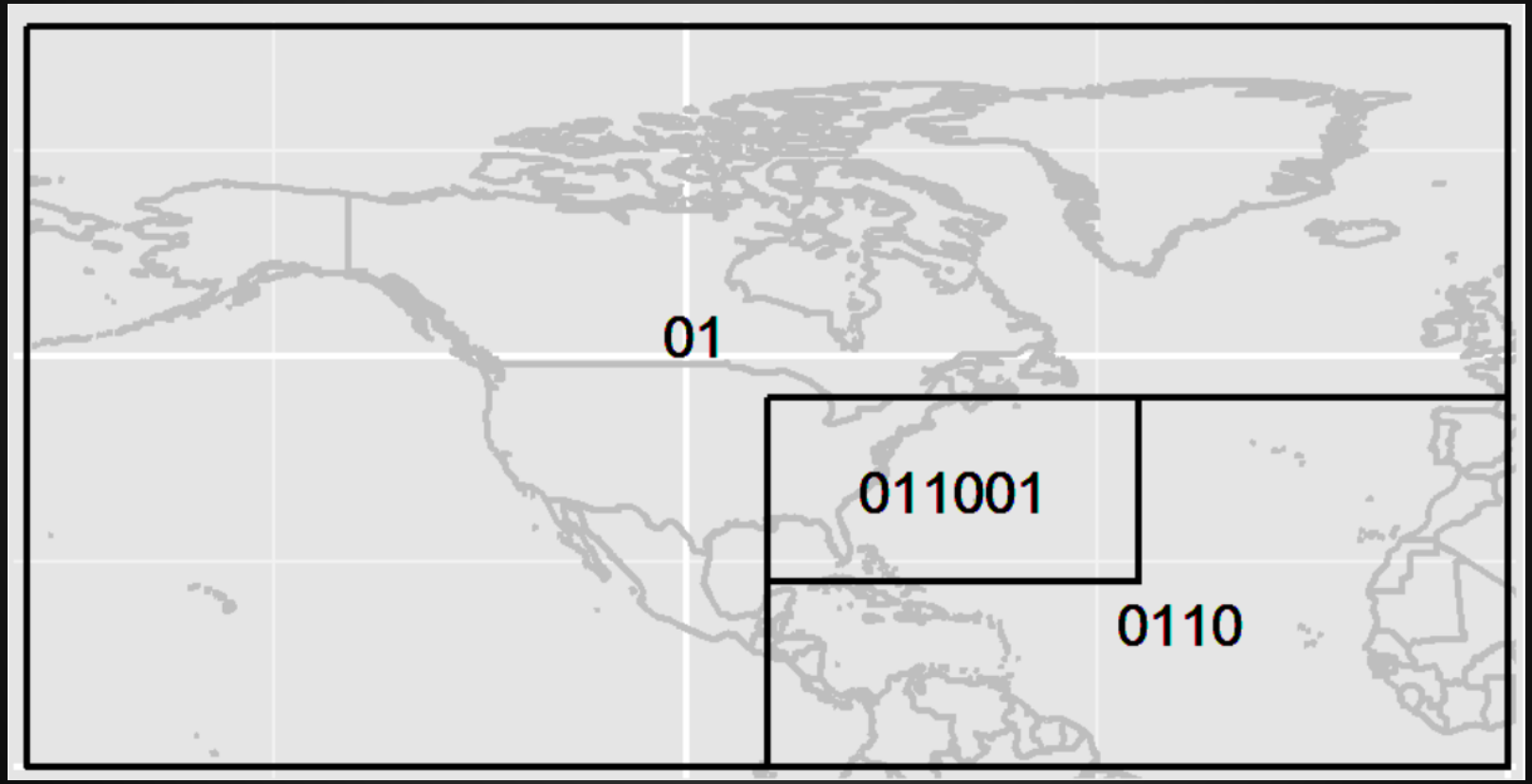
Linearize the Keyspace



Higher Precision



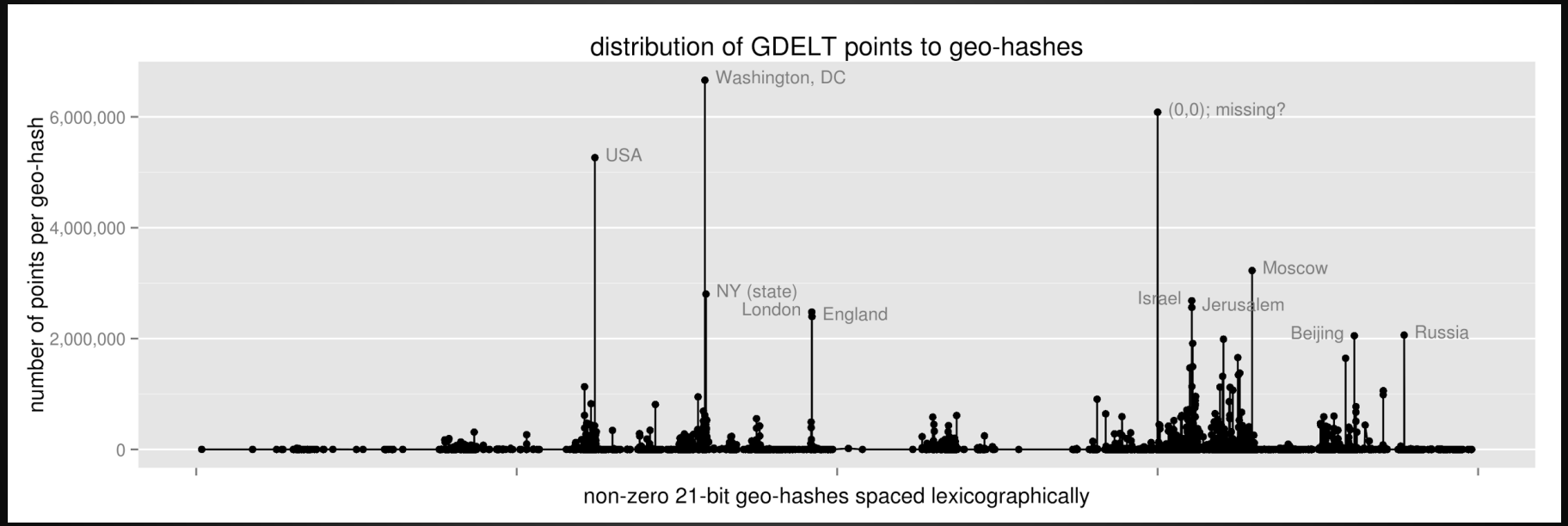
Locality and Containment



Locality and Containment

Baltimore (downtown)	-76.6167, 39.2833	dqcx2x33v
Fells Point	-76.5928, 39.2831	dqcx3p6bd
Federal Hill	-76.6100, 39.2789	dqcx2ww21
Dundalk	-76.5053, 39.2658	dqcx7h4ec
Columbia	-76.8569, 39.2036	dqcr50f98
Annapolis	-76.5012, 38.9729	dqctenv98
Philadelphia	-75.1667, 39.9500	dr4e389m8
New York	-74.0059, 40.7127	dr5regw89

Geospatial Hotspots



San Fran


Needles
Alcatraz Island
Treasure Island
9q8zh
9q8zj
9q8zn
9q8zp
Yerba Buena Island
Fort Point Rock
Helmet Rock
Little Mile Rock
Pyramid Rock
Black Head Rock
San Francisco, CA, USA
San Francisco
9q8yu
9q8yv
9q8yy
9q8yz
9q8ys
9q8yt
9q8yw
Image © 2014 TerraMetrics
© 2014 Google
Data CSUMB SFML, CA OPC
Double Rock
4.83 km

Sharding San Fran

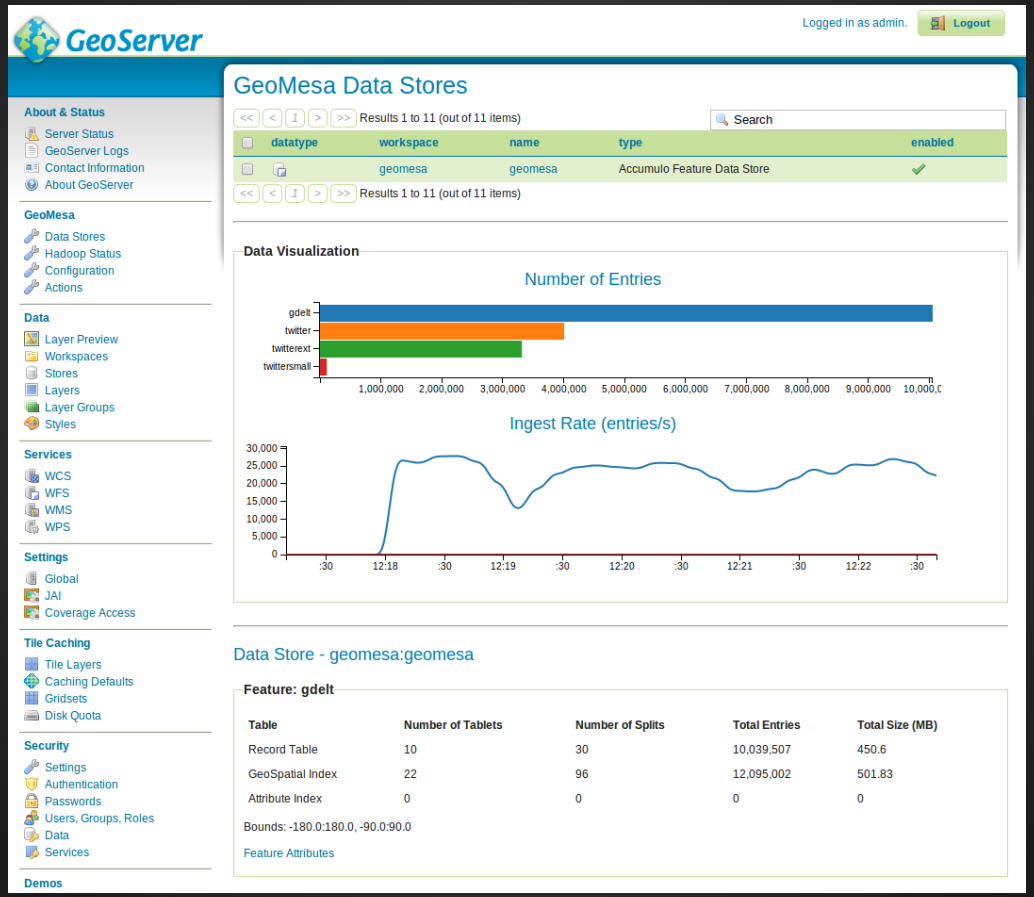
Key Structure

	Partition ID	L1 Spatial Hash	L1Temporal Hash	L2 Spatial Hash	L2 Temporal Hash	ID	Geometry	Time
Index	**Row Key**			**Column Family**		**Column Qualifier**	**Value**	
	001	9q8yy	201010	u	10	tweet.27963	POINT (-151,37)	20101009T23:47:07.234
	001	9q8yy	201010	v	15	tweet.12000	POINT (-151,37)	20101009T23:47:07.234
	001	gq89z	201112	x	25	tweet.32773	POINT (-151,37)	20101009T23:47:07.2343
	...	...	...	...	...	...	...	...
	001	t8vtw	200805	a	10	tweet.65432	POINT (-151,37)	20101009T23:47:07.234
Data	**Row Key**			**Column Family**		**Column Qualifier**	**Value**	
	001	9q8yy	201010	tweet.27963		#OGC, #BigData	id=xyz|green|48	
	001	9q8yy	201010	tweet.12000		#Accumulo	id=abc|yellow|12	
	001	gq89z	201112	tweet.32773		#BigTable	state=red	
	...	...	...	...		...	...	
	001	t8vtw	200805	tweet.65432		#Geohash	value=conference	

Building an Architecture

- GeoServer Integration
- Security
- Command Line Tools
- Indexing, Querying Planning
- Compute + Analytics

GeoServer Integration



Command Line Tools

- Feature Ingest
- Query & Export
  - Shape Files, csv, tsv
- Data Management
  - List, Describe, Add, Remove
- Explain Queries

Query Planning and Indexing

- SpatioTemporal Index
- Secondary Index on Attributes

- Queries:
  - How do presidents' word choices change based on the state they're visiting?
  - Where are people eating in Baltimore before Orioles games?

SpatioTemporal Index

- Political Language

```
bbox("United States") AND
    time between 2000-2014
```

- O's Game

```
dwithin("Camden Yards", 3, "km") AND
    text like %food% AND
    (time between t1 AND t2 OR
    time between t3 and t4, ...)
```

Attribute Index

- Political Language

```
bbox(“United States”) AND
    time between 2000-2014
    user_id in (1, 2, 3, 4, 5, etc)
```

- O’s Game
  - What are all the user id’s?
  - Full table scan

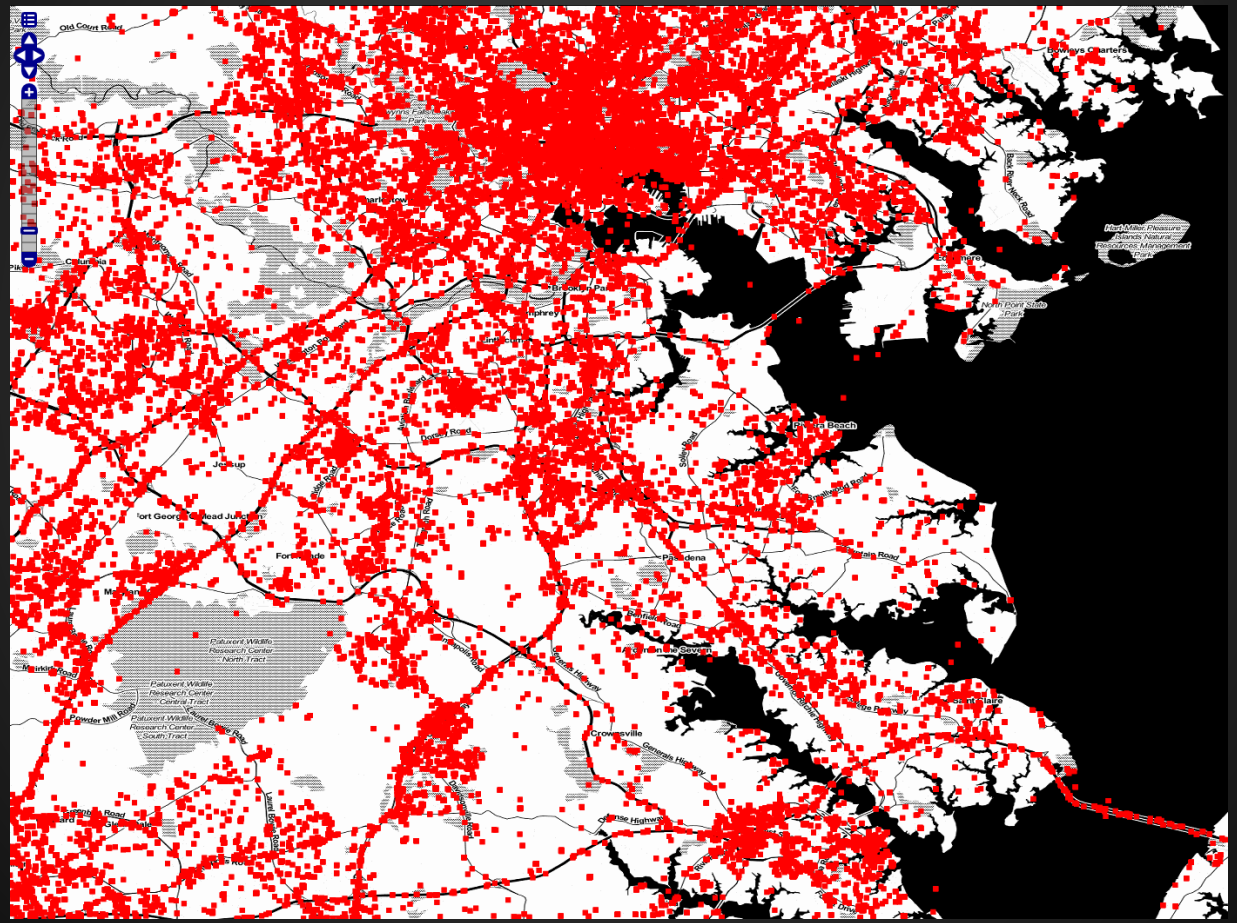
Query Planning

- Table Statistics
- Query Statistics
- Attribute Cardinality
- Iterators
  - SpatioTemporal Intersecting
  - Attribute Filtering
  - ST Index Only
  - Attribute Index

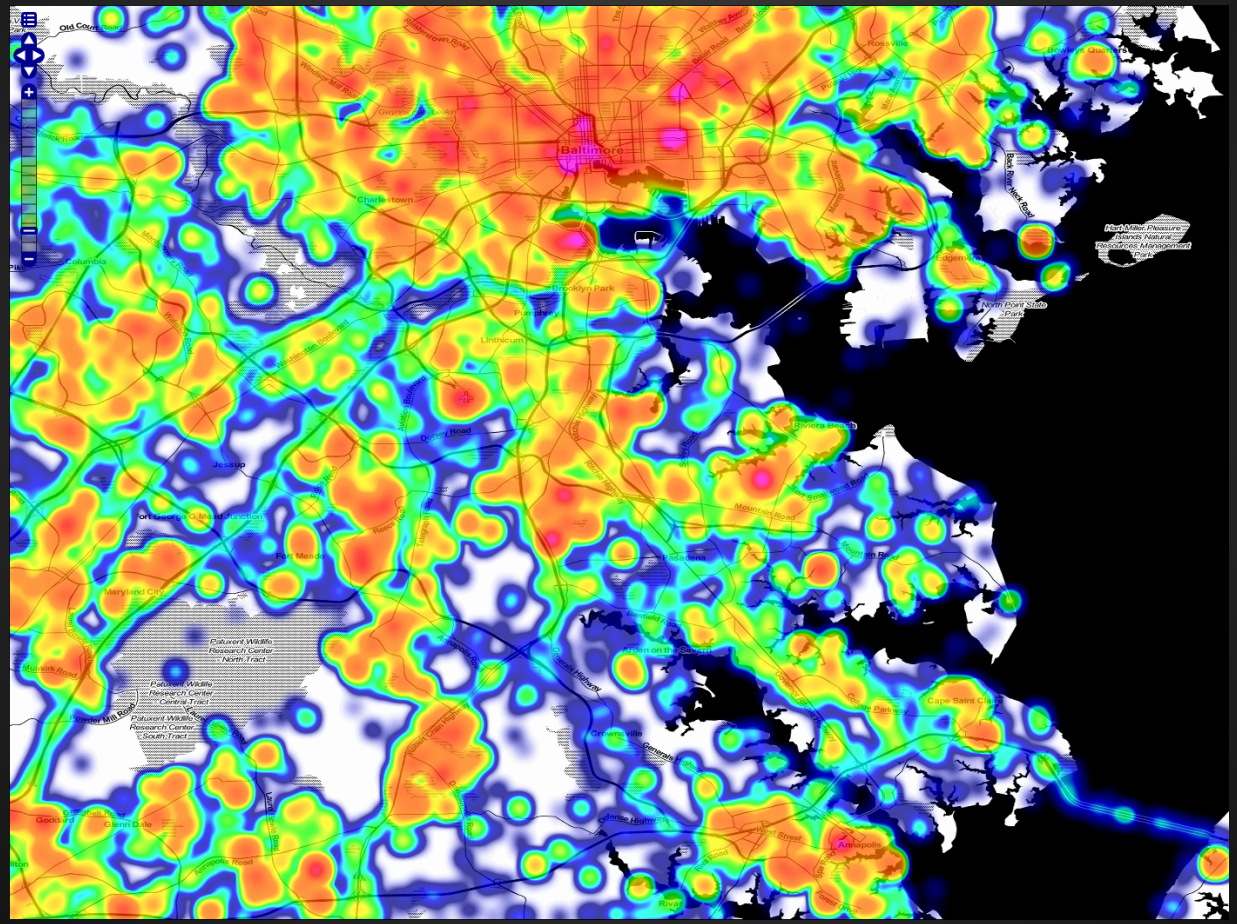
Compute & Analytics

- Queries & Mapping
- Accumulo Iterators
- GeoServer WPS Processes
  - AaaS (Analytics as a Service)
- Scalding
- Spark
- Data Export

CQL Queries


Old Court Road
Pasadena
Crownsville
Patuxent Wildlife Research Center North Tract
Patuxent Wildlife Research Center Central Tract
Patuxent Wildlife Research Center South Tract
Powder Mill Road
Generals Highway
Patuxent Road
Hart-Miller Pleasure Islands Natural Resources Management Area
North Point State Park

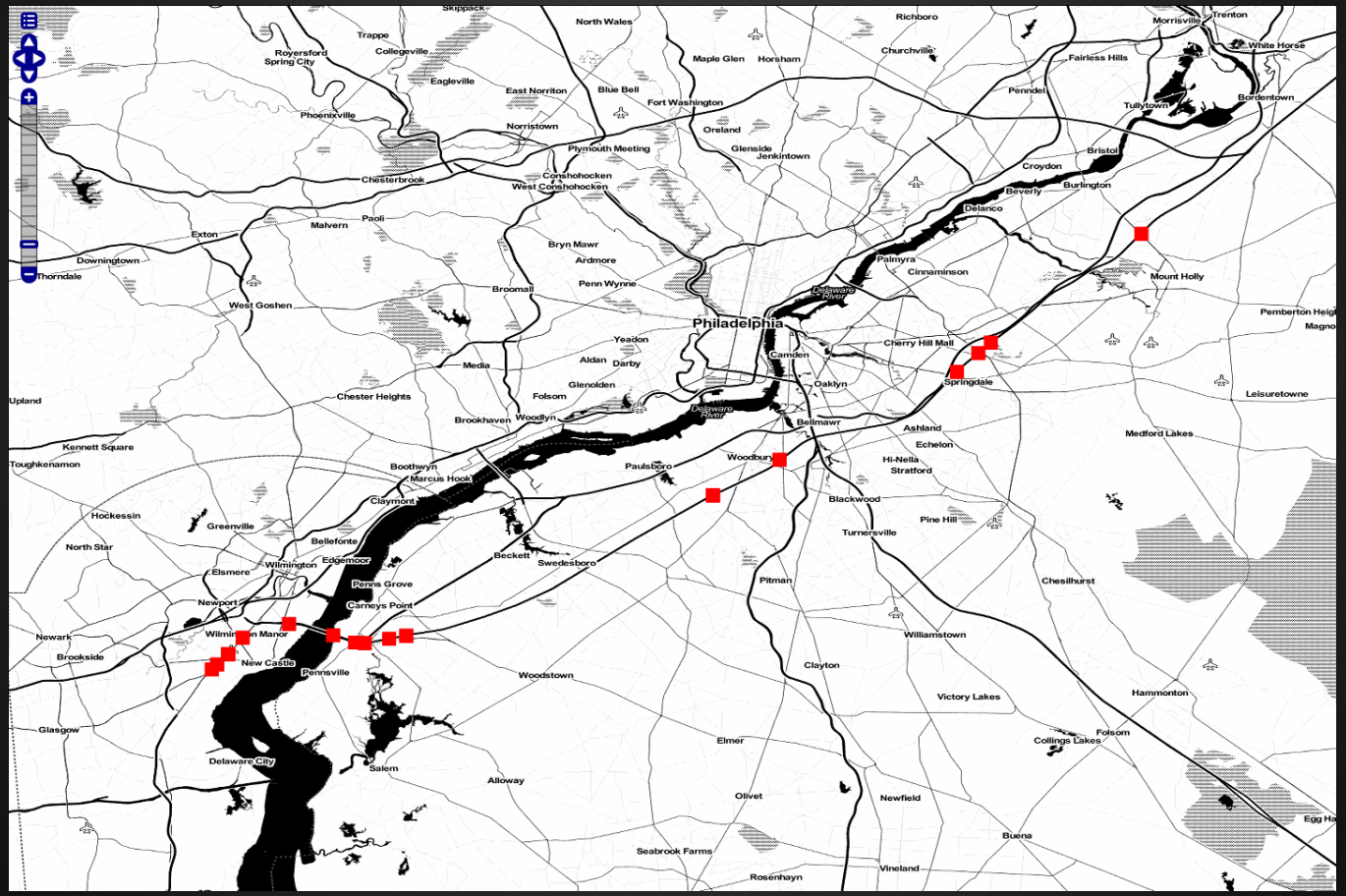
Parallel Density Calculations



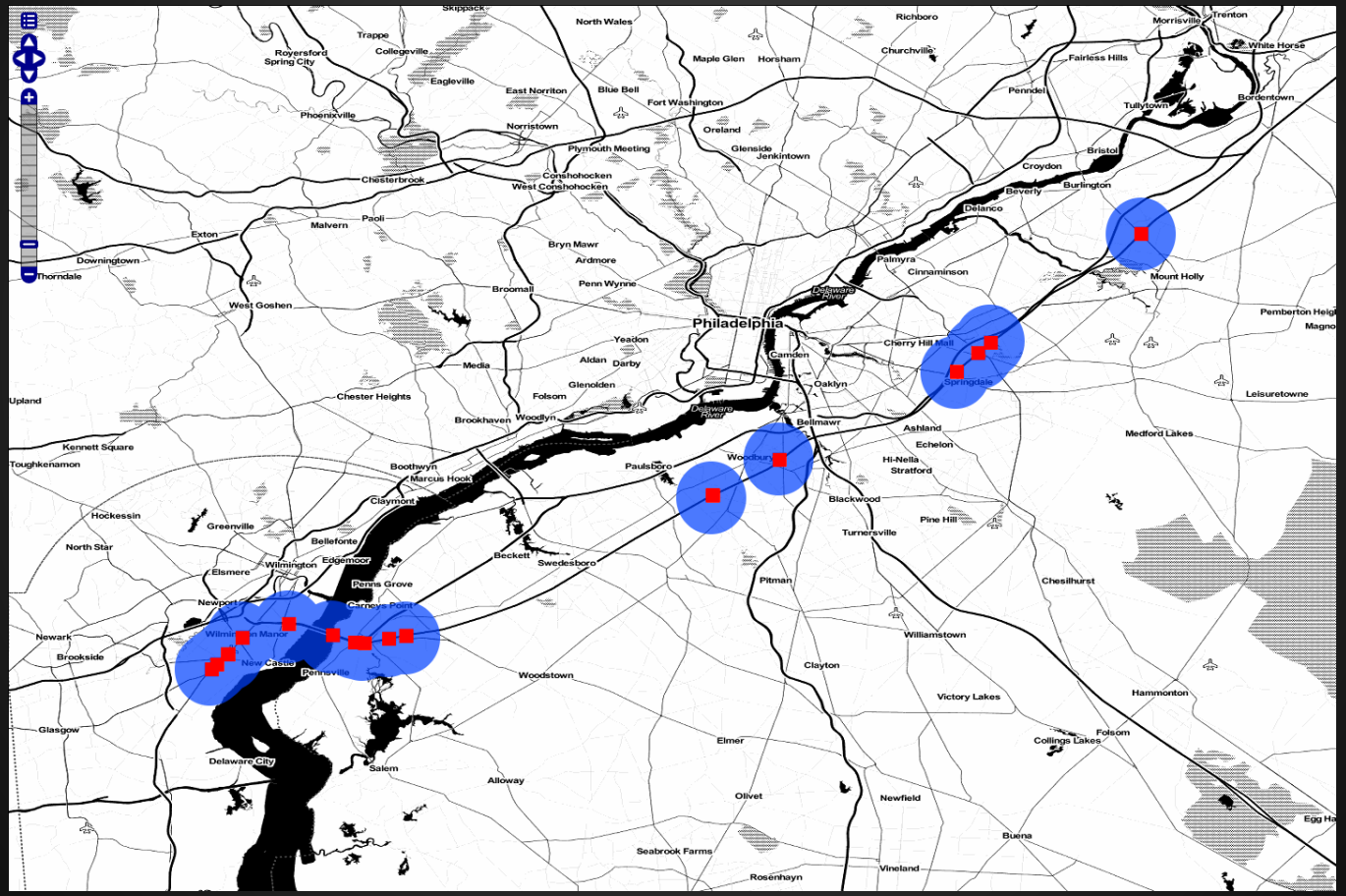
Time-Interpolated Queries

- Spatio-temporal gradient
  - Interpolate space (max speed, max time)
  - Interpolate time
- Co-Travelers
- Alias detection
  - Clickstream by phone IP addr
- Candidate Interactions
  - Subsequent scoring
  - Motion Models

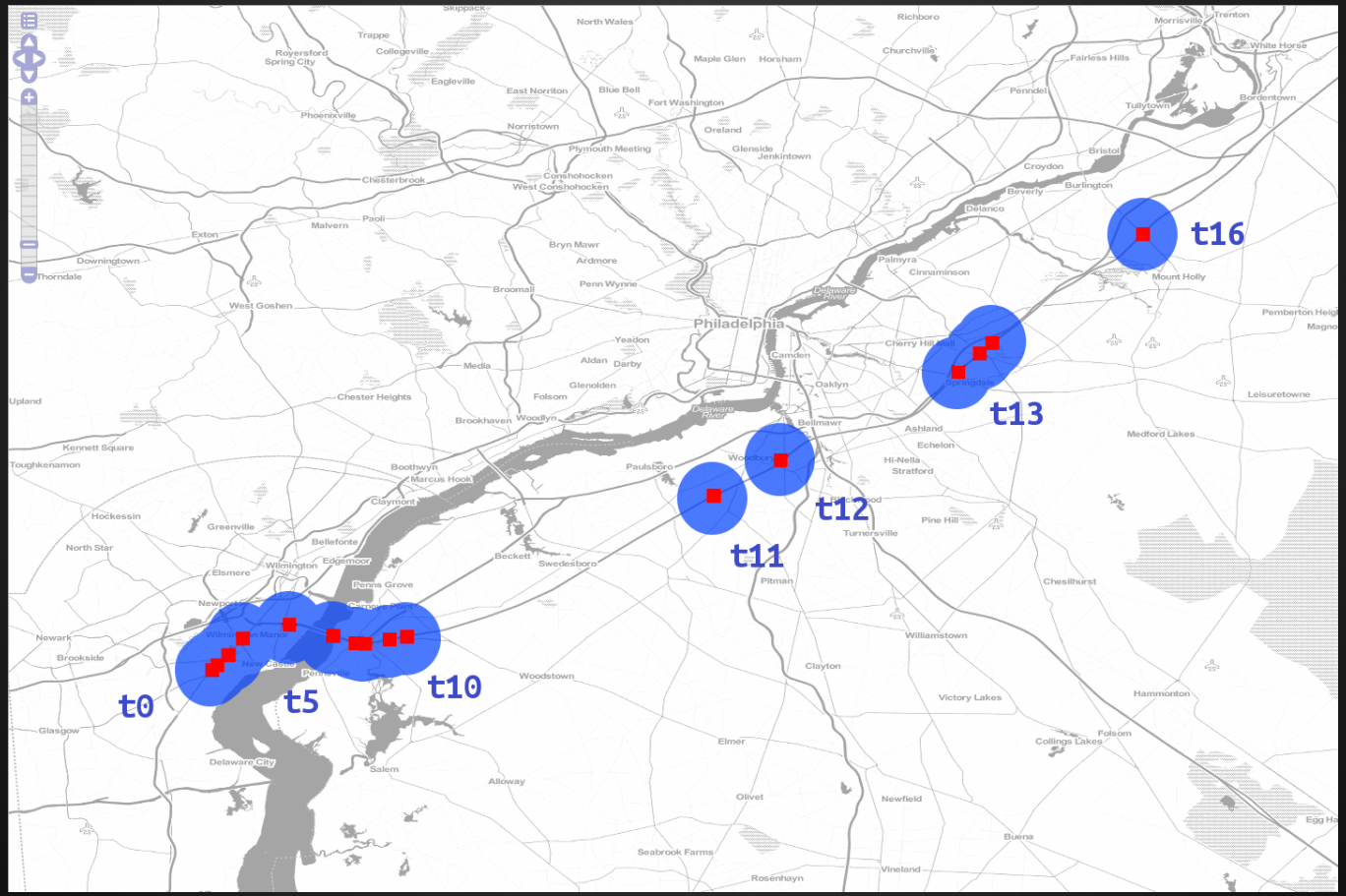
Tweeting the NJ Turnpike



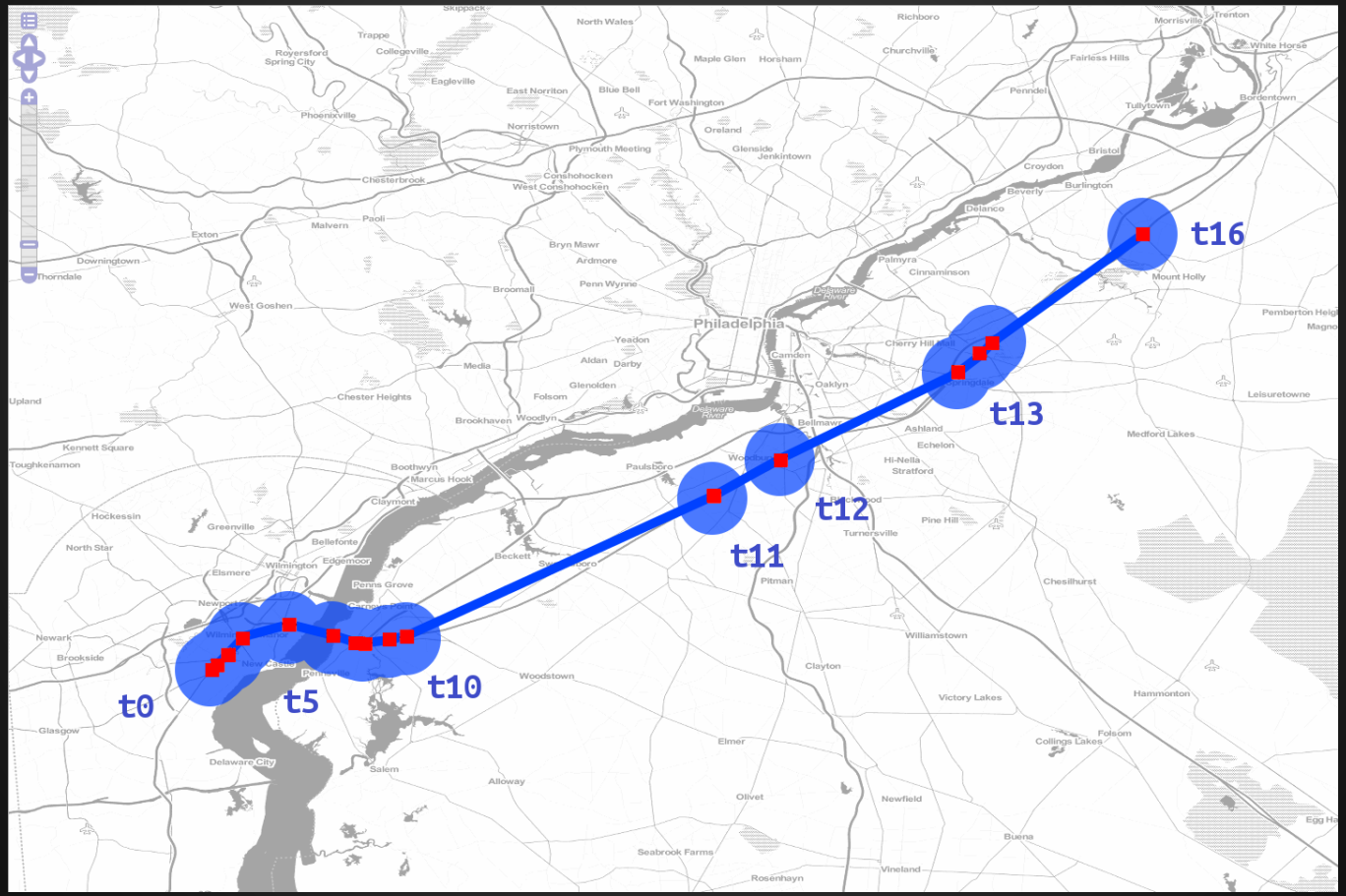
Interpolating Space-Time



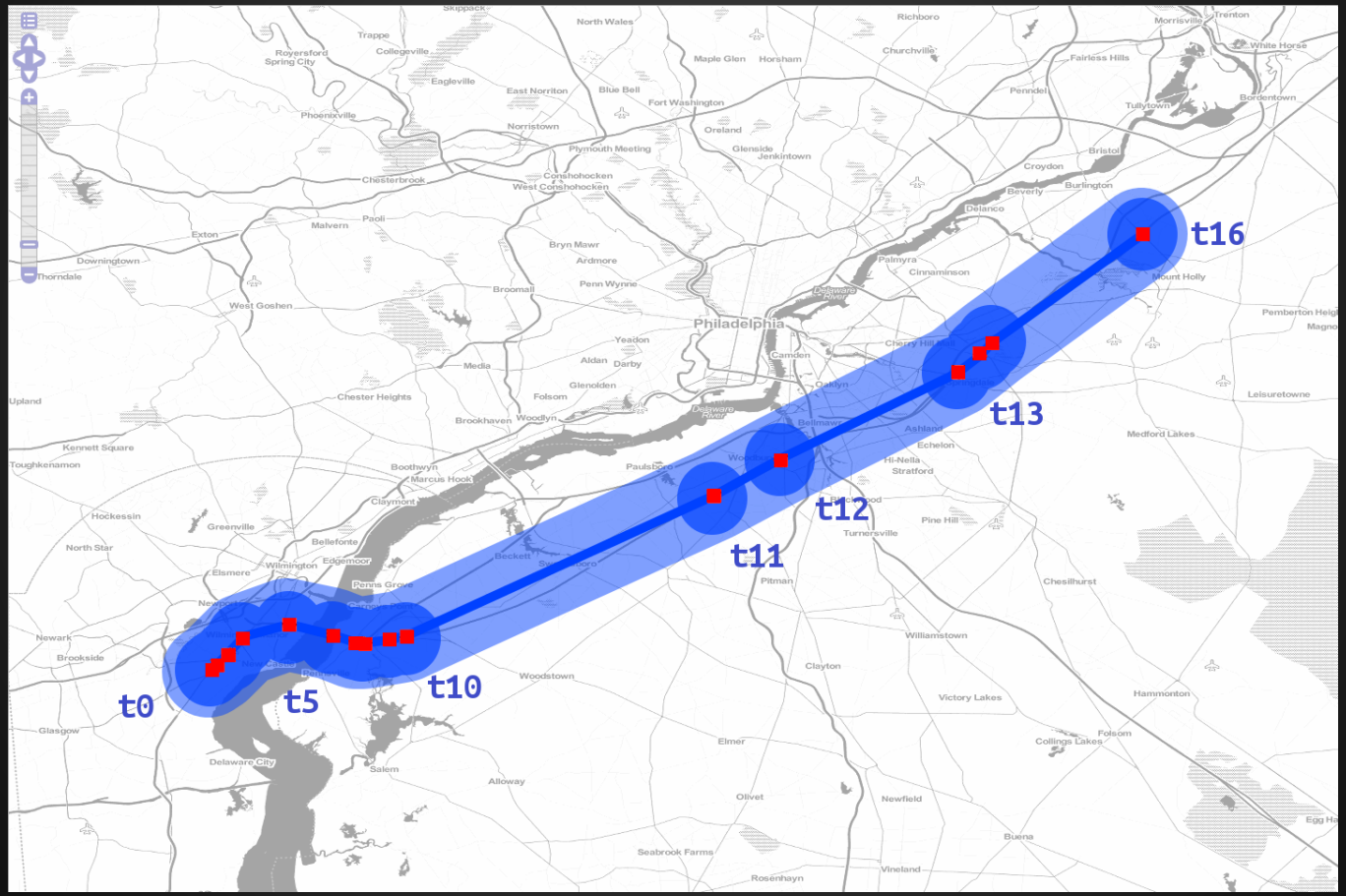
Interpolating Space-Time



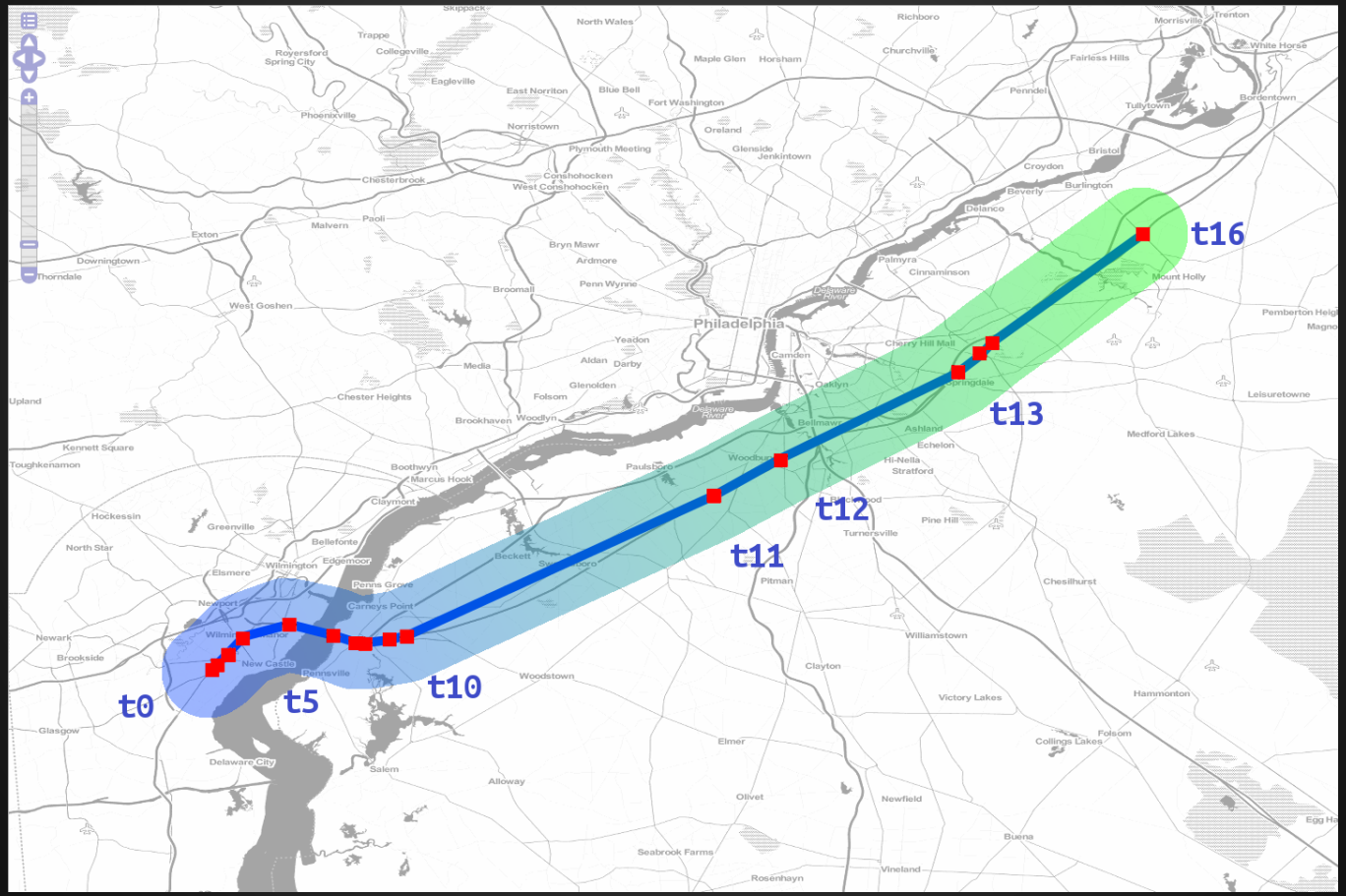
Interpolating Space-Time



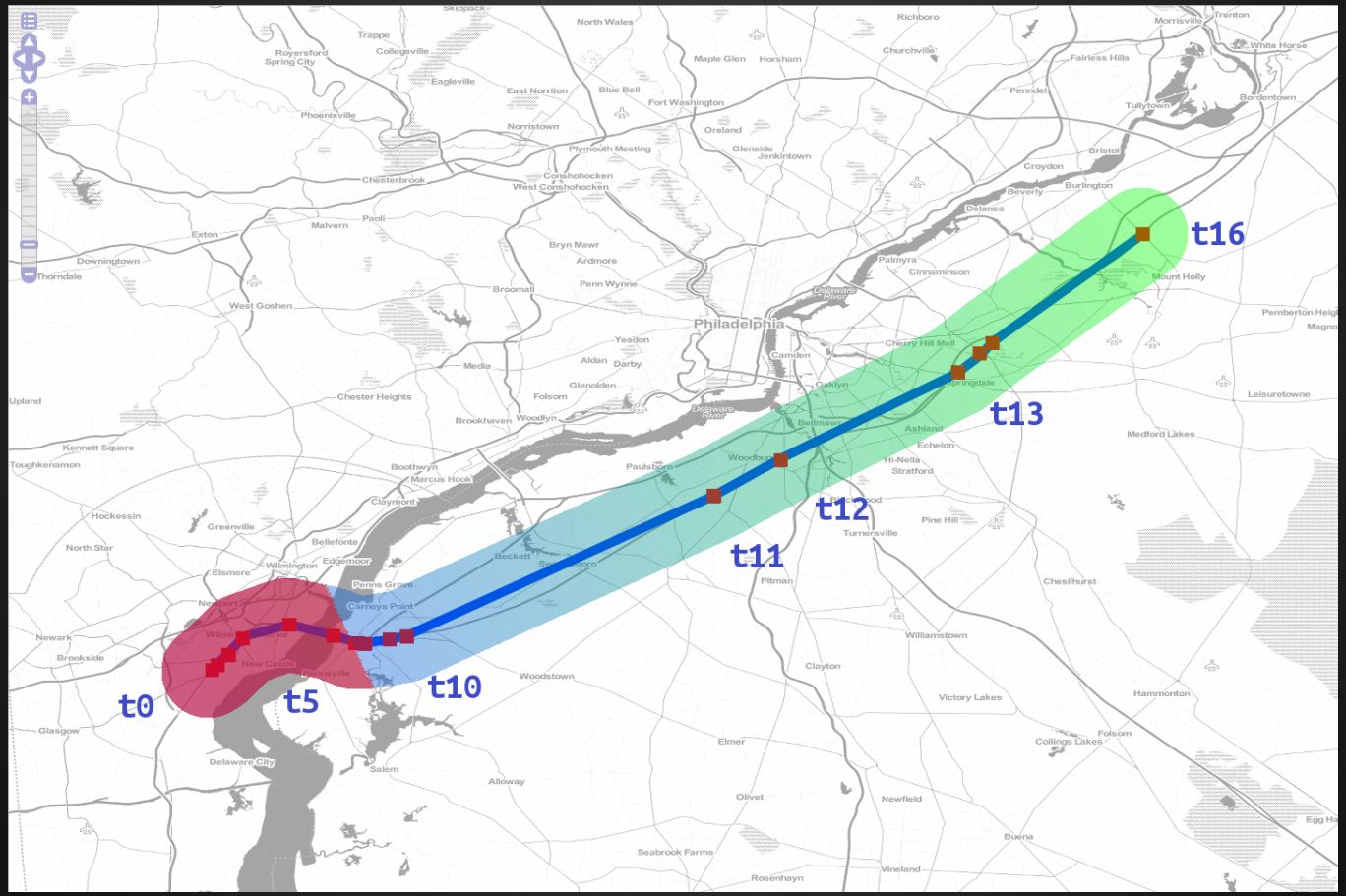
Interpolating Space-Time



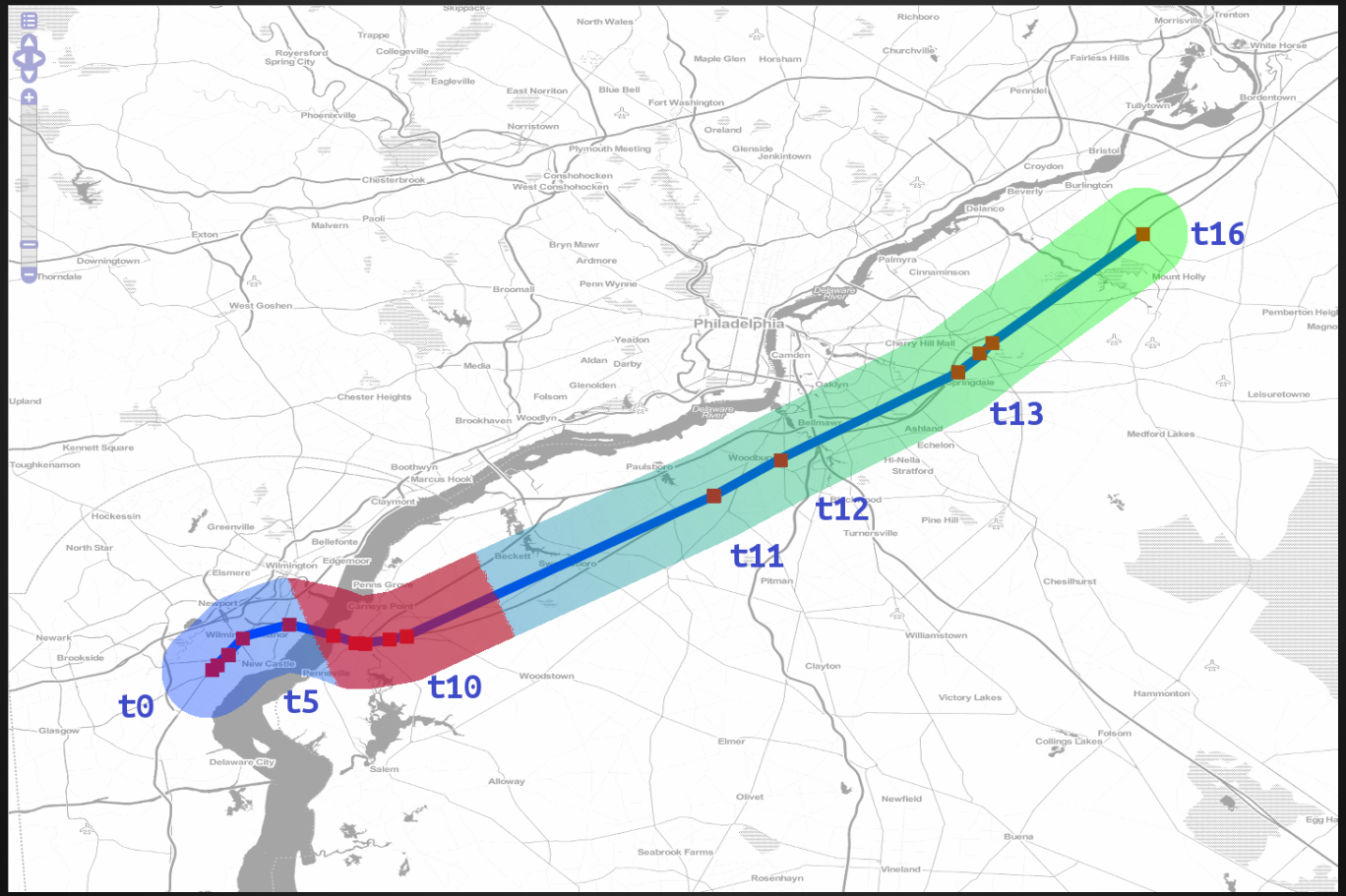
Interpolating Space-Time



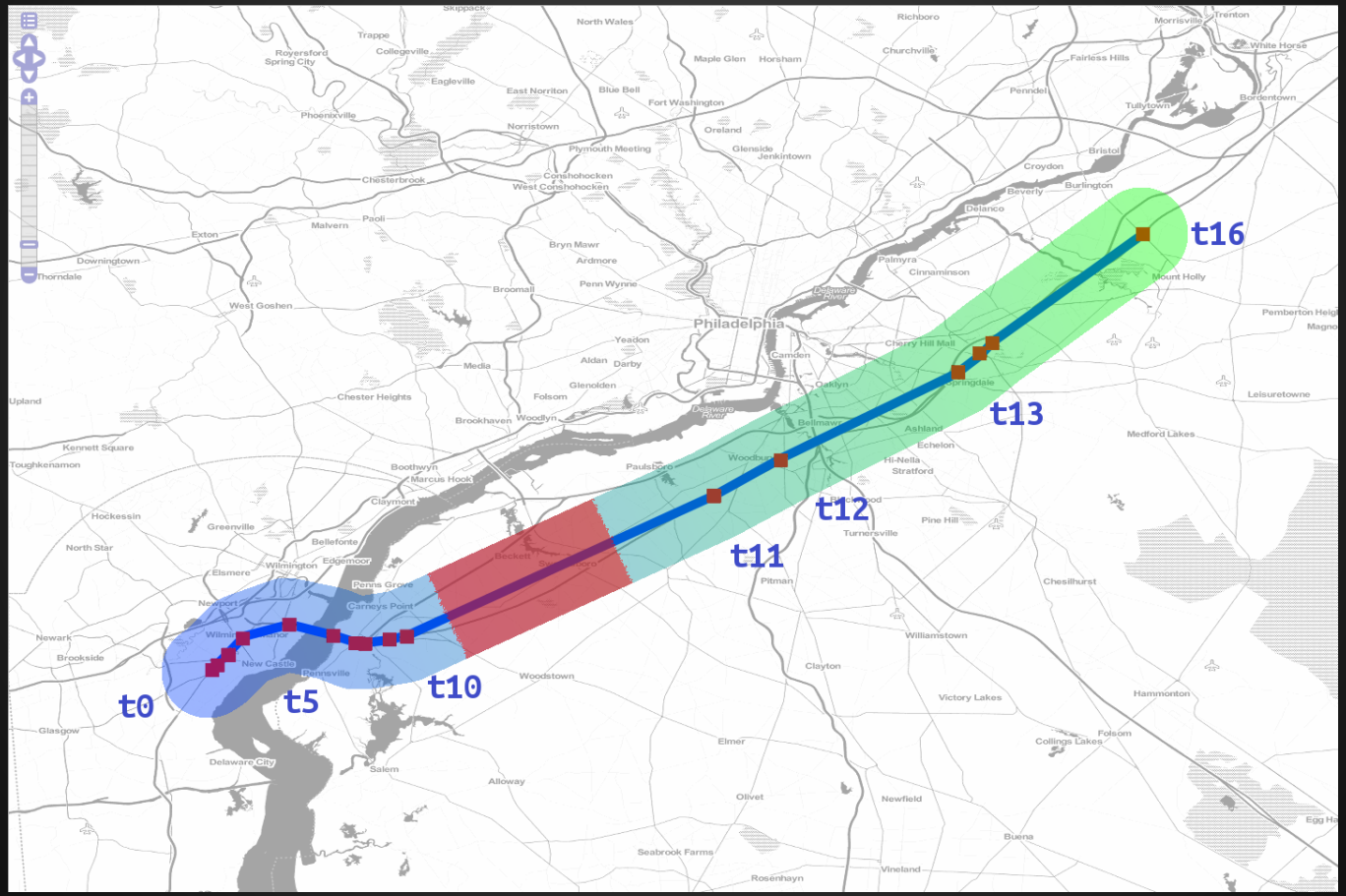
Query Planning



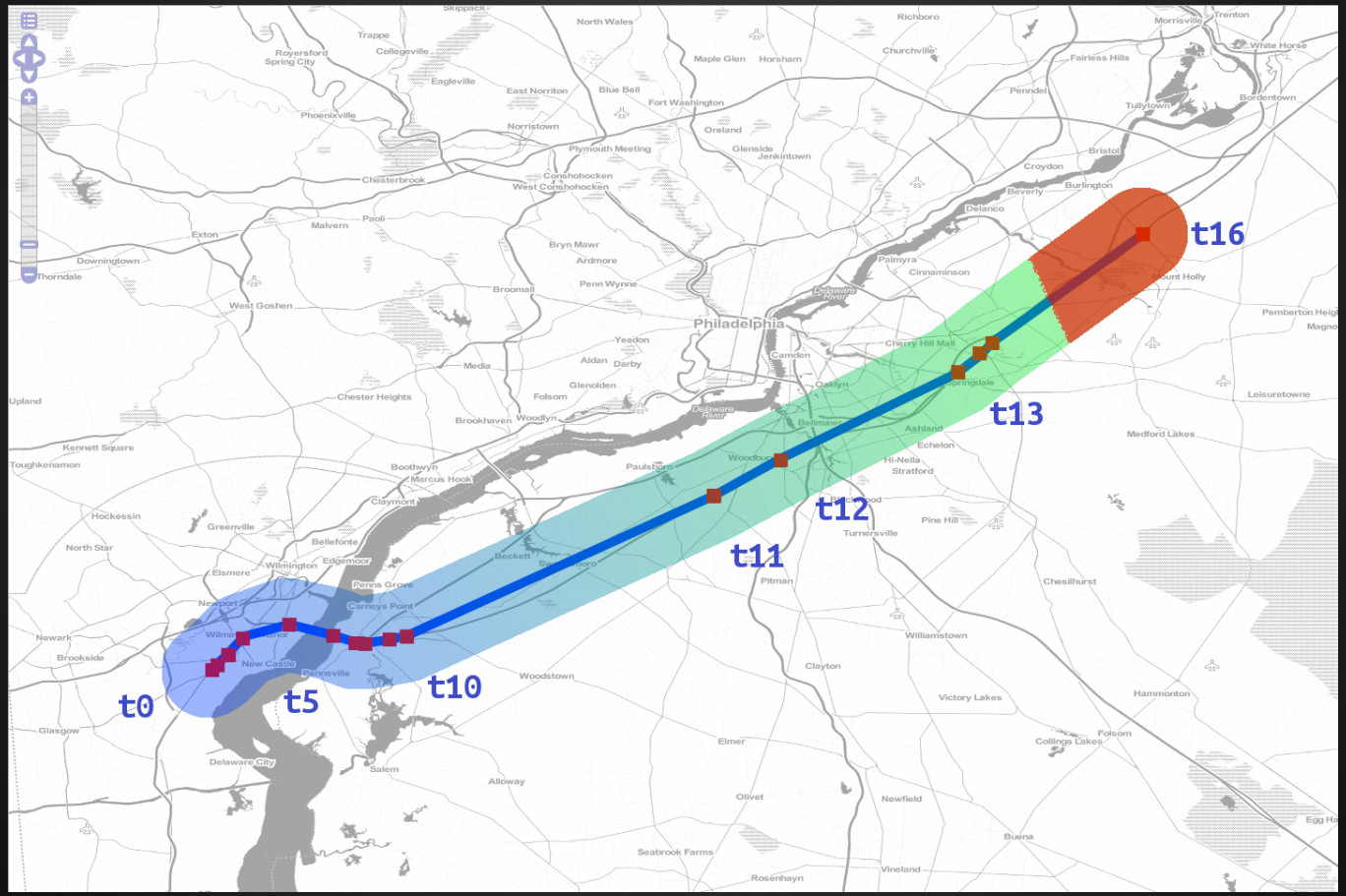
Query Planning



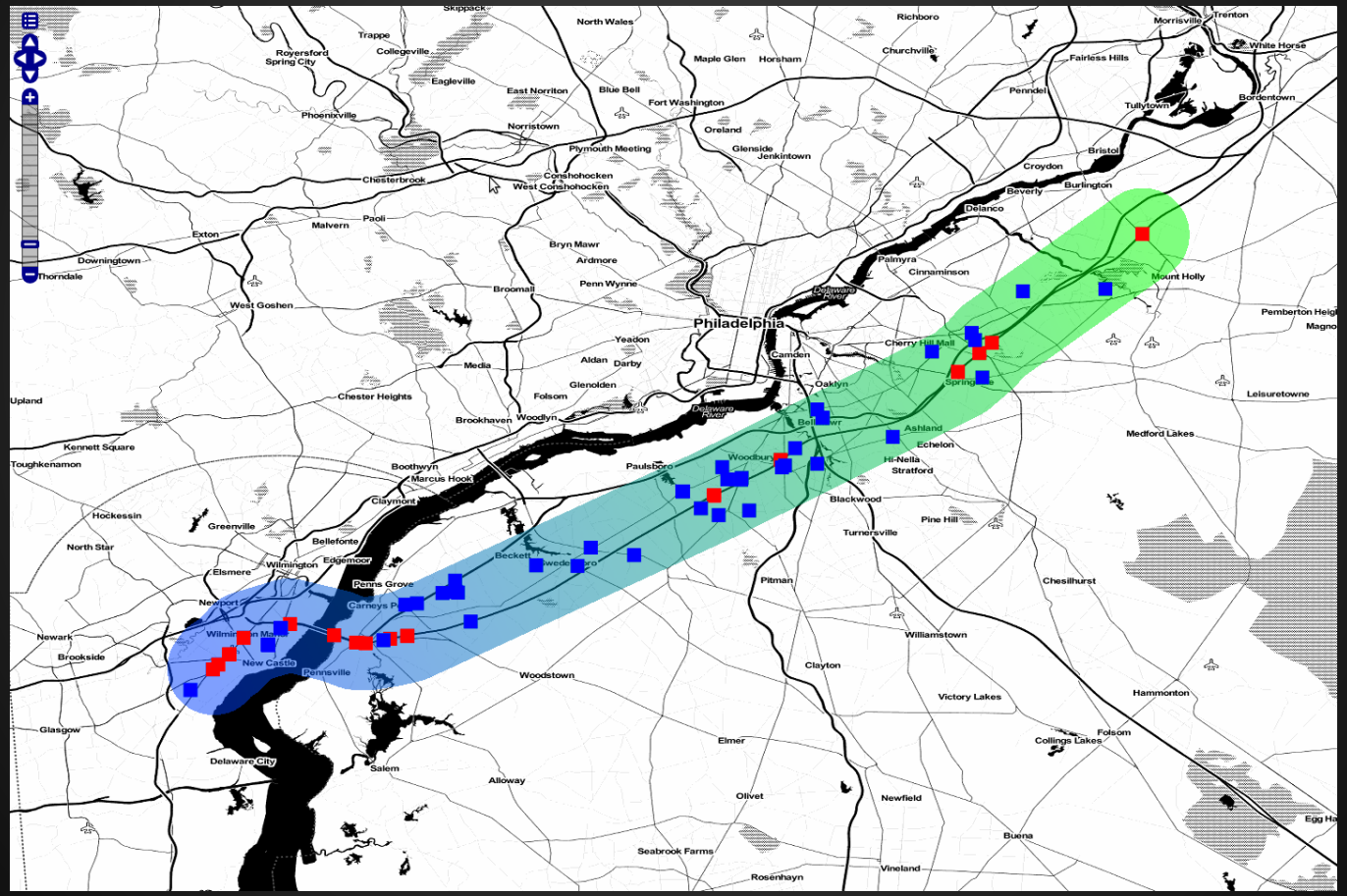
Query Planning



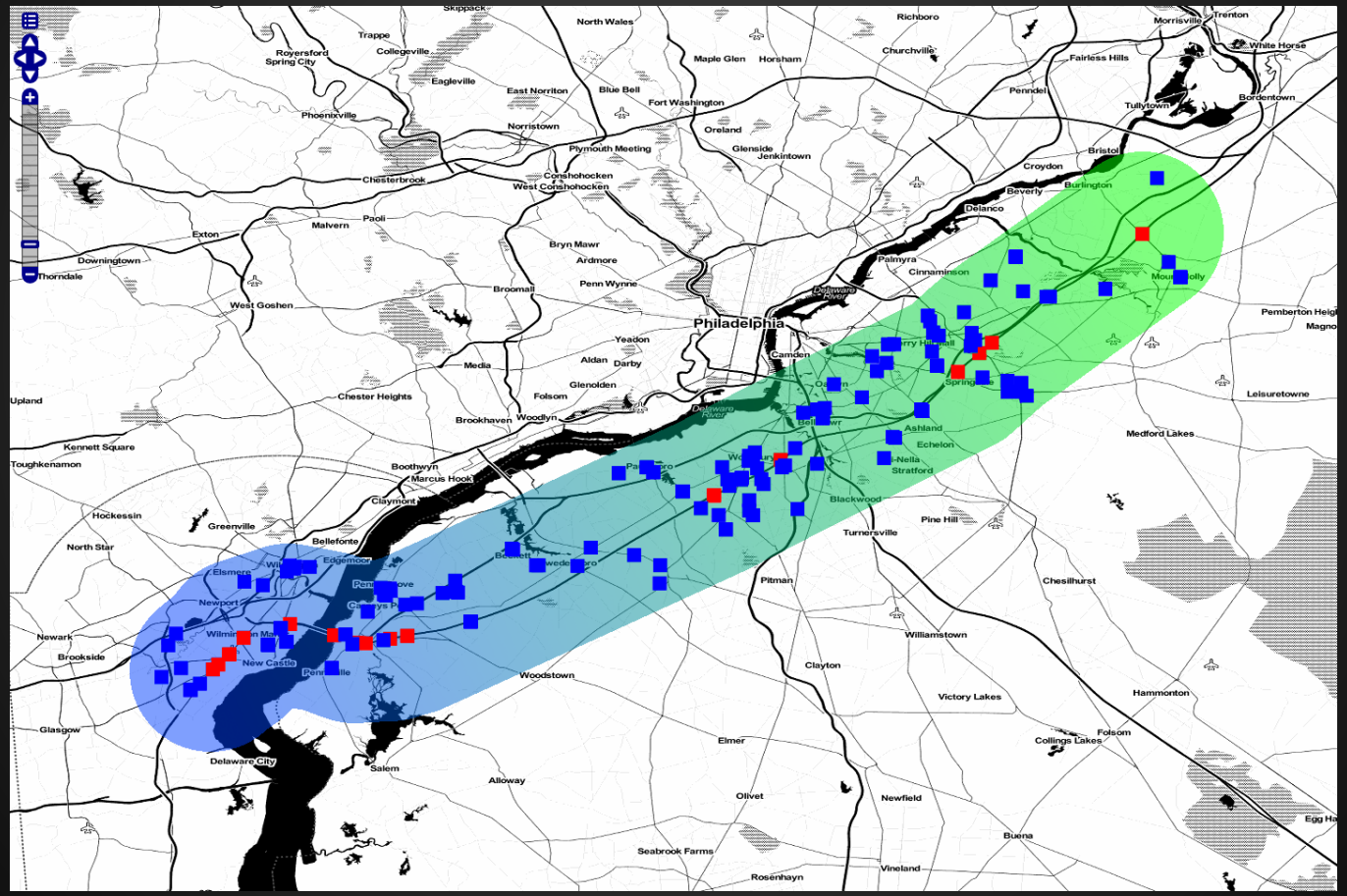
Query Planning



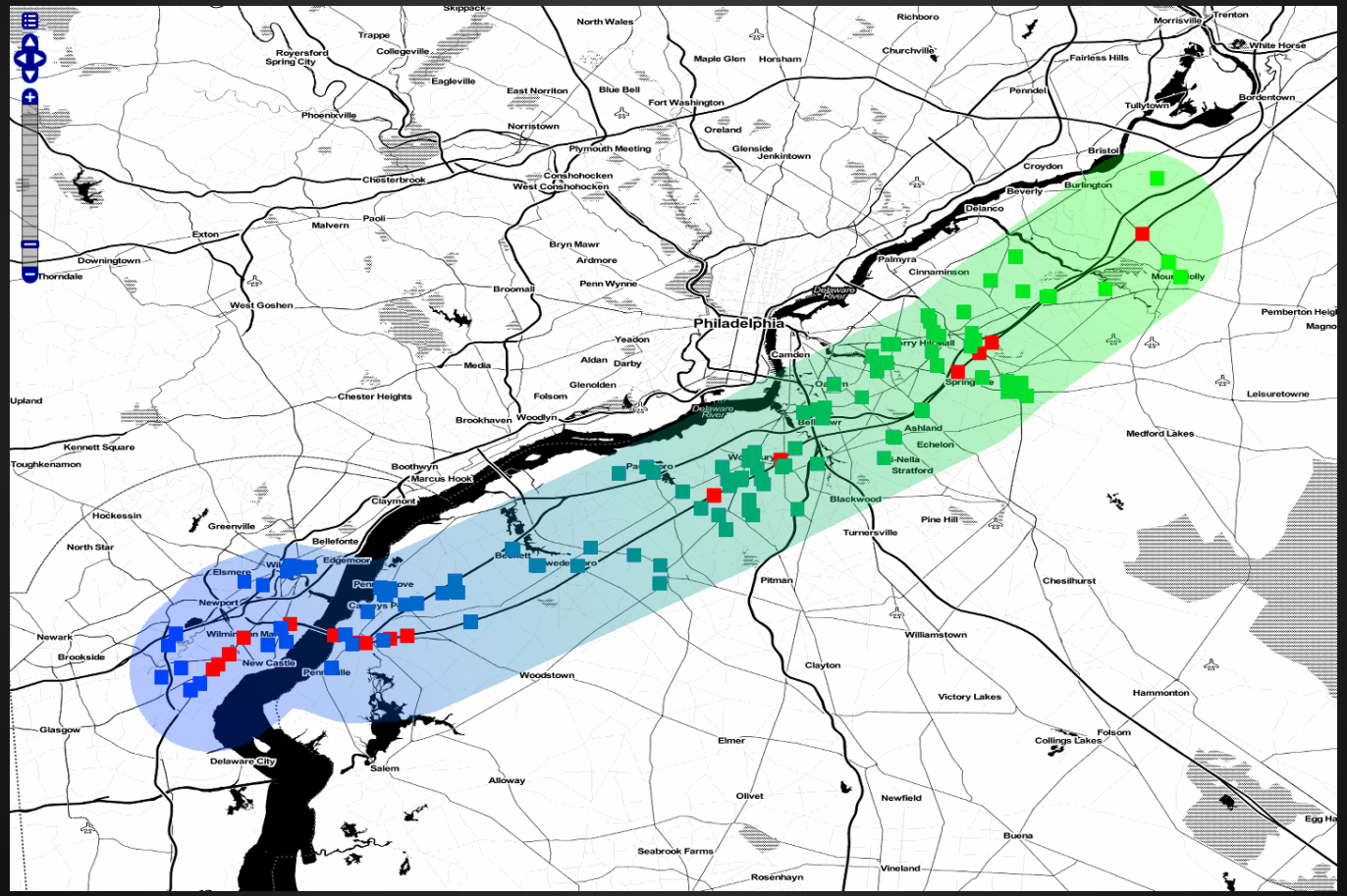
Interpolating Space-Time



Interpolating Space-Time



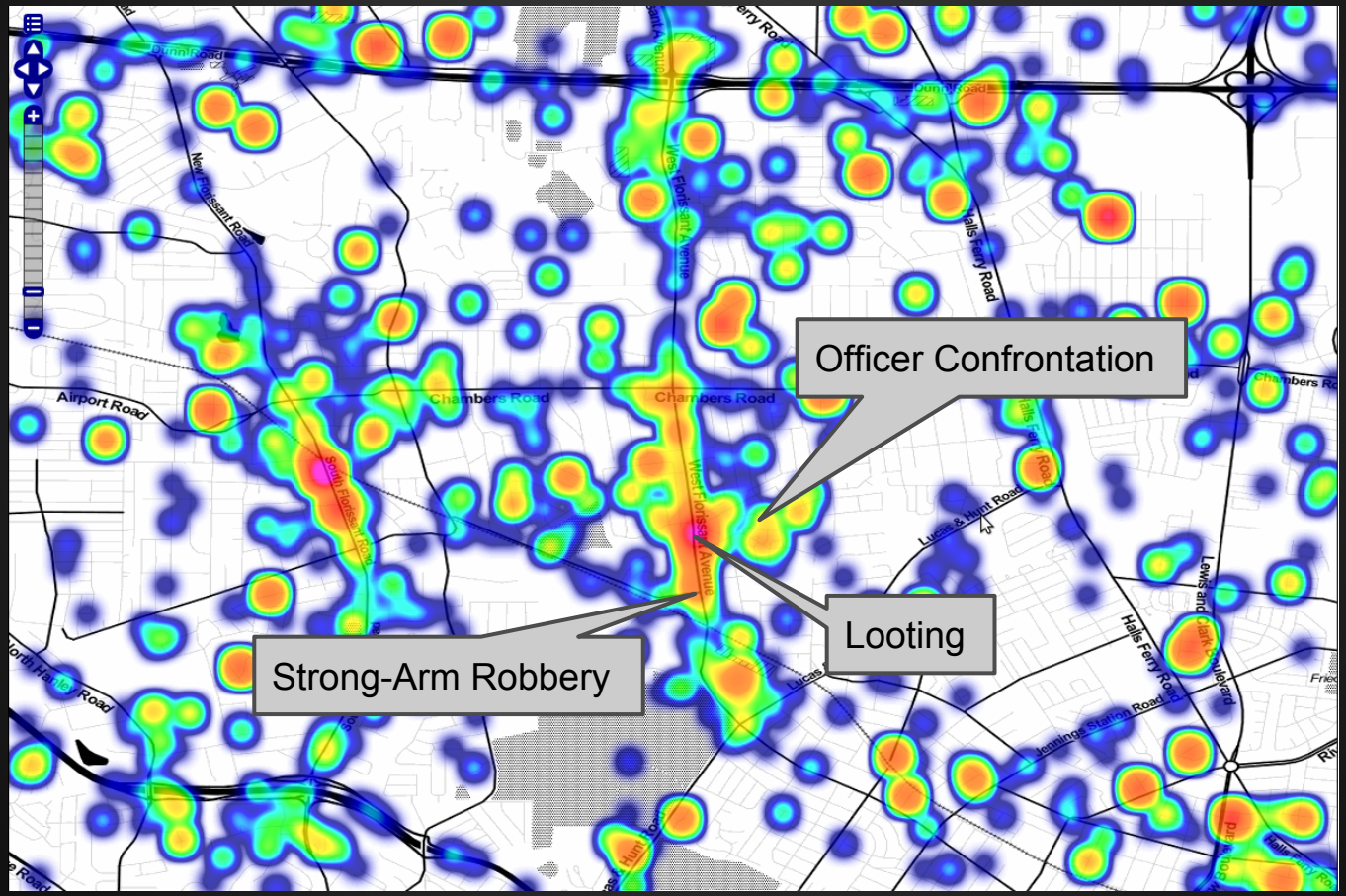
Interpolating Space-Time



Candidate Track Heuristics

- Cells Covered
- Route Cell Deviation
- tf*idf
- Motion Scoring
- Combined Scores

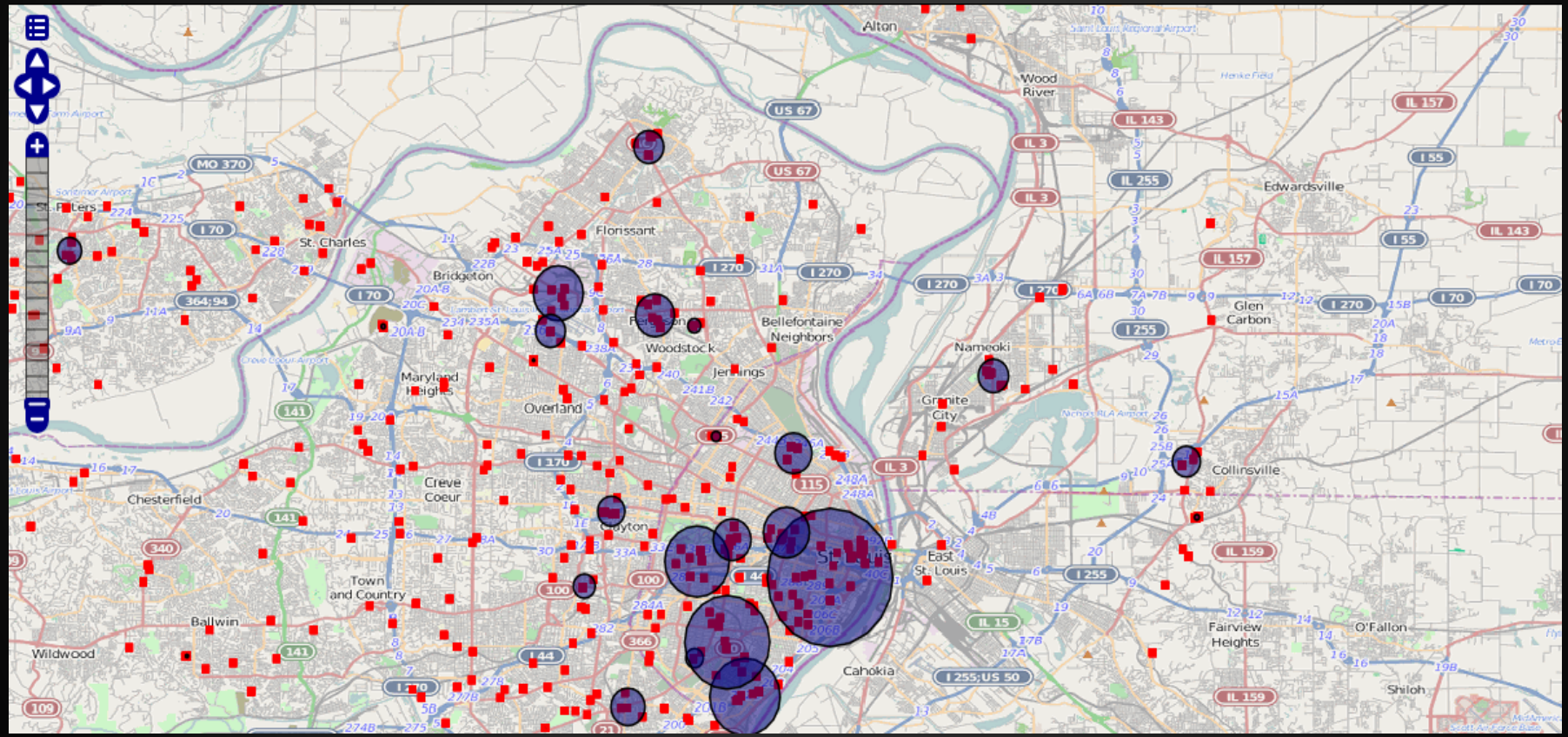
Current Events Analysis



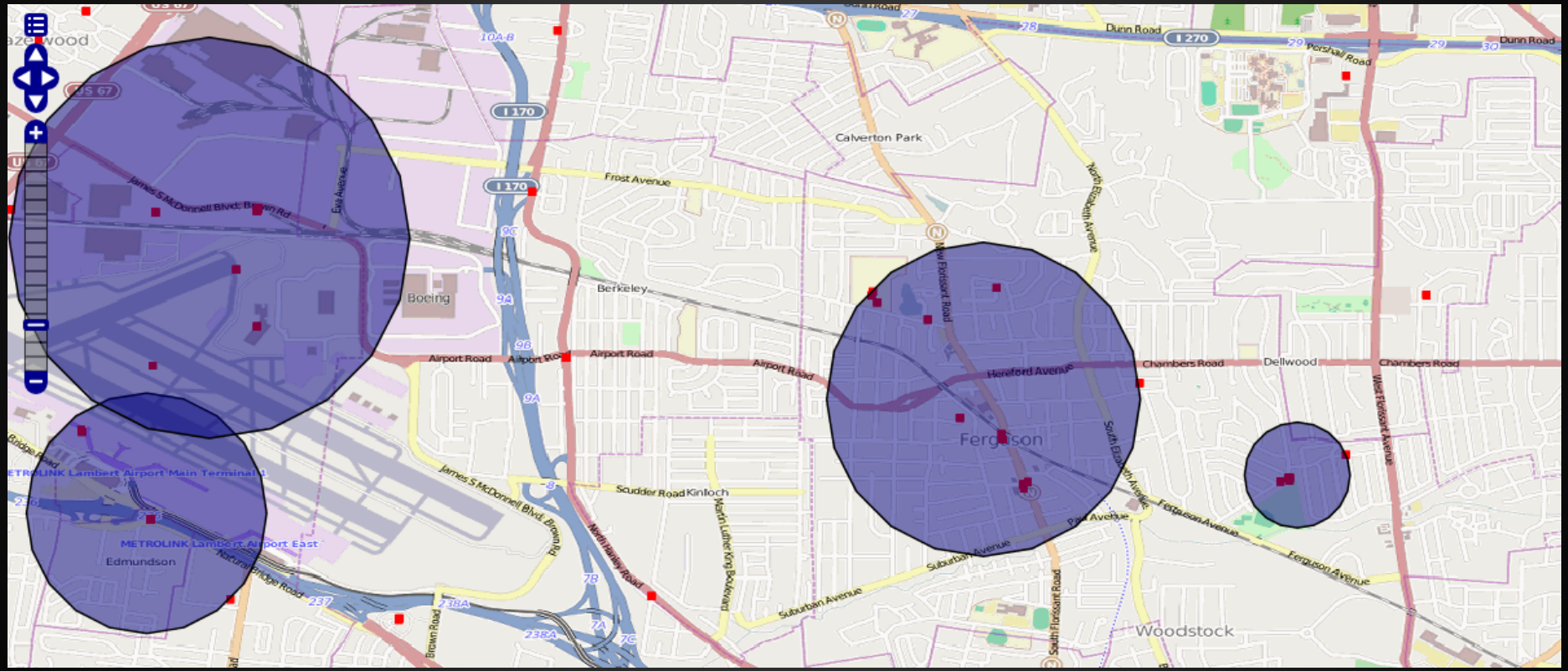
DBSCAN Clustering

- **D**ensity-**b**ased **S**patial **C**lustering of **A**pplications with **N**oise
- Clusters Algorithmically
  - No need to specify *k* (number of clusters)
- Arbitrarily shaped clusters

Tweet Clustering - St. Louis



Tweet Clustering - St. Louis



Apache Spark

- Real-time data analysis
- RDD (Resilient Distributed Dataset)
  - Transformations
  - Filters
  - Aggregations
- Example Time Series Output
  - Visualize via R

CQL BBOX Query

```
// Get a handle to the data store
val params = Map(
  "instanceId" -> "myinstance",
  "zookeepers" -> "zoo1,zoo2,zoo3",
  "user"       -> "username",
  "password"   -> "password",
  "tableName"  -> "geomesa_catalog")

val ds = DataStoreFinder.getDataStore(params).asInstanceOf[AccumuloDataStore]

// Construct a CQL query to filter by bounding box
val ff = CommonFactoryFinder.getFilterFactory2
val f = ff.bbox("geom",  -90.32023,38.72009,-90.23957,38.77019, "EPSG:4326")
val q = new Query(feature, f)
```

GeoMesa RDD

```
val conf = new Configuration
val sconf = init(new SparkConf(true), ds)
val sc = new SparkContext(sconf)

val queryRDD = geomesa.compute.spark.GeoMesaSpark.rdd(conf, sconf, ds, query)
```

Projection & Aggregation

```
// Convert RDD[SimpleFeature] to RDD[(String, SimpleFeature)] where the first
// element of the tuple is the date to the day resolution
val dayAndFeature = queryRDD.mapPartitions { iter =>
  val df = new SimpleDateFormat("yyyyMMdd")
  val ff = CommonFactoryFinder.getFilterFactory2
  val exp = ff.property("dtg")
  iter.map { f => (df.format(exp.evaluate(f).asInstanceOf[java.util.Date]), f) }
}

// Aggregate and output
val groupedByDay = dayAndFeature.groupBy { case (date, _) => date }
val countByDay = groupedByDay.map { case (date, iter) => (date, iter.size) }
countByDay.collect.foreach(println)
```

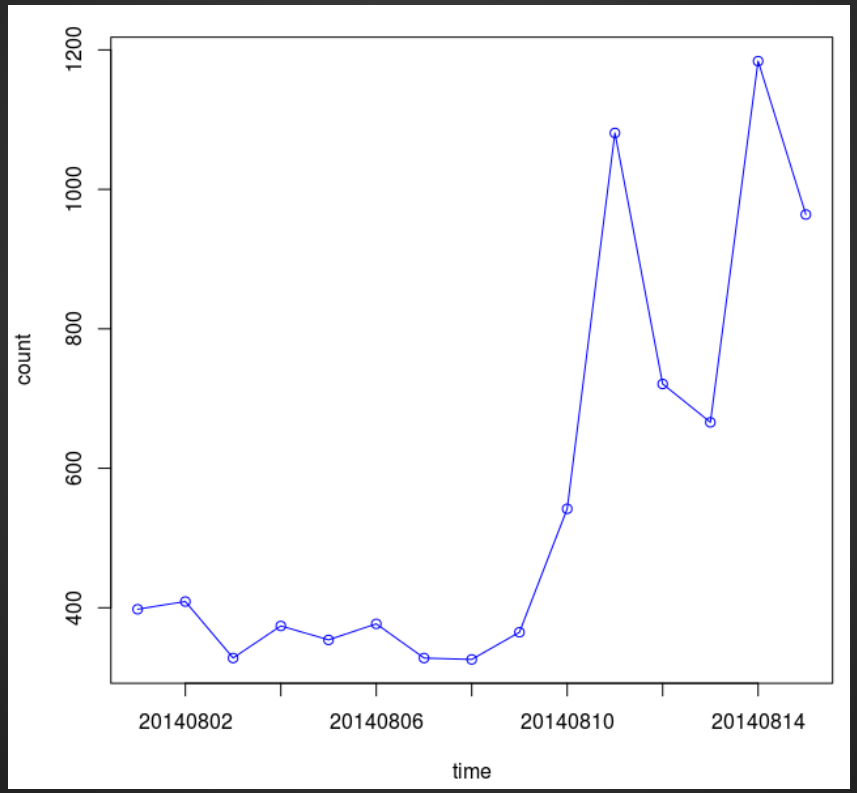
Time Series in R

```
14/08/18 01:05:57 INFO SparkContext: Job finished: collect at
Runner.scala:61, took 44.154914093 s
(20140801,398)
(20140802,409)
(20140803,328)
(20140804,374)
(20140805,354)
(20140806,377)
(20140807,328)
(20140808,326)
(20140809,365)
(20140810,542)
(20140811,1081)
(20140812,721)
(20140813,666)
(20140814,1184)
(20140815,964)
```

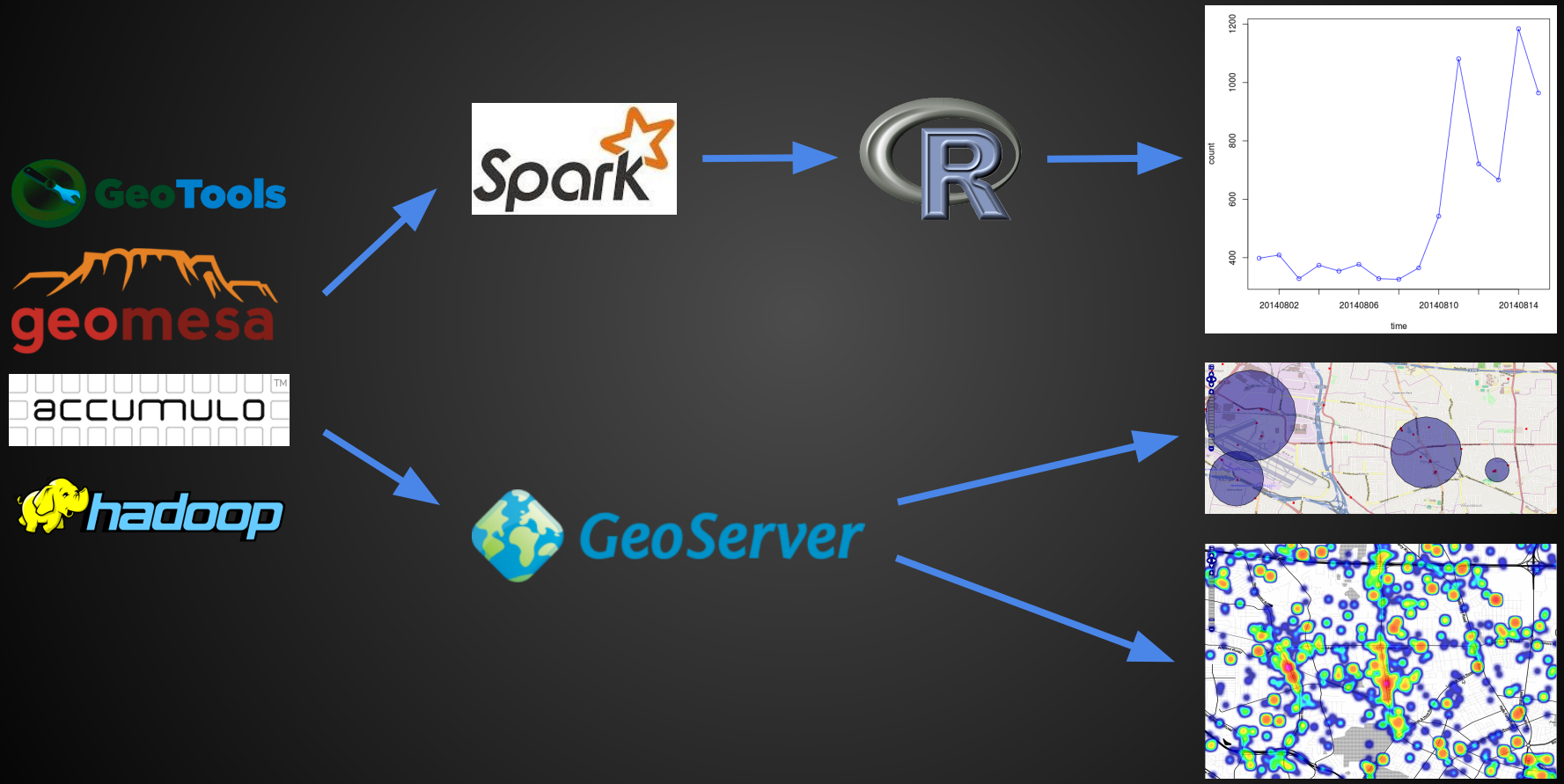
```r
# read tweets
tweets = read.csv("~/Desktop/twitter.csv")

# plot tweets
plot(tweets,  col="blue", type="o")
```

R Visualization



OpenSource Geospatial Analytic Pipeline



Roadmap

- Query Planning
  - Stats & Automatic Index Optimization
  - Multi-level Geospatial Indexes
- Enhance Security
- Ease Developer on-ramping
- Analytics
  - Predictive spatio-temporal analytics as a Service
  - Parallel DBSCAN
- LocationTech Community Growth

La fin du monde

Tutorials, Papers, Code, & more

http://geomesa.org

http://github.com/locationtech/geomesa

Email

geomesa-users@locationtech.org

geomesa-dev@locationtech.org

CCRi

DATA TO KNOWLEDGE