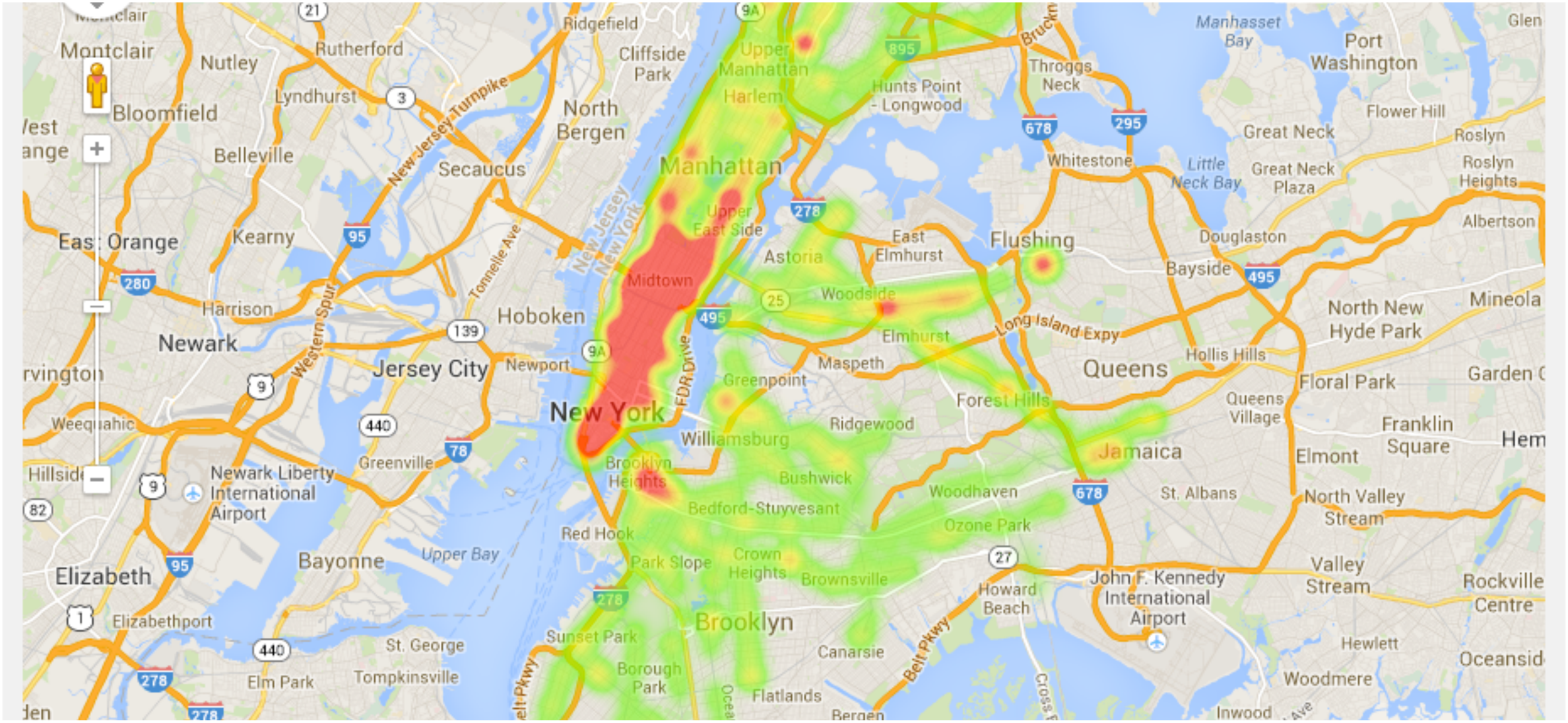
Advanced Data Management (CSCI 680/490)

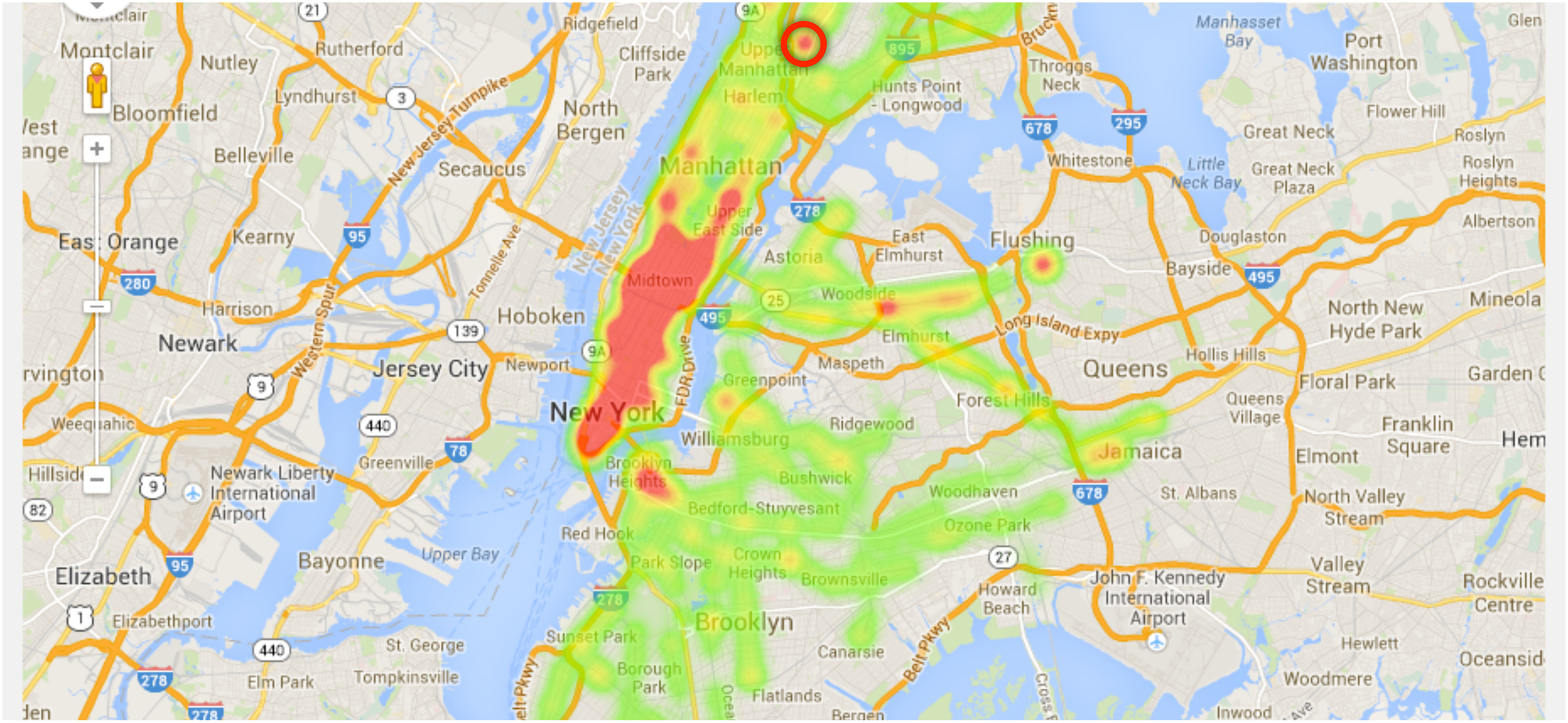
Spatial Data

Dr. David Koop

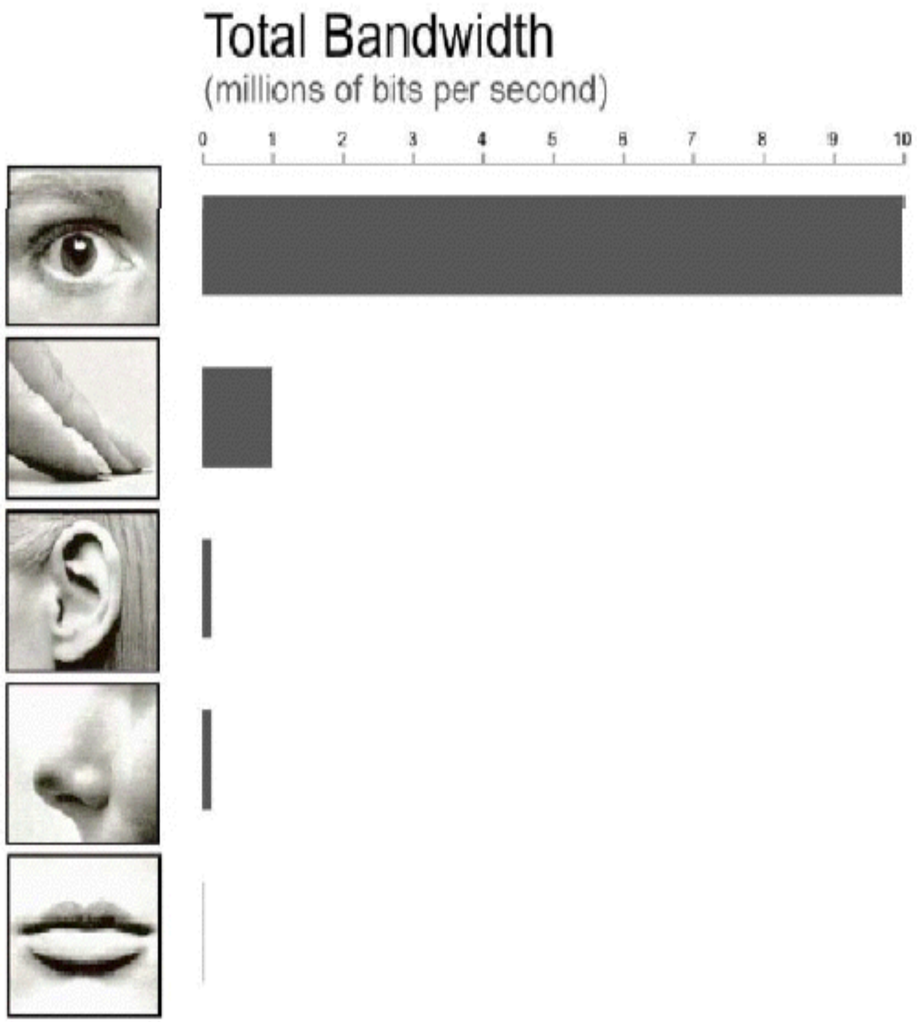
Data Exploration Through Visualization



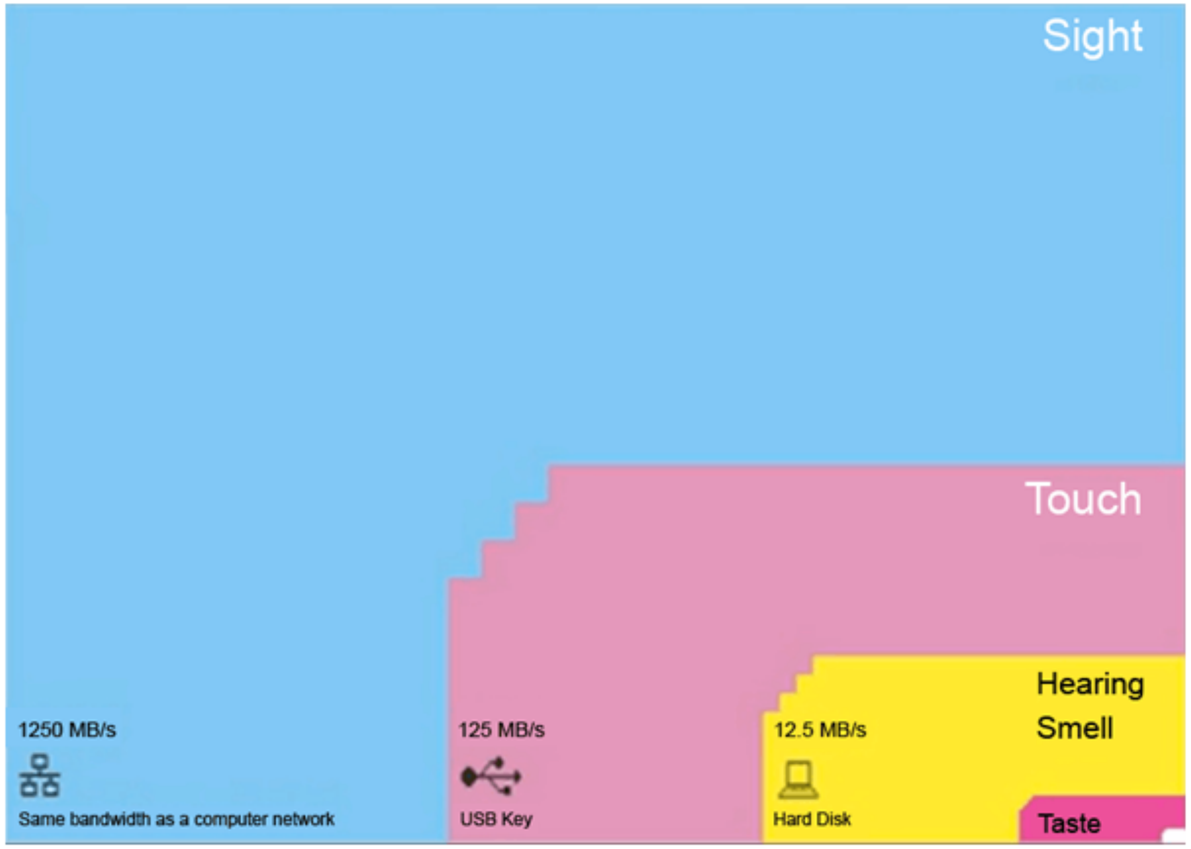
Data Exploration Through Visualization



Why do we visualize data?



[via A. Lex]



[T. Nørretranders]

Why Visual?

I		II		III		IV	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

[F. J. Anscombe]

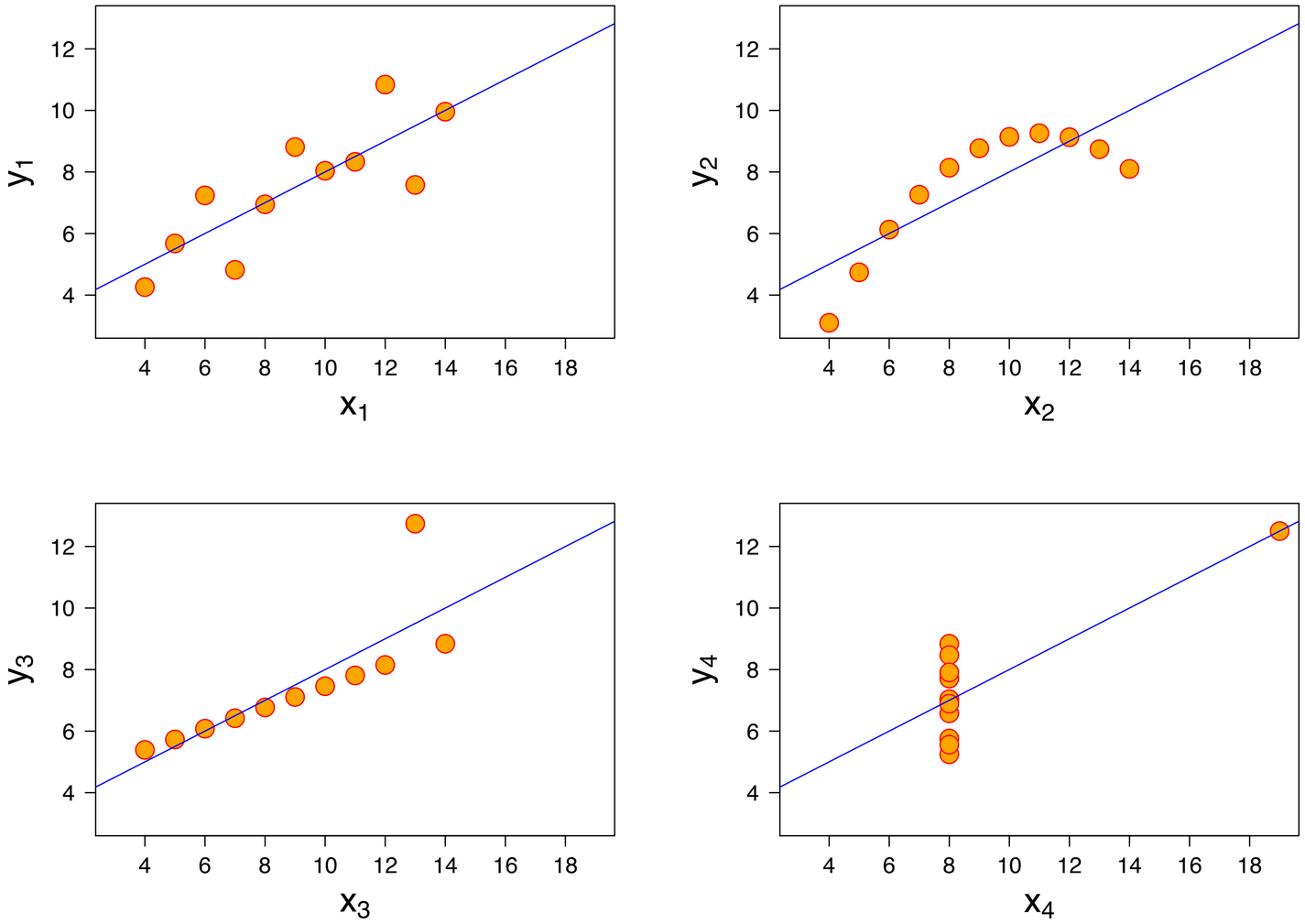
Why Visual?

I		II		III		IV	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

Mean of x	9
Variance of x	11
Mean of y	7.50
Variance of y	4.122
Correlation	0.816

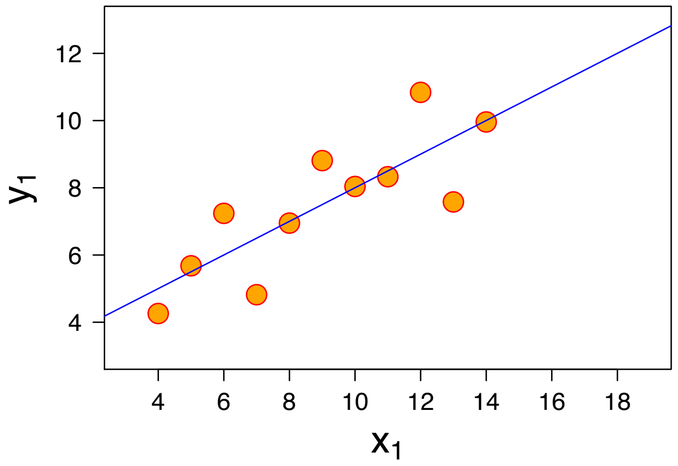
[F. J. Anscombe]

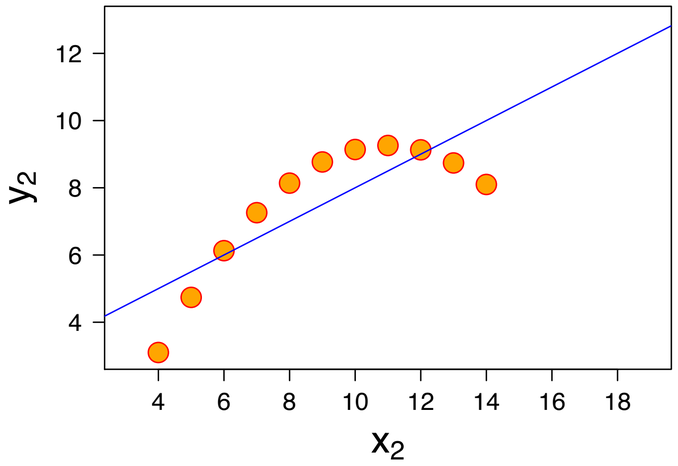
Why Visual?

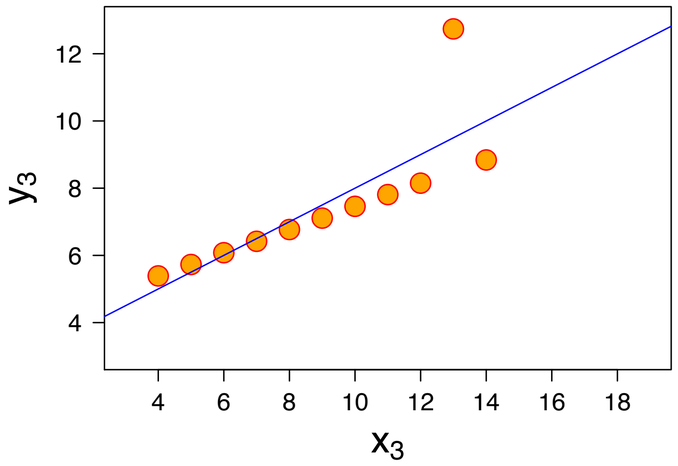


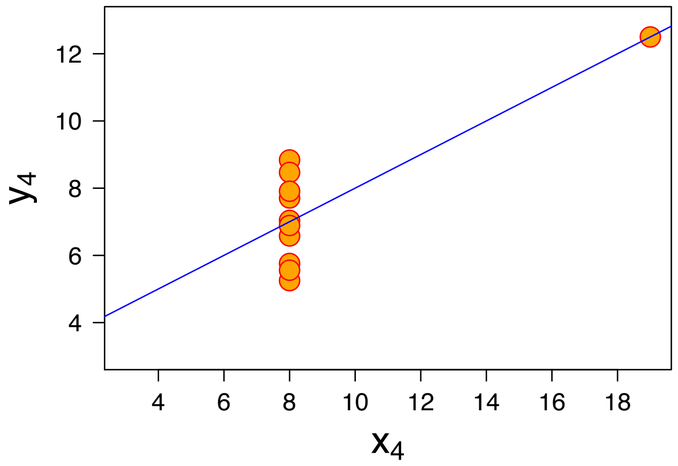
[F. J. Anscombe]

Why Visual?





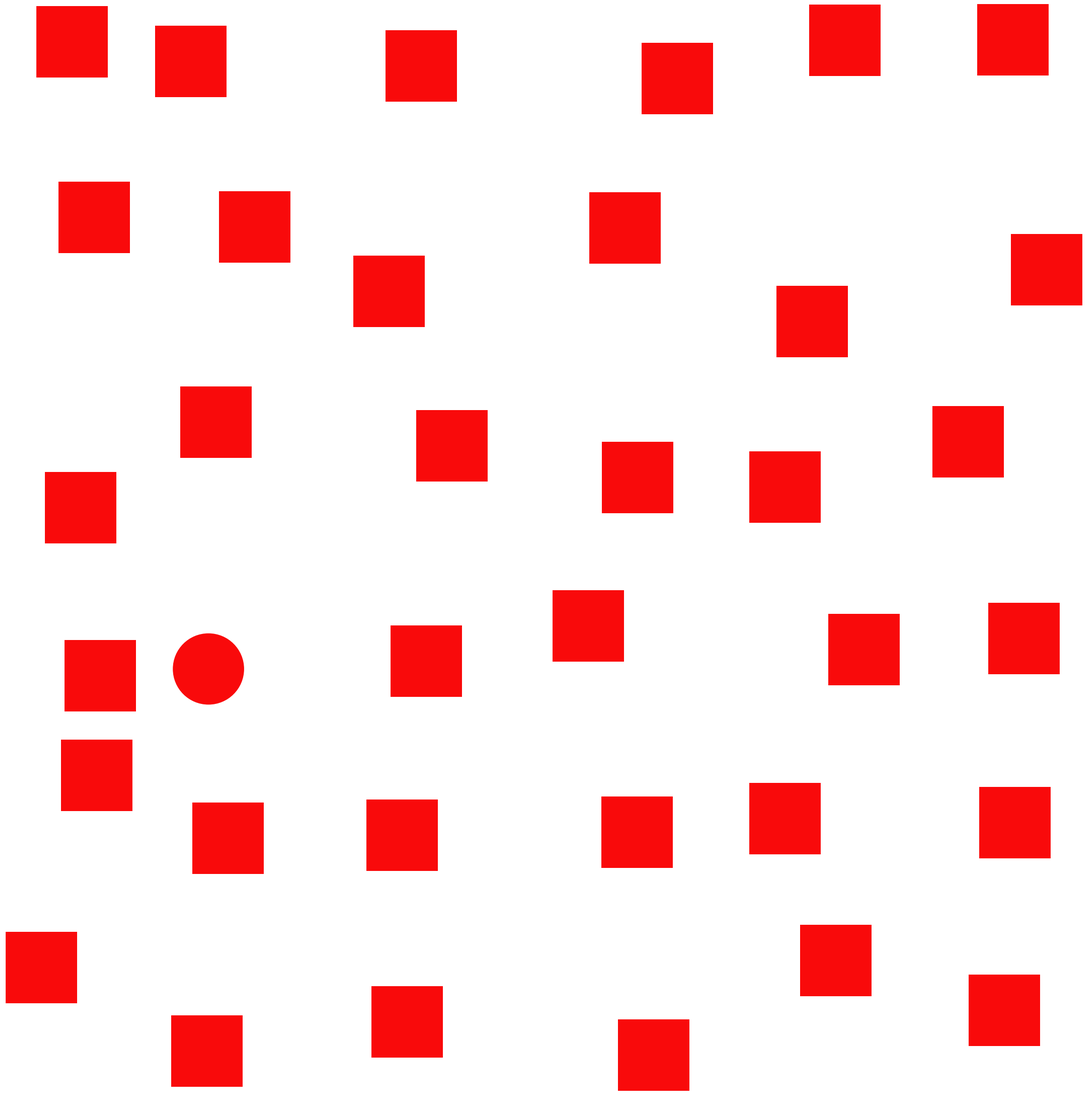




Mean of x	9
Variance of x	11
Mean of y	7.50
Variance of y	4.122
Correlation	0.816

[F. J. Anscombe]

Visual Pop-out



[C. G. Healey]

Supporting Scalable Visualization

- Two Problems:
  - **Lots of data**, how to display (encode) it
  - User **interaction** is key to gaining insight, requires **low latency**
- Addressing big data:
  - Encoding should focus on **available resolution**, not size of data
  - Approaches:
    - Sampling
    - Modeling
    - **Binning**
  - Bin → Aggregate (→ Smooth) → Plot

[J. Heer]

Effects of Latency

- **Higher latency** leads to...
  - Reduced user activity and data set coverage
  - Significantly fewer brushing actions
  - Less observation, generalization & hypothesis
- **Interaction effect**: Exposure to delay reduces subsequent performance in low-latency interface.
- Different interactions exhibit **varied sensitivity** to latency. Brushing is highly sensitive!

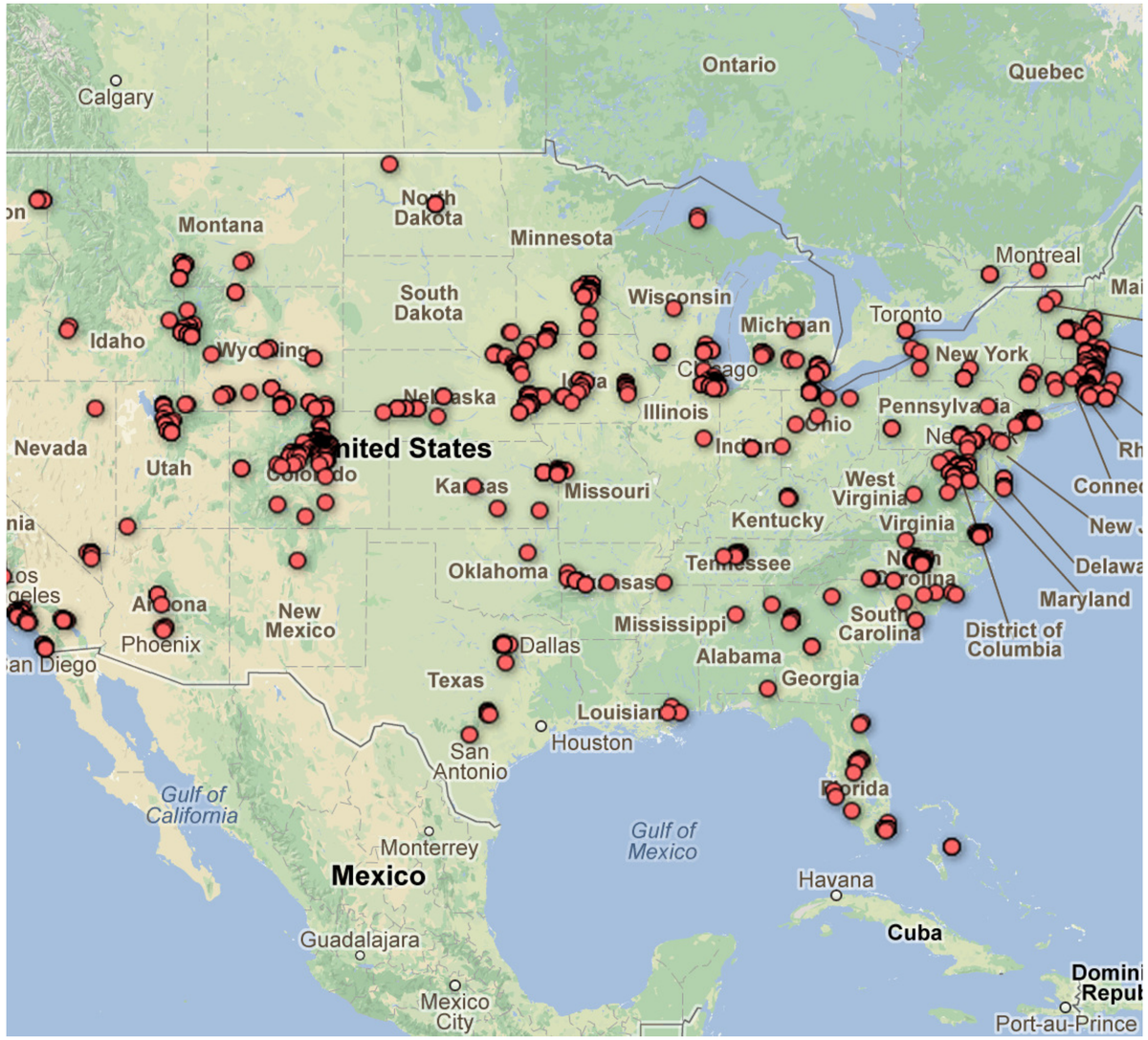
[Liu et al. via J. Heer]

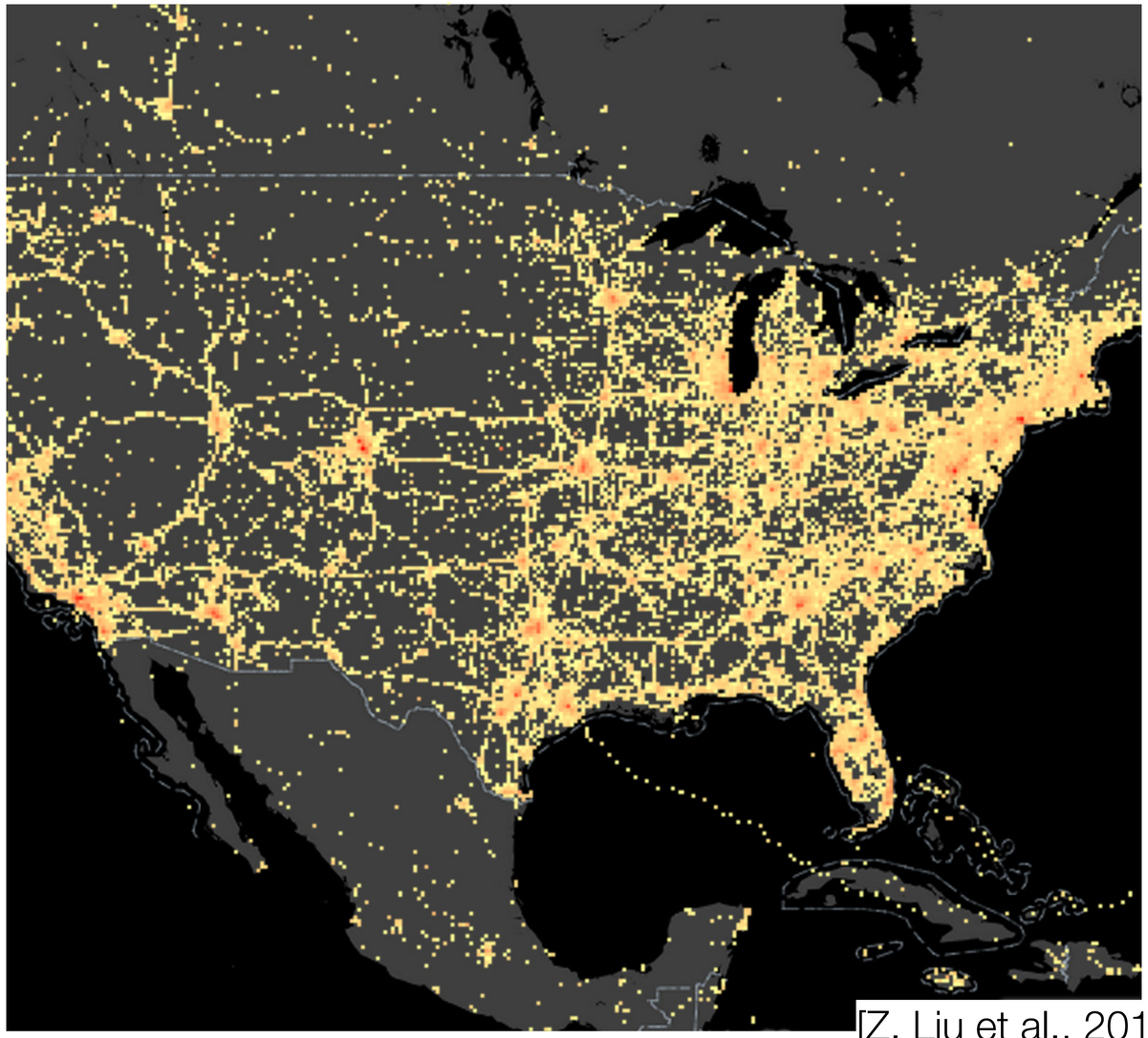
Interactive Scalability Solutions

- Use Database Technology: databases built for scalability
- Client-side Indexing (Data Cubes): take advantage of data structure
- Prefetching: load data before requests based on predictions
- Approximation: show estimates early but with error information

[J. Heer]

Sampling vs. Aggregation

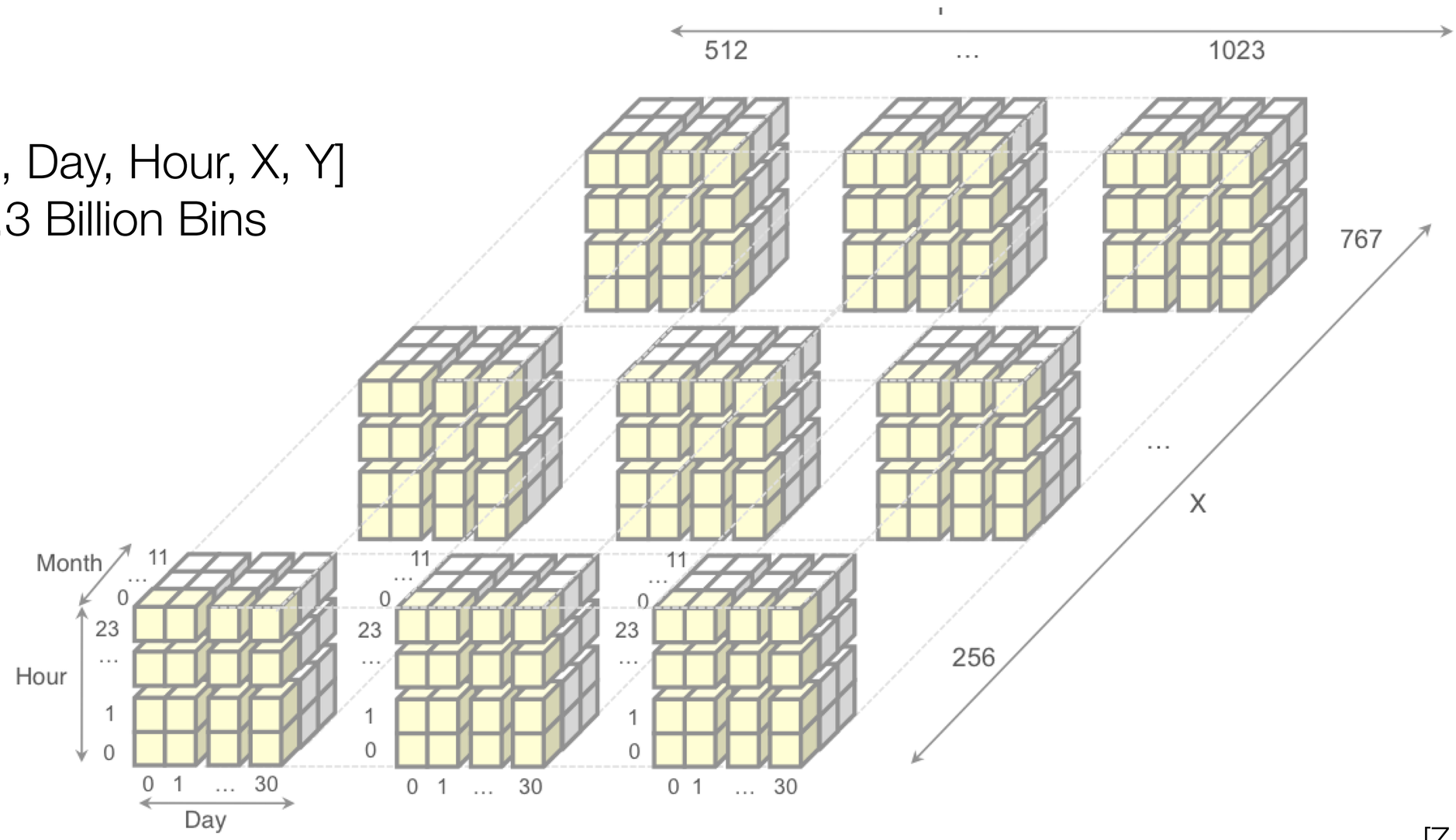




[Z. Liu et al., 2013]

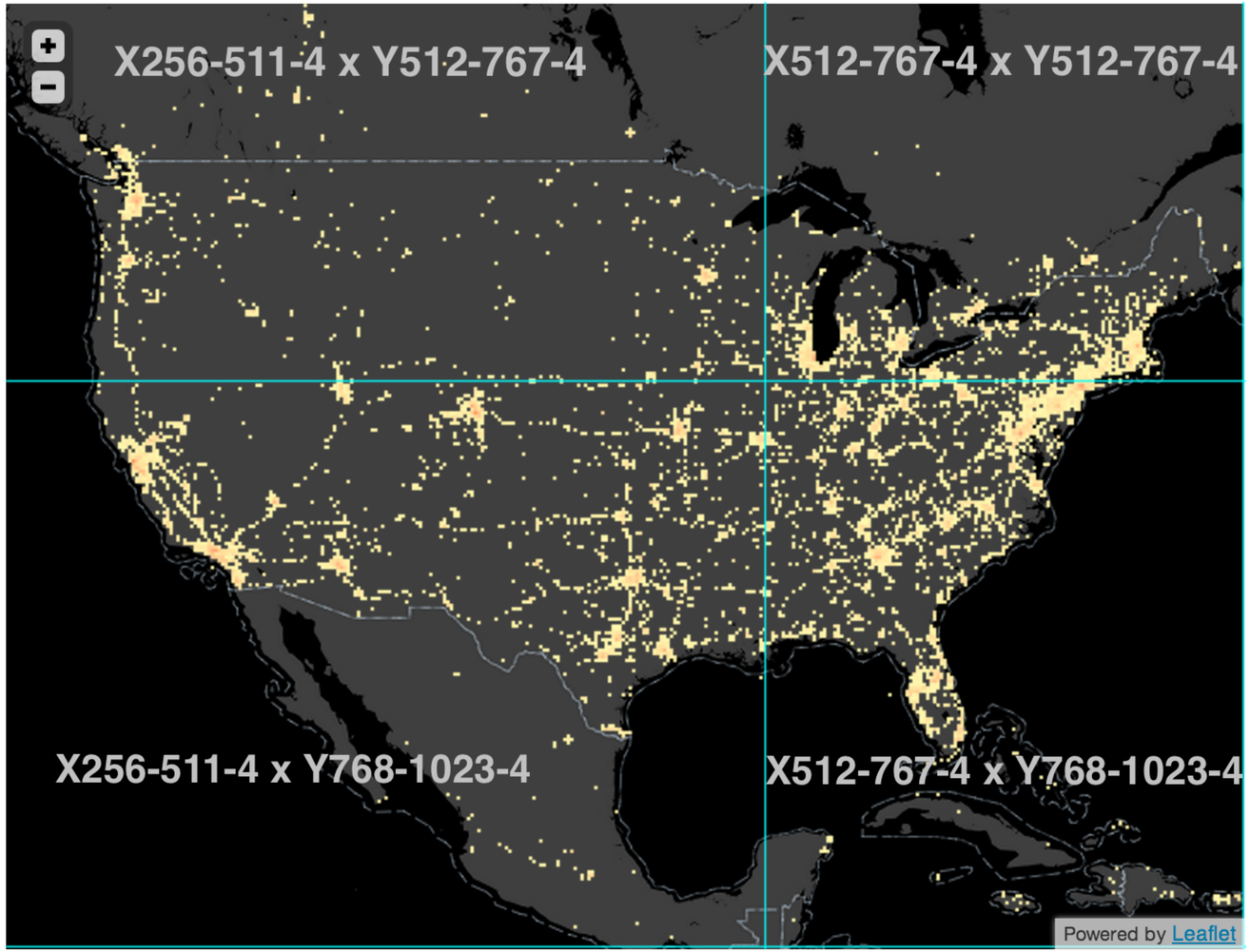
Full 5-D Data Cube

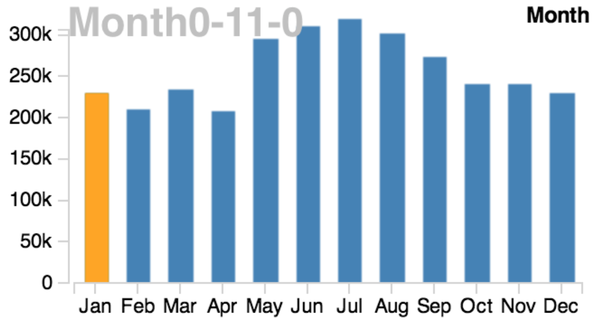
[Month, Day, Hour, X, Y]
~2.3 Billion Bins

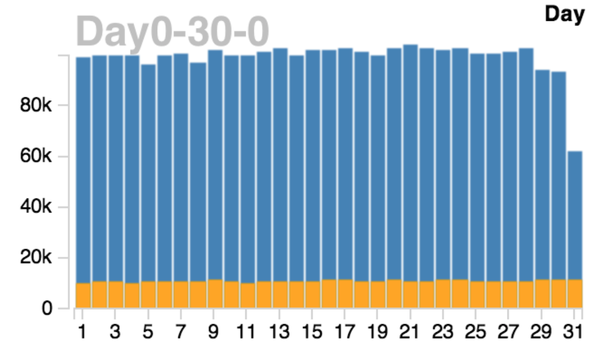


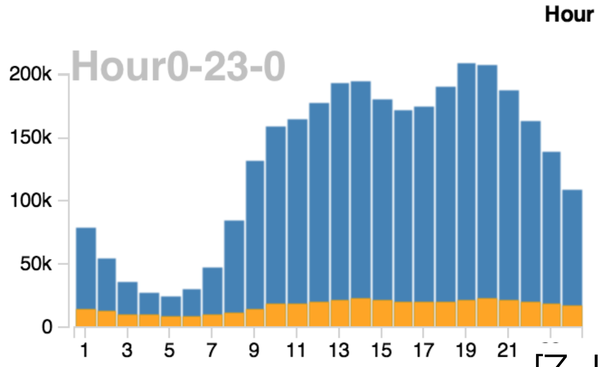
[Z. Liu et al., 2013]

Break into Tiles



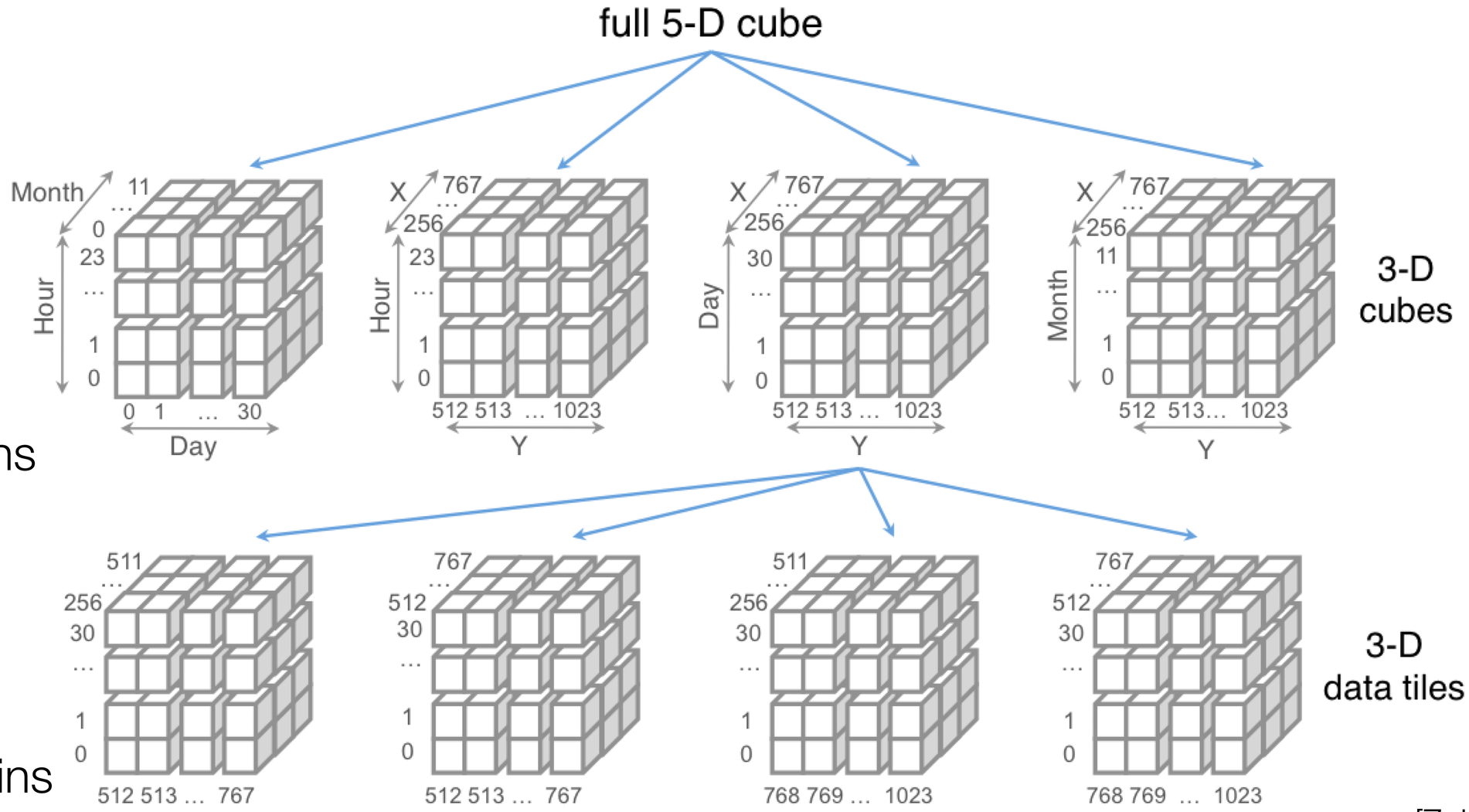






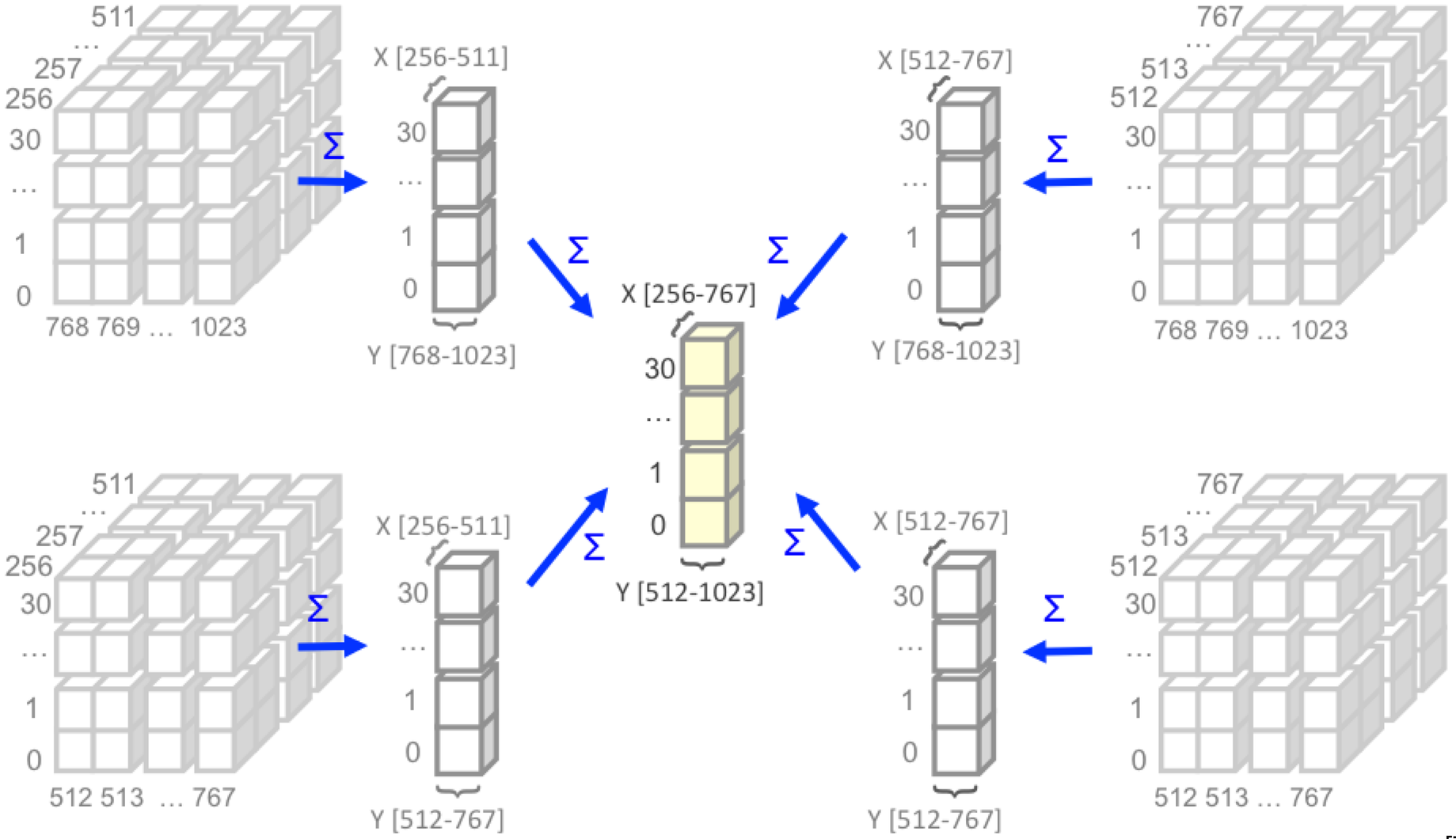
[Z. Liu et al., 2013]

Data Cube Decomposition



[Z. Liu et al., 2013]

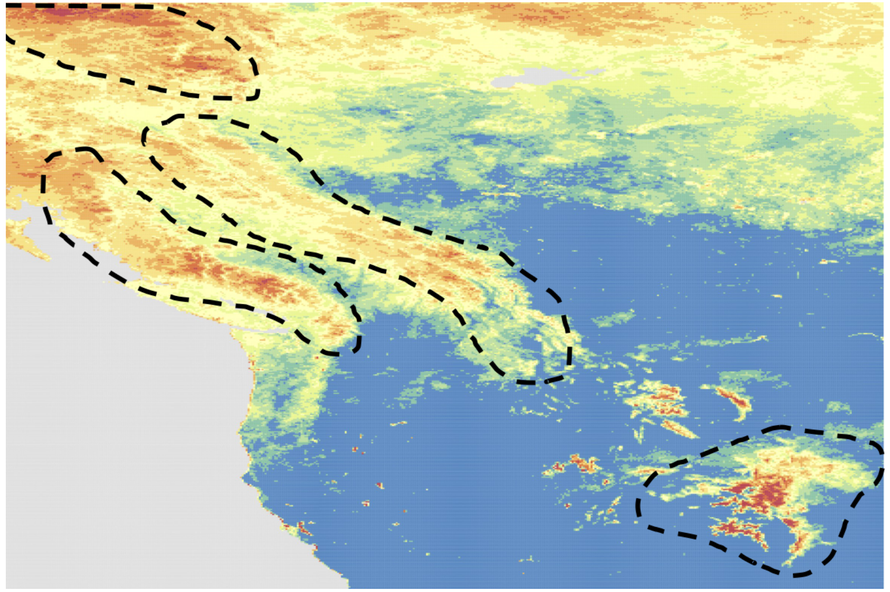
Aggregation of Tiles for Results

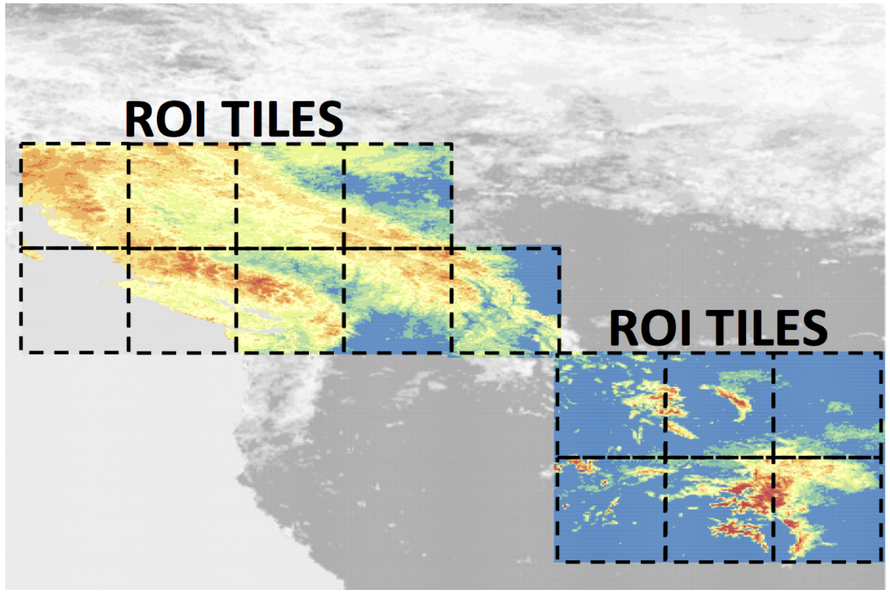


[Z. Liu et al., 2013]

Prefetching (ForeCache)

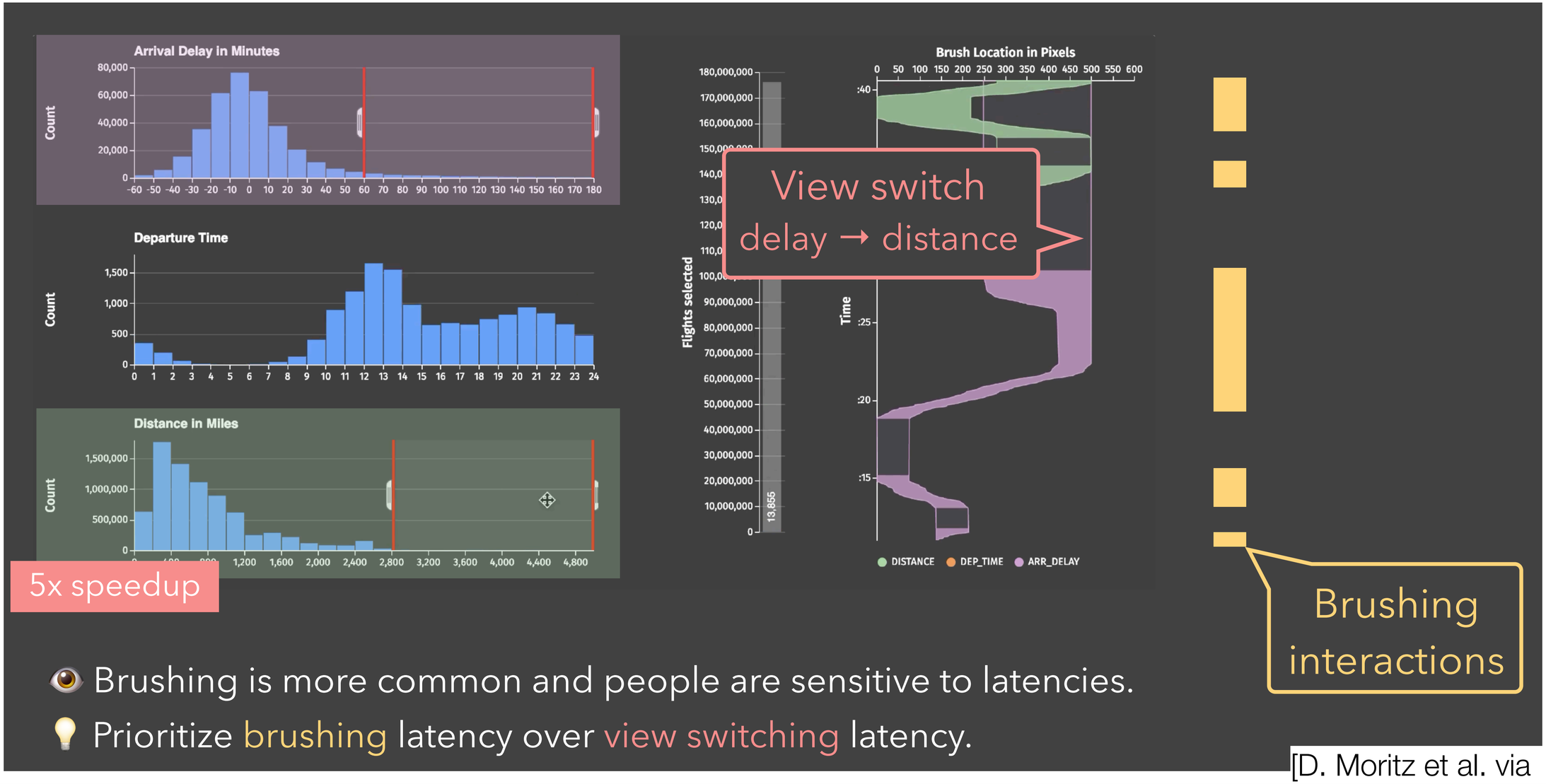
- Predict which tiles a user will need next and prefetch those
  - Use common patterns (zoom, pan)
  - Use regions of interest (ROIs)





[Battle et al., 2016]

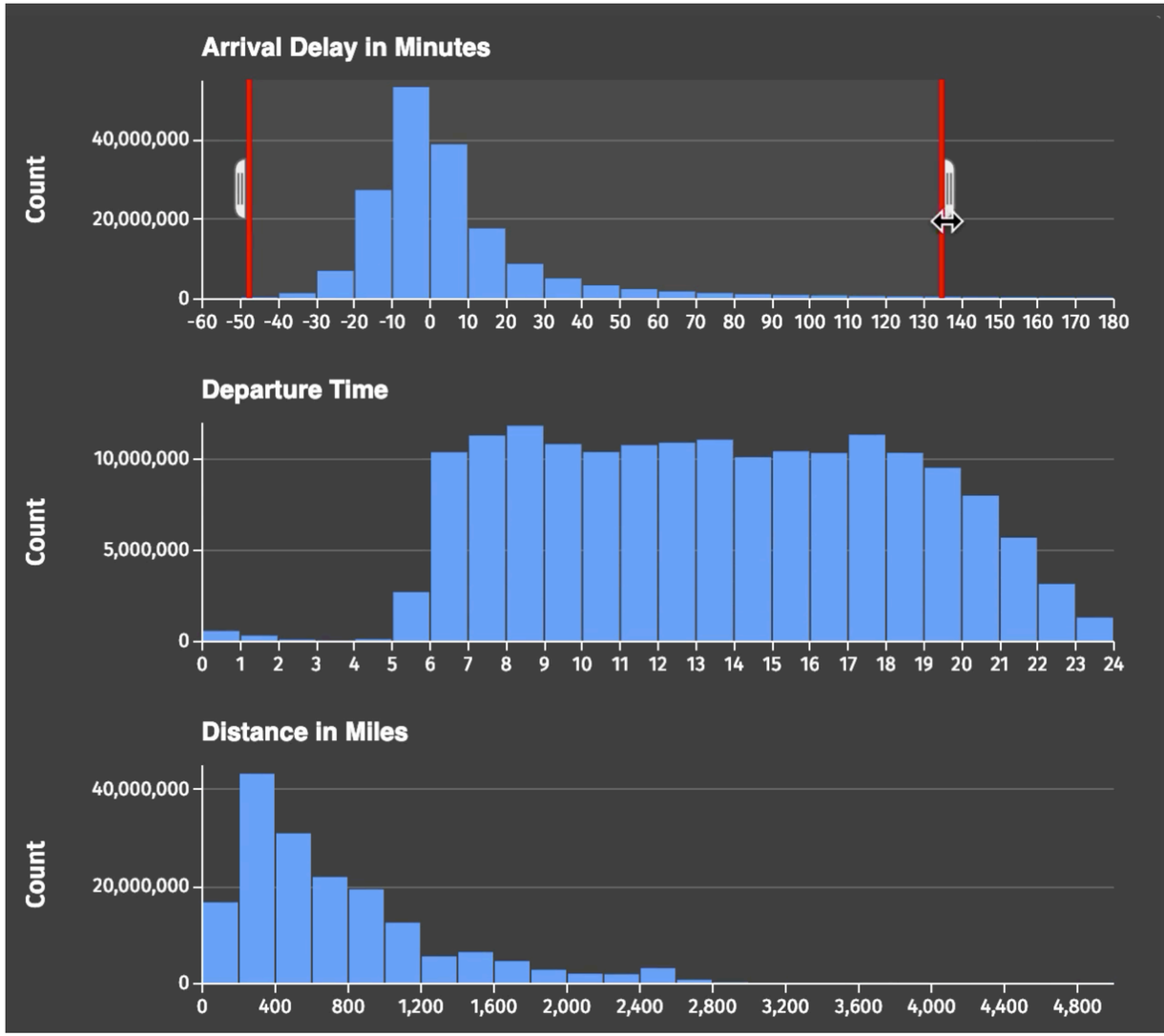
Latency Differences in Tasks



- 👁️ Brushing is more common and people are sensitive to latencies.
- 💡 Prioritize brushing latency over view switching latency.

[D. Moritz et al. via J. Heer]

Task-Prioritized Prefetching



brushes in the precomputed view

serves requests from a data cube

Data Cube. Gray et al. *1997.*

interacts with a new view

query for new data cubes

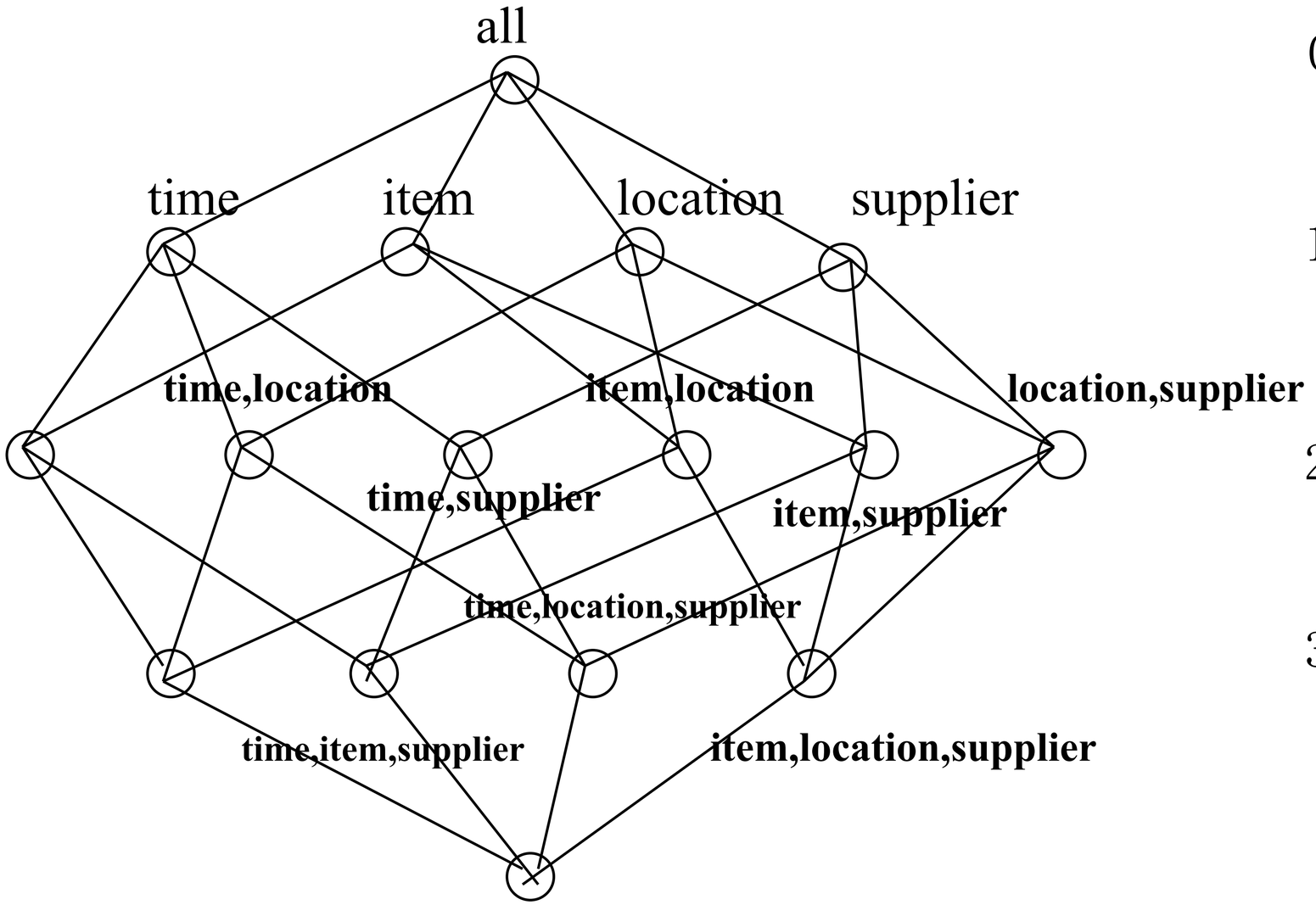
[D. Moritz et al. via J. Heer]

Reading Quiz

Data Cubes

J. Han, M. Kamber, and J. Pei

Data Cube: A Lattice of Cuboids



0−D (*apex*) cuboid

1−D cuboids

2−D cuboids

3−D cuboids

4−D (*base*) cuboid

[Han et al., 2011]

Cube Operations

- Roll-up: aggregate up the given hierarchy
- Drill-down: refine down the given hierarchy
- Roll-up and drill-down are "inverses"

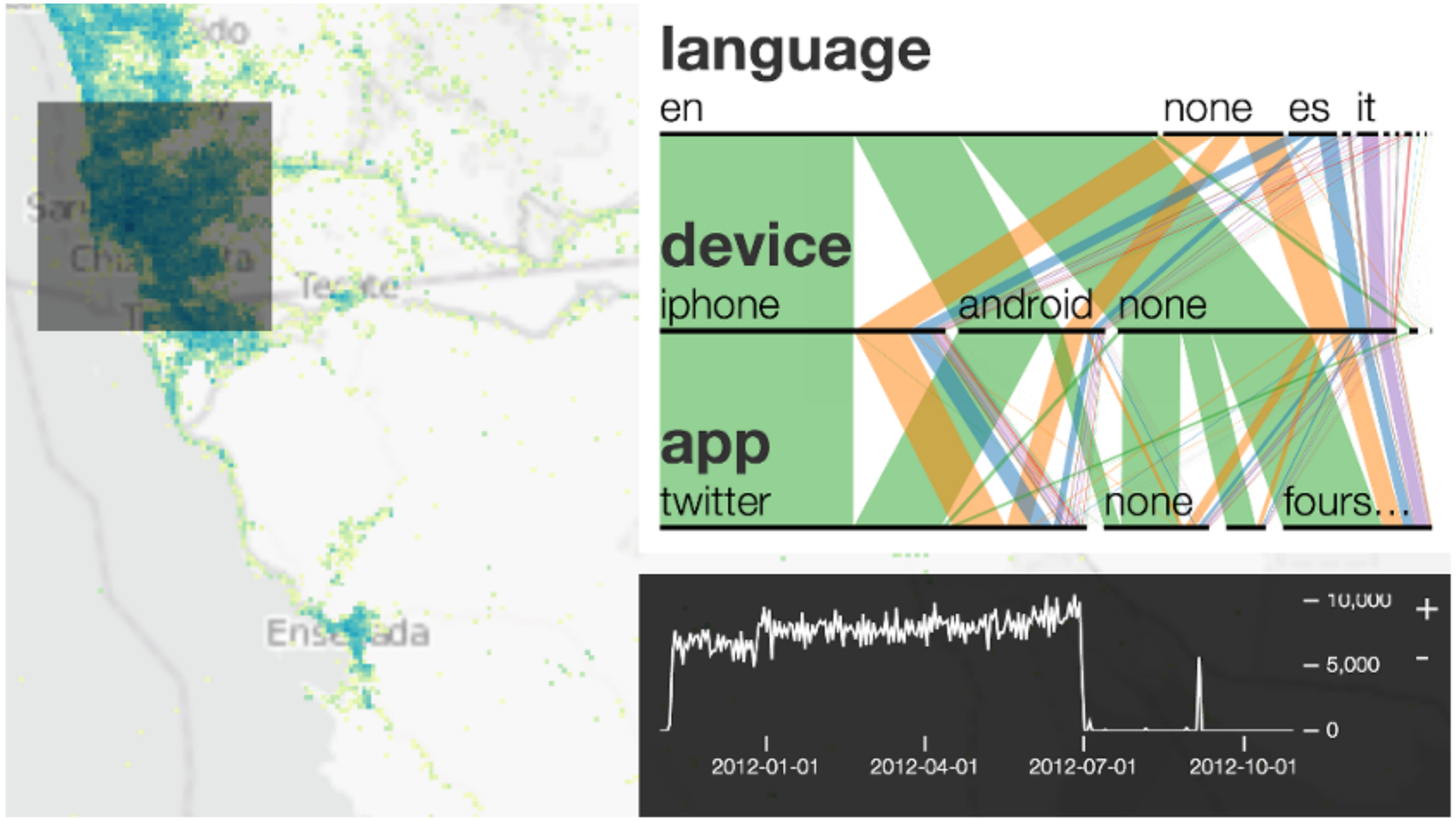
Spatial Data Exploration Motivation

L. Battle

Nanocubes for Real-Time Exploration of Spatiotemporal Datasets

L. Lins, J. T. Klosowski, and C. Scheidegger

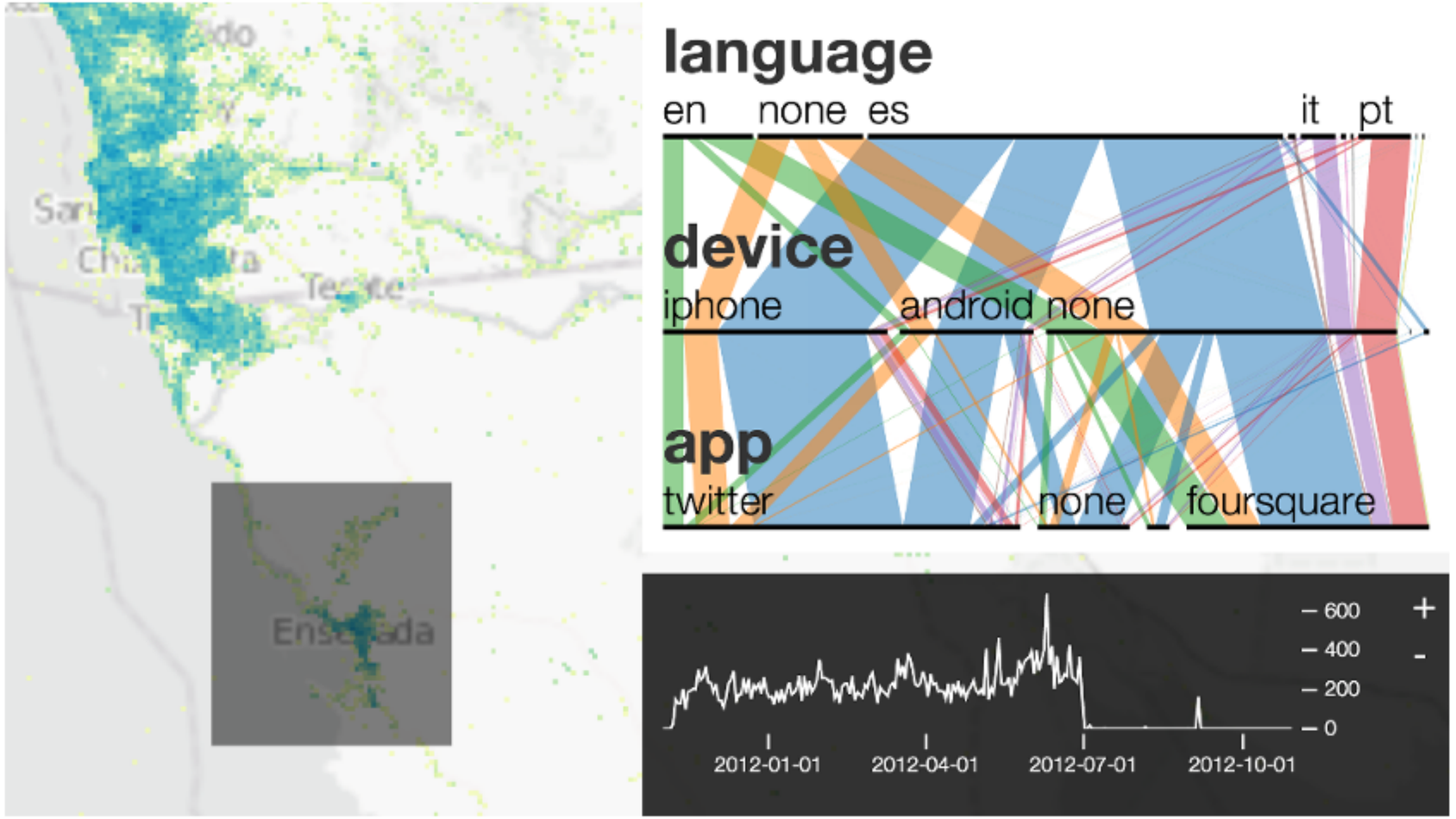
Goal: Interactive Exploration of Data Cubes



Linked view of tweets in San Diego, US

[Lins et. al, 2013]

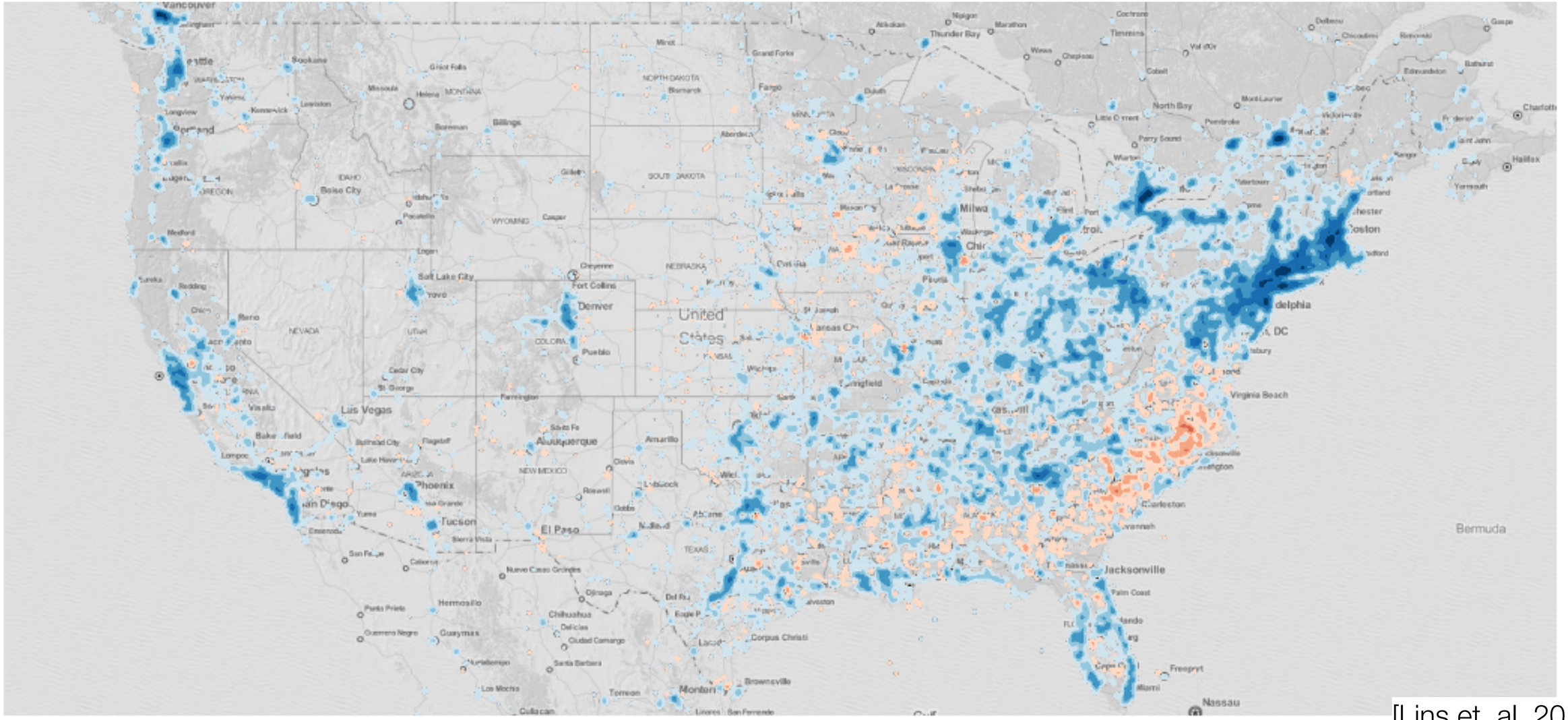
Move to Another Location



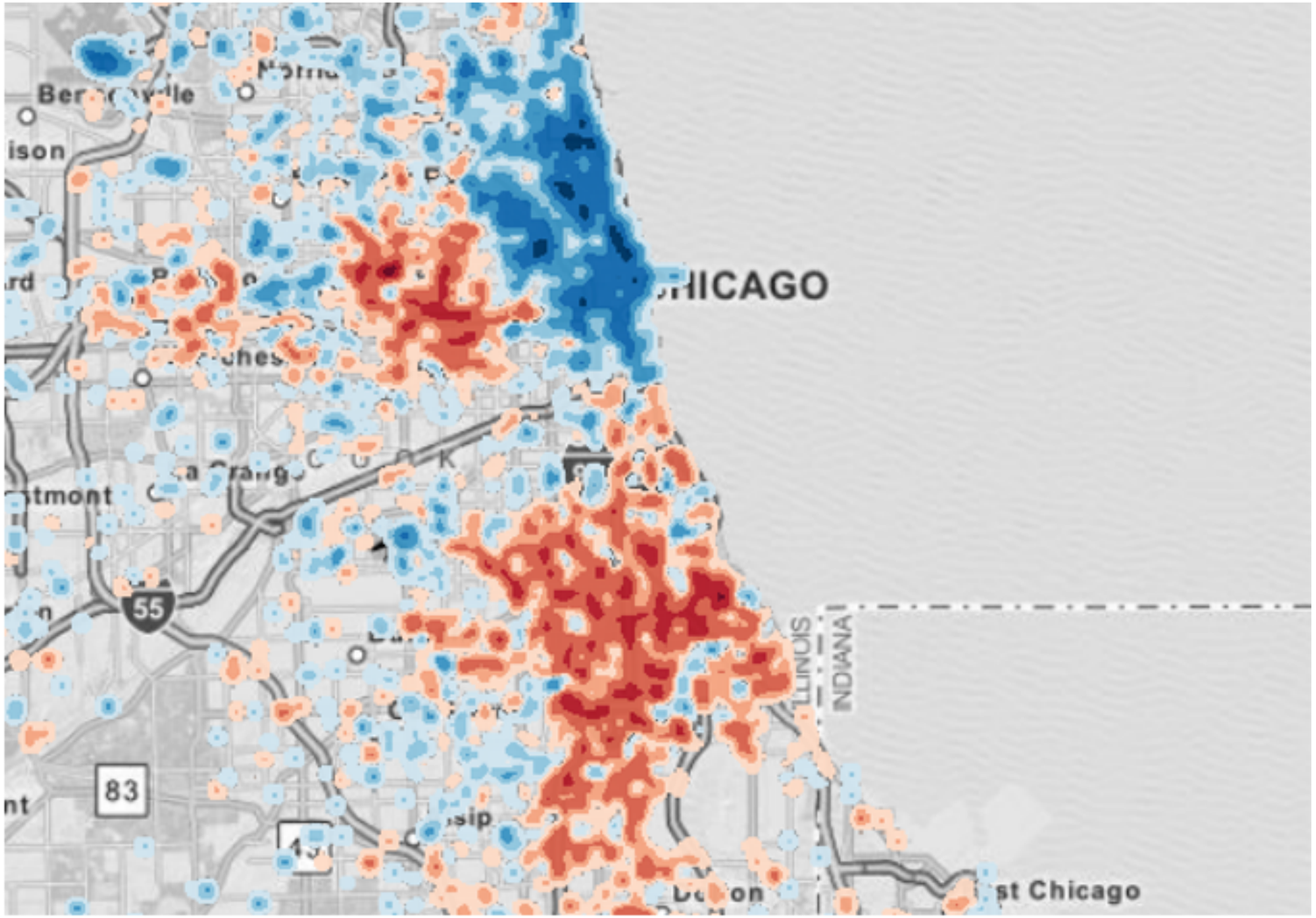
Linked view of tweets in Ensenada, Mexico

[Lins et. al, 2013]

iPhone vs. Android Map

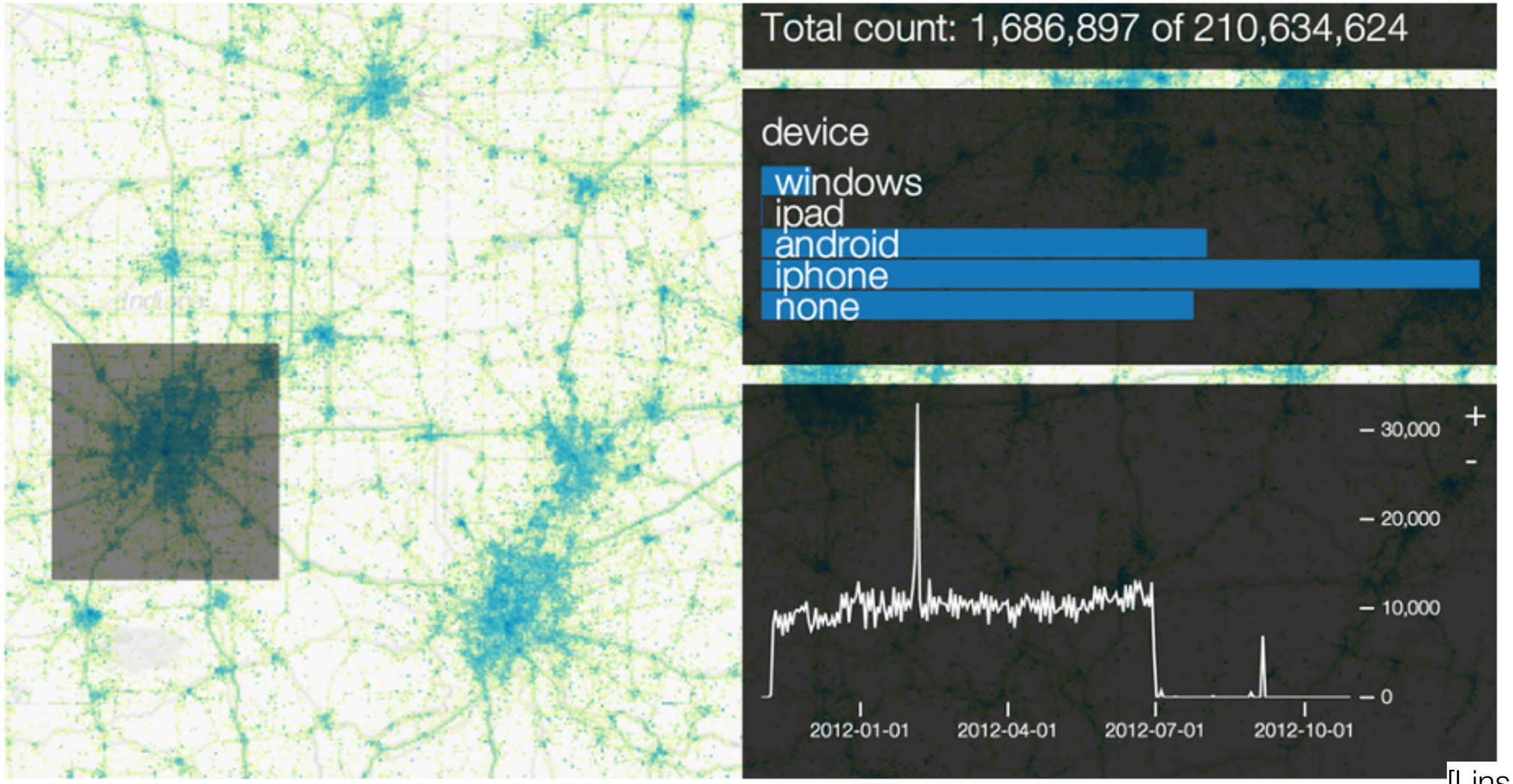

Vancouver
Seattle
Spokane
Great Falls
Minot
Grand Forks
Thunder Bay
Timmins
Sudbury
Sault Ste. Marie
Bathurst
Edmundston
Charlottetown
Saint John
Halifax
Yarmouth
Portland
Boston
Providence
Hartford
Albany
Buffalo
Toronto
Hamilton
London
Windsor
Detroit
Cleveland
Pittsburgh
Philadelphia
Baltimore
Washington, DC
Virginia Beach
Norfolk
Richmond
Charleston
Savannah
Jacksonville
Palm Coast
Orlando
Tampa
Miami
Freeport
Nassau
Bermuda
Atlanta
Birmingham
Nashville
Memphis
Louisville
Cincinnati
Columbus
Indianapolis
Chicago
Milwaukee
Madison
Green Bay
Minneapolis
Duluth
Fargo
Bismarck
Pierre
Aberdeen
Rapid City
Billings
Missoula
Helena
Bozeman
Great Falls
Kalispell
Spokane
Coeur d'Alene
Lewiston
Yakima
Kennewick
Longview
Portland
Eugene
Salem
Medford
Redding
Eureka
Chico
Reno
Sacramento
San Francisco
Oakland
San Jose
Fresno
Visalia
Bakersfield
Los Angeles
San Diego
Riverside
San Bernardino
Las Vegas
Henderson
Boise City
Idaho Falls
Pocatello
Twin Falls
Salt Lake City
Provo
Ogden
Cedar City
St. George
Cheyenne
Casper
Gillette
Rock Springs
Fort Collins
Denver
Colorado Springs
Pueblo
Grand Junction
Farmington
Santa Fe
Albuquerque
Las Cruces
Roswell
Amarillo
Lubbock
Abilene
Wichita
Wichita Falls
Oklahoma City
Tulsa
Kansas City
Omaha
Lincoln
Grand Island
Scottsbluff
North Platte
Sioux Falls
Des Moines
Cedar Rapids
Iowa City
Davenport
Rockford
Peoria
Springfield
Champaign
Evansville
Terre Haute
Fort Wayne
South Bend
Gary
Flint
Saginaw
Grand Rapids
Kalamazoo
Lansing
Toledo
Akron
Canton
Youngstown
Erie
Scranton
Wilkes-Barre
Allentown
Reading
Harrisburg
Lancaster
York
Trenton
Newark
Jersey City
Paterson
Elizabeth
Edison
Atlantic City
Dover
Wilmington
Salisbury
Ocean City
Annapolis
Frederick
Hagerstown
Charleston
Huntington
Parkersburg
Morgantown
Wheeling
Roanoke
Lynchburg
Charlottesville
Fredericksburg
Richmond
Petersburg
Hampton
Newport News
Chesapeake
Suffolk
Virginia Beach
Norfolk
Raleigh
Durham
Chapel Hill
Greensboro
Winston-Salem
High Point
Fayetteville
Wilmington
Asheville
Charlotte
Gastonia
Concord
Rock Hill
Columbia
Greenville
Spartanburg
Anderson
Myrtle Beach
Florence
Sumter
Charleston
North Charleston
Savannah
Augusta
Macon
Columbus
Albany
Valdosta
Brunswick
Jacksonville
Gainesville
Ocala
Daytona Beach
Orlando
Kissimmee
Sanford
Melbourne
Palm Bay
Vero Beach
Fort Pierce
Port St. Lucie
West Palm Beach
Boca Raton
Delray Beach
Boynton Beach
Fort Lauderdale
Hollywood
Miami
Miami Beach
Homestead
Key West
Naples
Fort Myers
Cape Coral
Sarasota
Bradenton
Tampa
St. Petersburg
Clearwater
Lakeland
Winter Haven
Bartow
Plant City
Brandon
Pasco
Hernando
Citrus
Levy
Dixie
Taylor
Jefferson
Leon
Gadsden
Liberty
Franklin
Wakulla
Tallahassee
Panama City
Panama City Beach
Pensacola
Fort Walton Beach
Destin
Crestview
Milton
Gulf Breeze
Navarre
Mobile
Baldwin
Escambia
Santa Rosa
Okaloosa
Walton
Bay
Jackson
Calhoun
Gulf
Holmes
Washington
Houston
Dothan
Enterprise
Ozark
Andalusia
Opp
Florala
Geneva
Covington
Conecuh
Monroe
Clarke
Wilcox
Dallas
Perry
Hale
Bibb
Tuscaloosa
Greene
Sumter
Pickens
Lamar
Fayette
Walker
Jefferson
Shelby
Chilton
Coosa
Tallapoosa
Chambers
Lee
Russell
Barbour
Bullock
Pike
Crenshaw
Butler
Lowndes
Montgomery
Elmore
Autauga
Marengo
Hale
Perry
Bibb
Tuscaloosa
Greene
Sumter
Pickens
Lamar
Fayette
Walker
Jefferson
Shelby
Chilton
Coosa
Tallapoosa
Chambers
Lee
Russell
Barbour
Bullock
Pike
Crenshaw
Butler
Lowndes
Montgomery
Elmore
Autauga
Marengo
Wilcox
Dallas
Perry
Hale

Zoom into Chicago



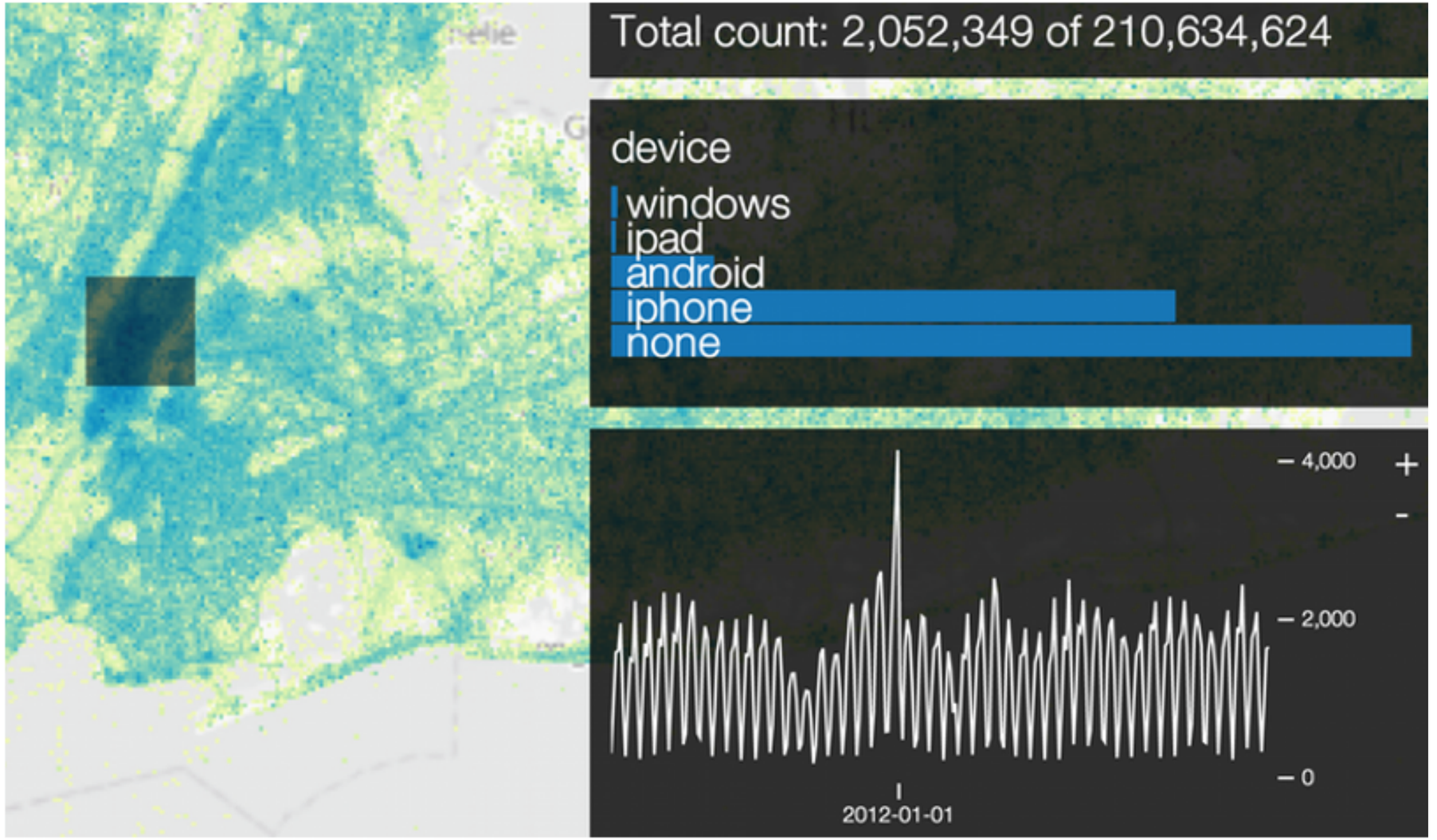
[Lins et. al, 2013]

SuperBowl in Indianapolis



[Lins et. al, 2013]

New Year's Eve in Manhattan



[Lins et. al, 2013]

Aggregations on Spatiotemporal Data

- Spatial: e.g. counting events in a spatial region (world or San Fran.)
- Temporal: e.g. queries at multiple scares (hour, day, week, month)
- Seek to address Visual Information Seeking Mantra:
- Overview first, zoom and filter, details-on-demand
- Multidimensional:
  - Latitude, Longitude, Time + **more**

Data Cube Aggregations

Relation A

Country	*Device*	*Language*
US	Android	en
US	iPhone	ru
South Africa	iPhone	en
India	Android	en
Australia	iPhone	en

Aggregation B

Country	*Device*	*Language*	*Count*
All	*All*	*All*	5

Group By on *Device, Language* C

Country	*Device*	*Language*	*Count*
All	Android	en	2
All	iPhone	en	2
All	iPhone	ru	1

Cube on *Device, Language* D

Country	*Device*	*Language*	*Count*
All	*All*	*All*	5
All	Android	*All*	2
All	iPhone	*All*	3
All	*All*	en	4
All	*All*	ru	1
All	iPhone	ru	1
All	Android	en	2
All	iPhone	en	2

Equivalent to Group By on all possible subsets of {*Device, Language*}

[Lins et. al, 2013]

Nanocube Queries

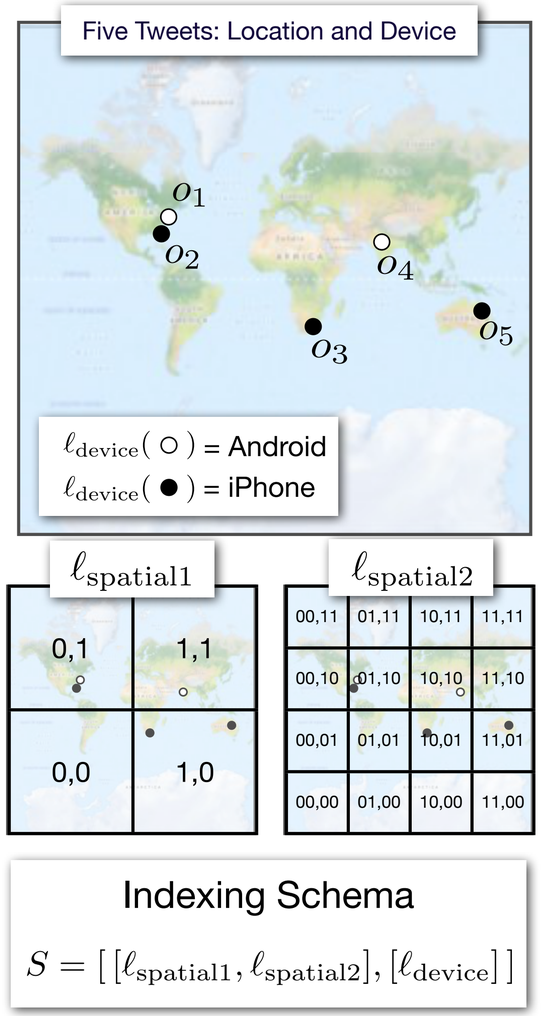
- Representing natural language queries as data cube queries

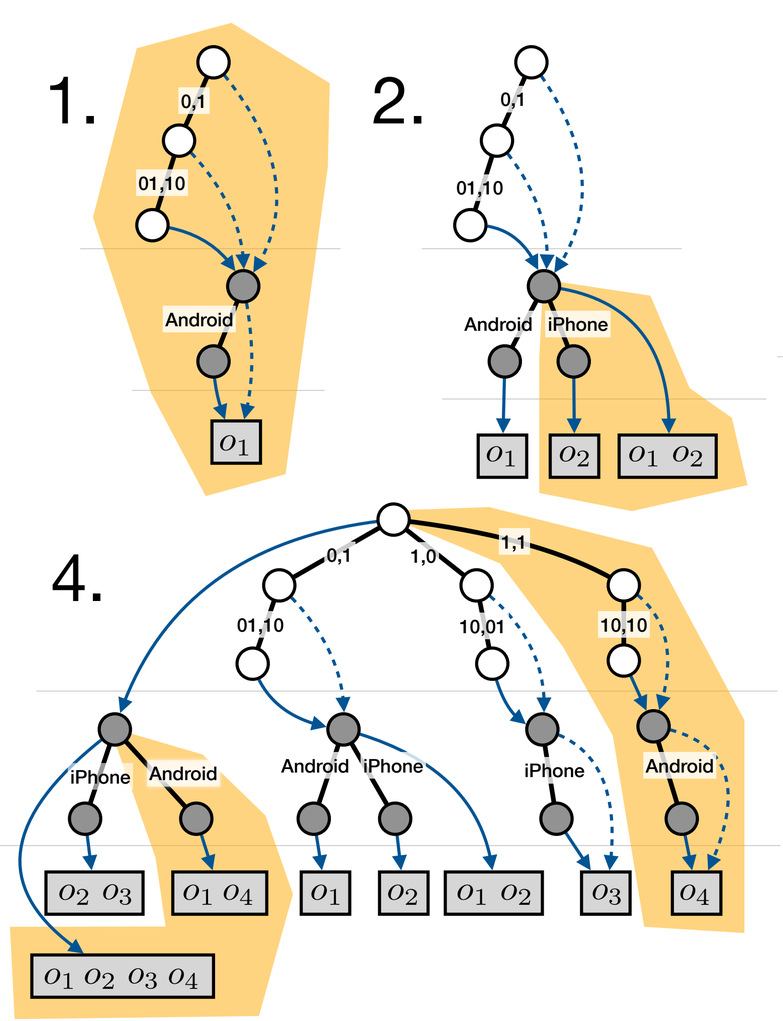
Natural language query	s		c		t		URL
count of all Delta flights	*R*	*U*	*R*	{ Delta }	*R*	*U*	/where/carrier=Delta
count of all Delta flights in the Midwest	*R*	Midwest	*R*	{ Delta }	*R*	*U*	/region/Midwest/where/carrier=Delta
count of all flights in 2010	*R*	*U*	*D*		*R*	2010	/field/carrier/when/2010
time-series of all United flights in 2009	*R*	*U*	*R*	{ United }	*D*	2009	/tseries/when/2009/where/carrier=United
heatmap of Delta flights in 2010	*D*	tile0	*R*	{ Delta }	*R*	2010	/tile/tile0/when/2010/where/carrier=Delta

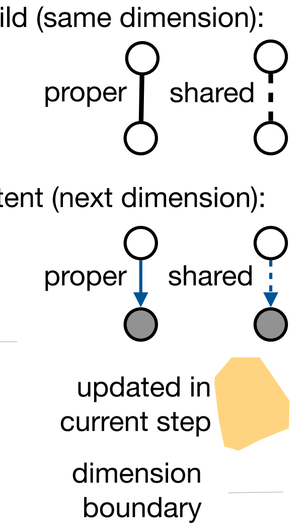
- s = space, c = category, t = time
- R = rollup, D = drill down
- <value> after RD = subset of dimension's domain, U = universe
- Note that time queries are stored in an array of cumulative counts

[Lins et. al, 2013]

Building a Nanocube



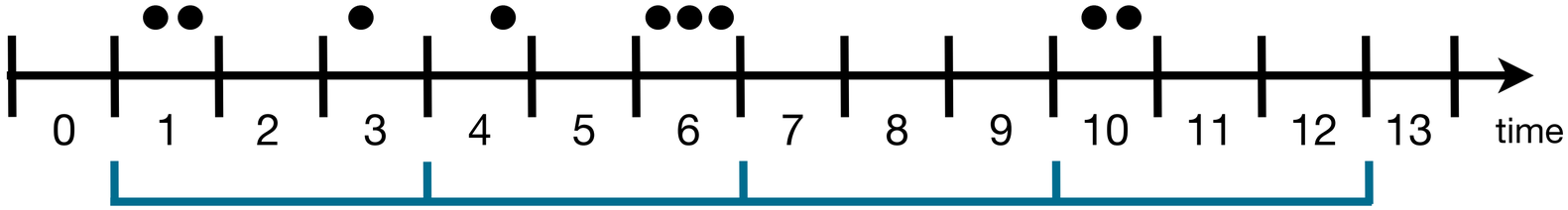




[Lins et. al, 2013]

Summed-area Table

- Every node in the previous figure stores an array of timestamped counts like this:



query/tseries/1/3/4

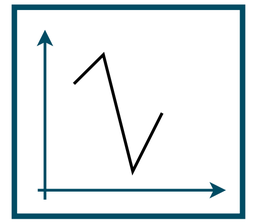
start at bin 1, use buckets of 3 bins each, and collect 4 of these buckets

solve using...

bin: 1 accum: 2	bin: 3 accum: 3	bin: 4 accum: 4	bin: 6 accum: 7	bin: 10 accum: 9

A Summed Table Sparse Representation for Counts

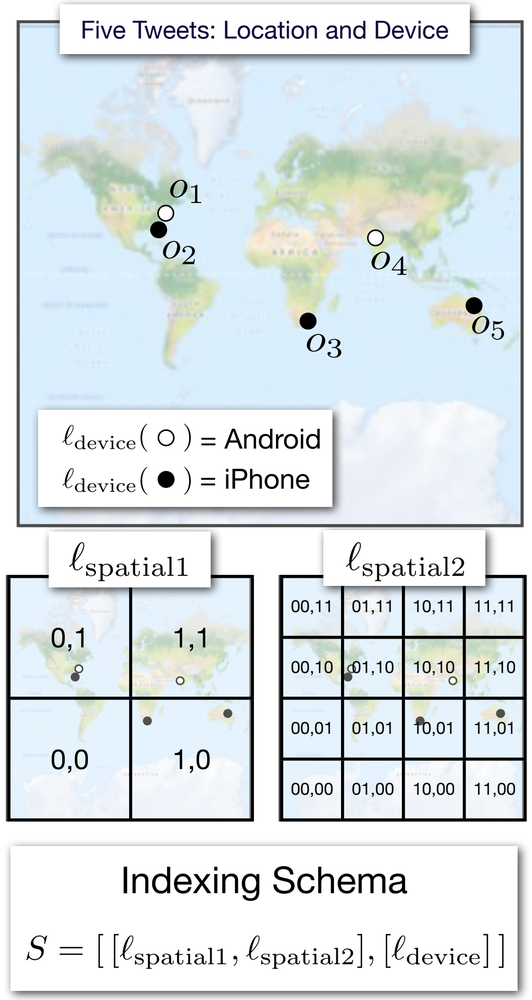
result

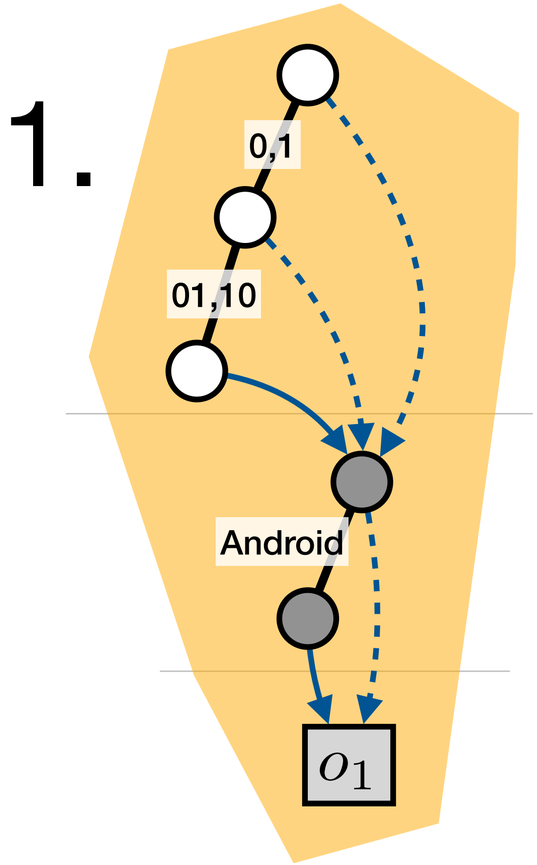


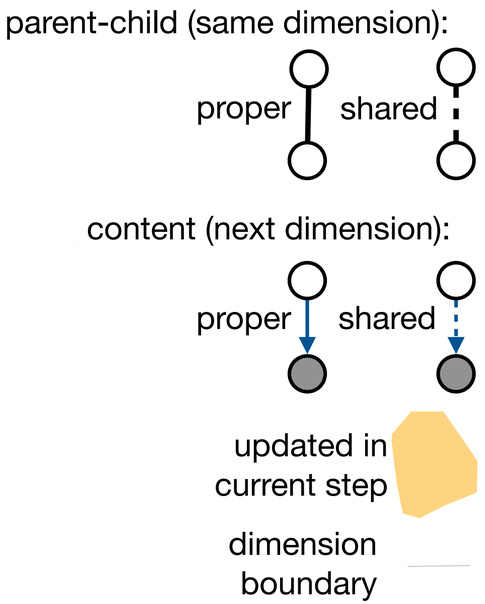
3	4	0	2

[Lins et. al, 2013]

Building a Nanocube: Step 1

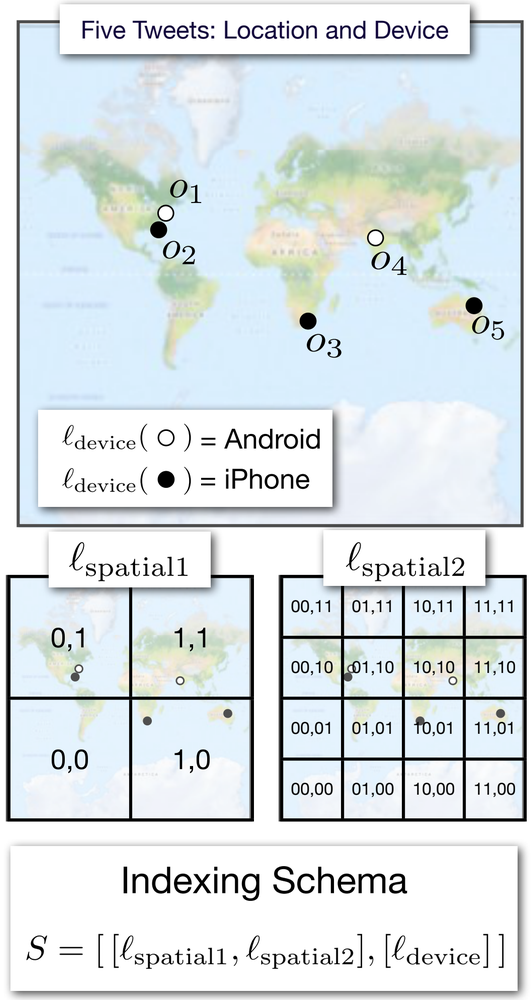




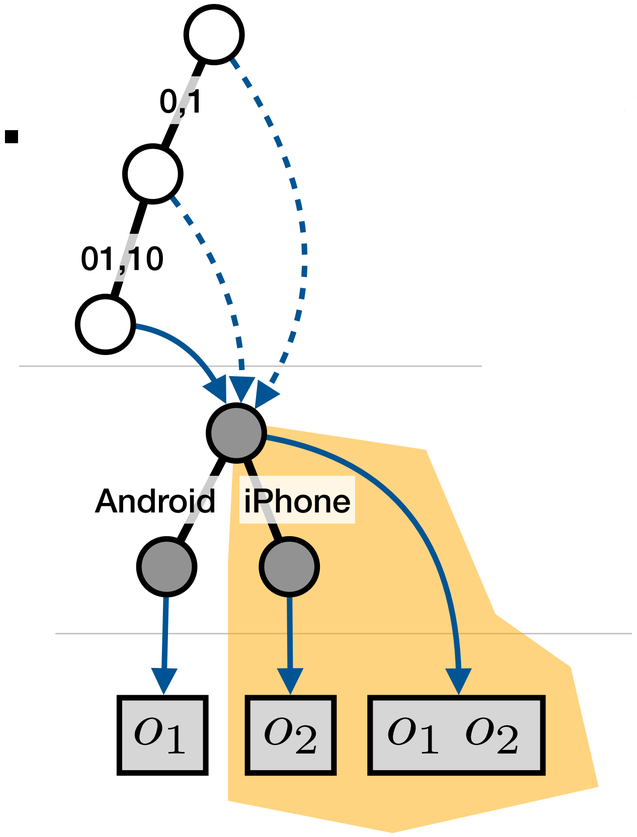


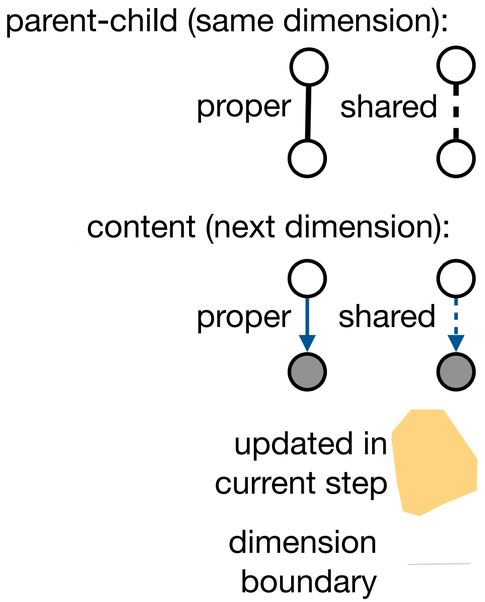
[Lins et. al, 2013]

Building a Nanocube: Step 2



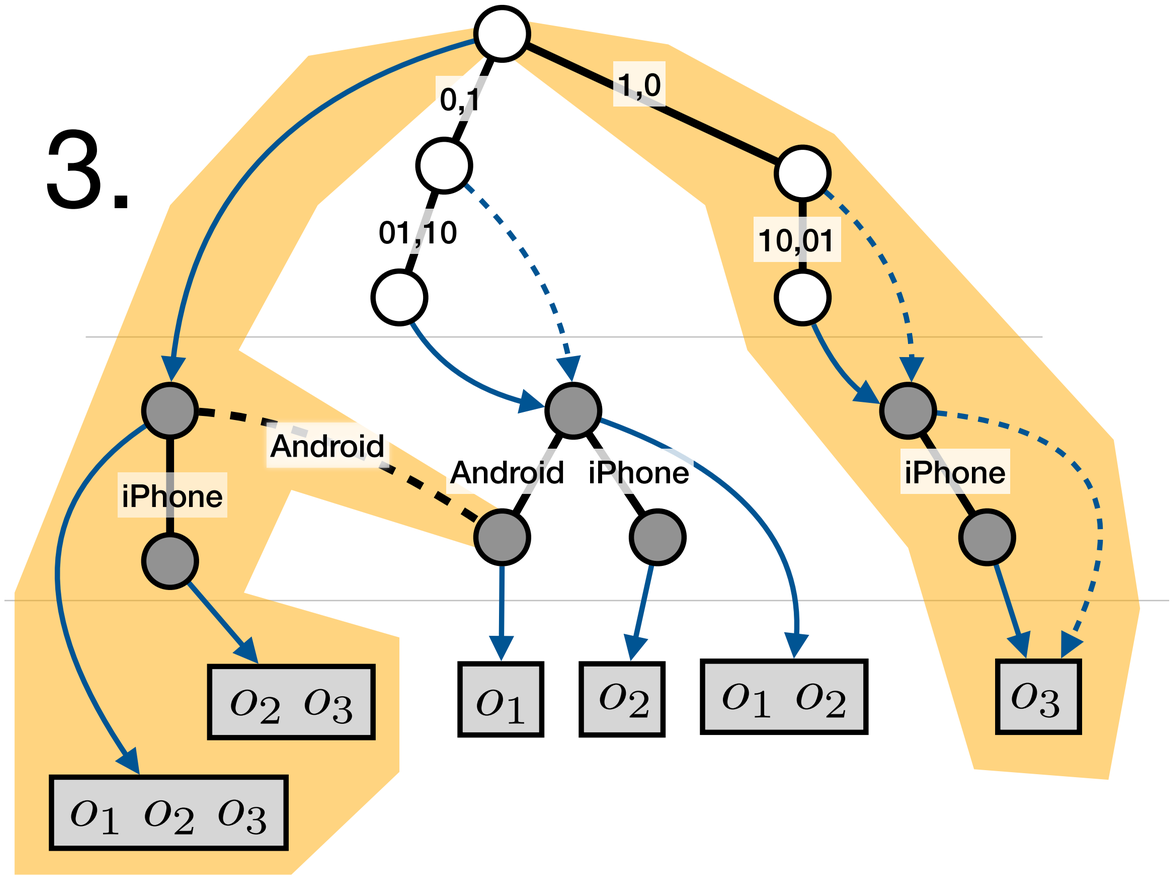
2.

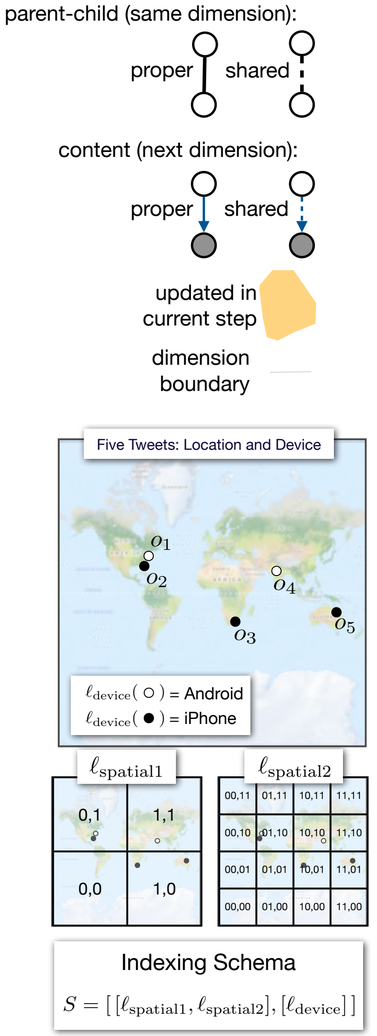




[Lins et. al, 2013]

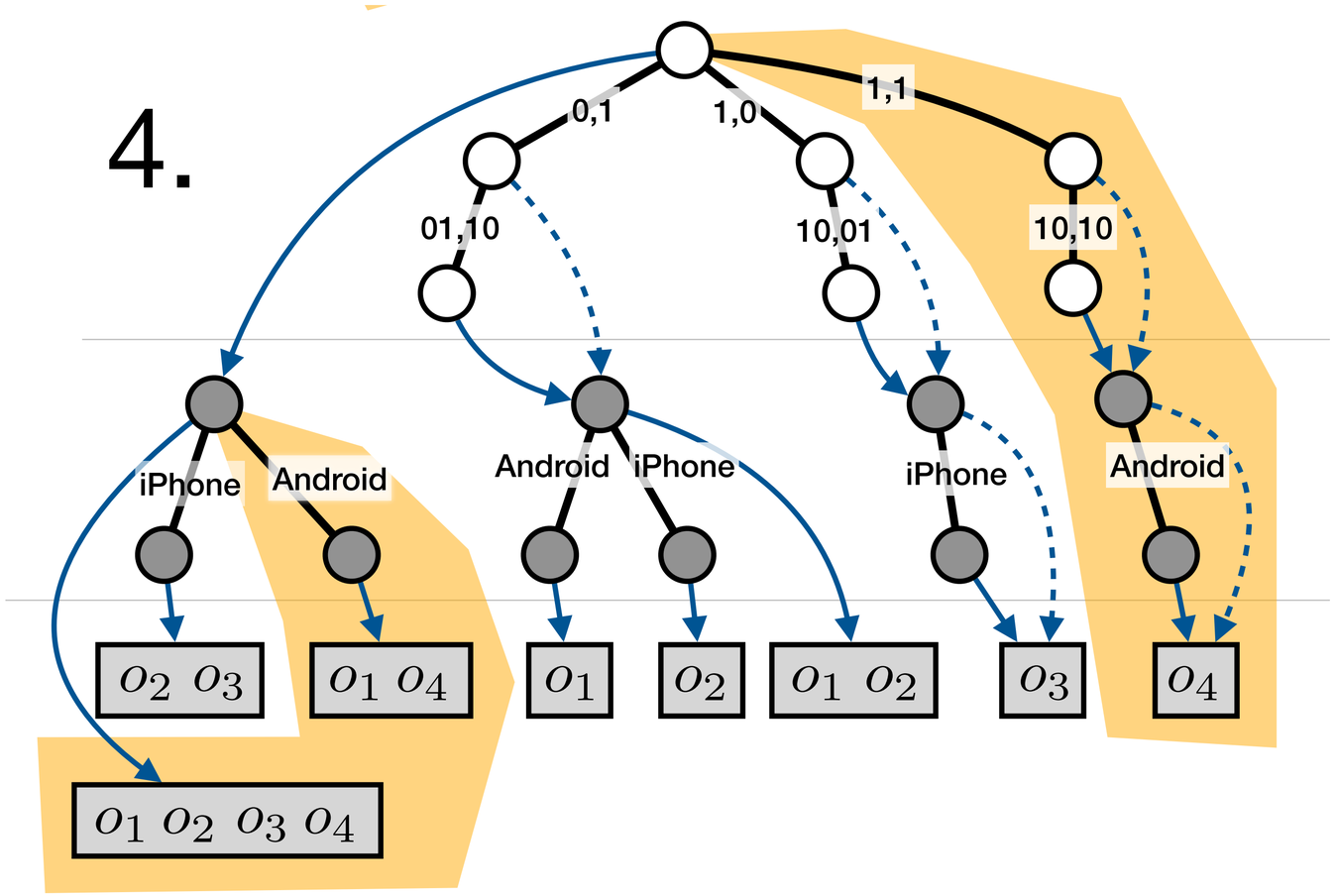
Building a Nanocube: Step 3

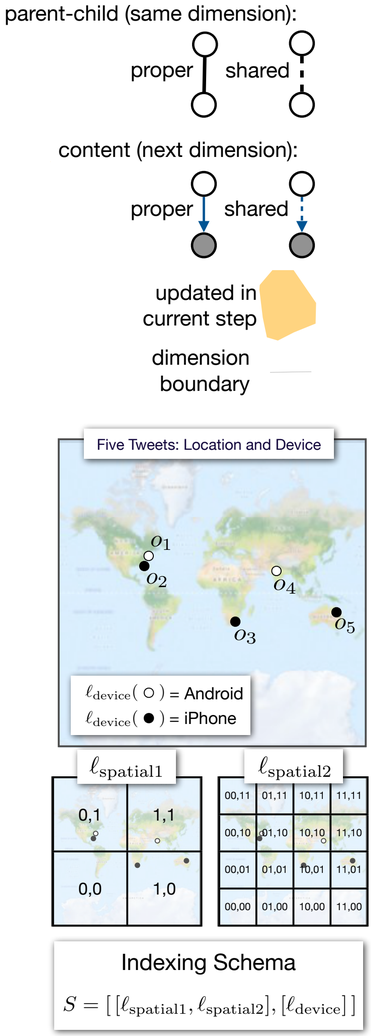




[Lins et. al, 2013]

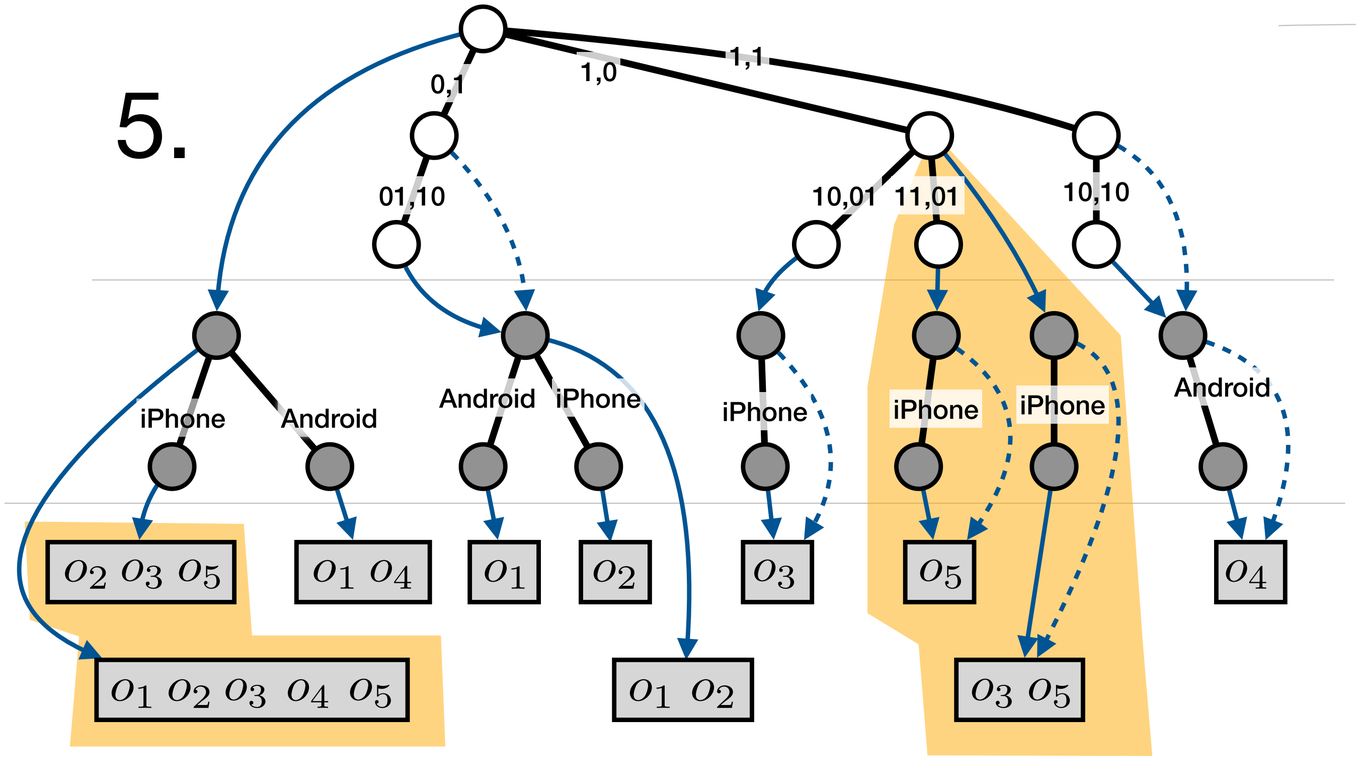
Building a Nanocube: Step 4

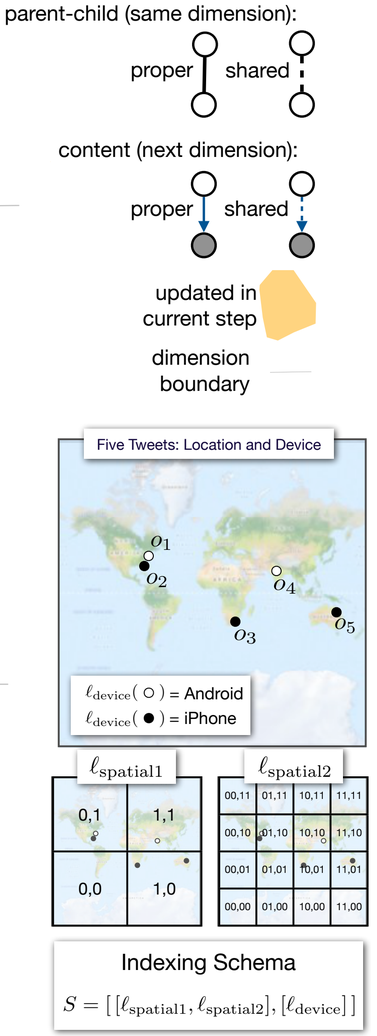




[Lins et. al, 2013]

Building a Nanocube: Step 5



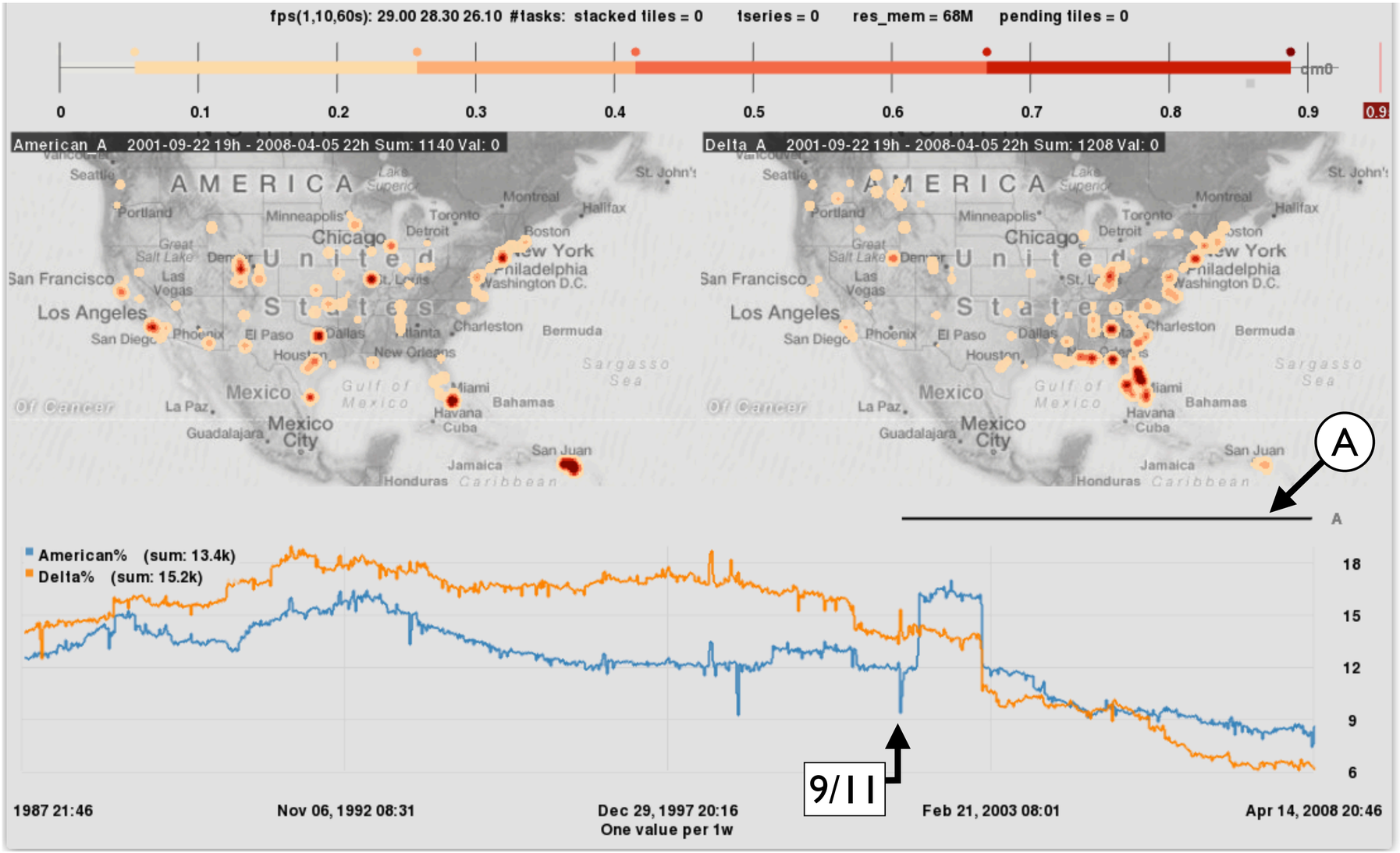


[Lins et. al, 2013]

Nanocubes Discussion

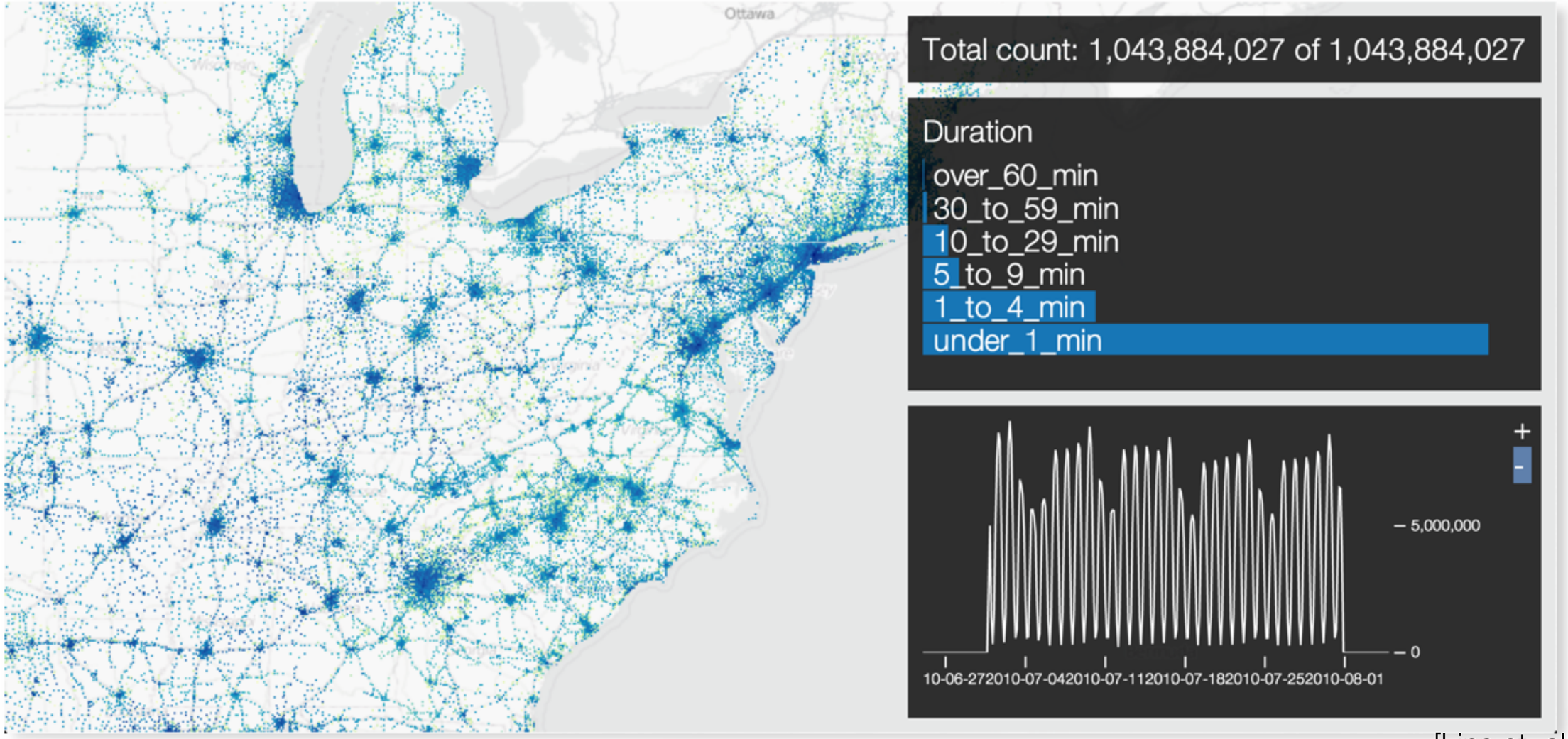
- Save space by organizing the data in a manner that takes advantage of data sparseness
- Limited to one spatial dimension, one temporal dimension
- Precompute **once**, then exploration has **low latency**

Example: American vs. Delta



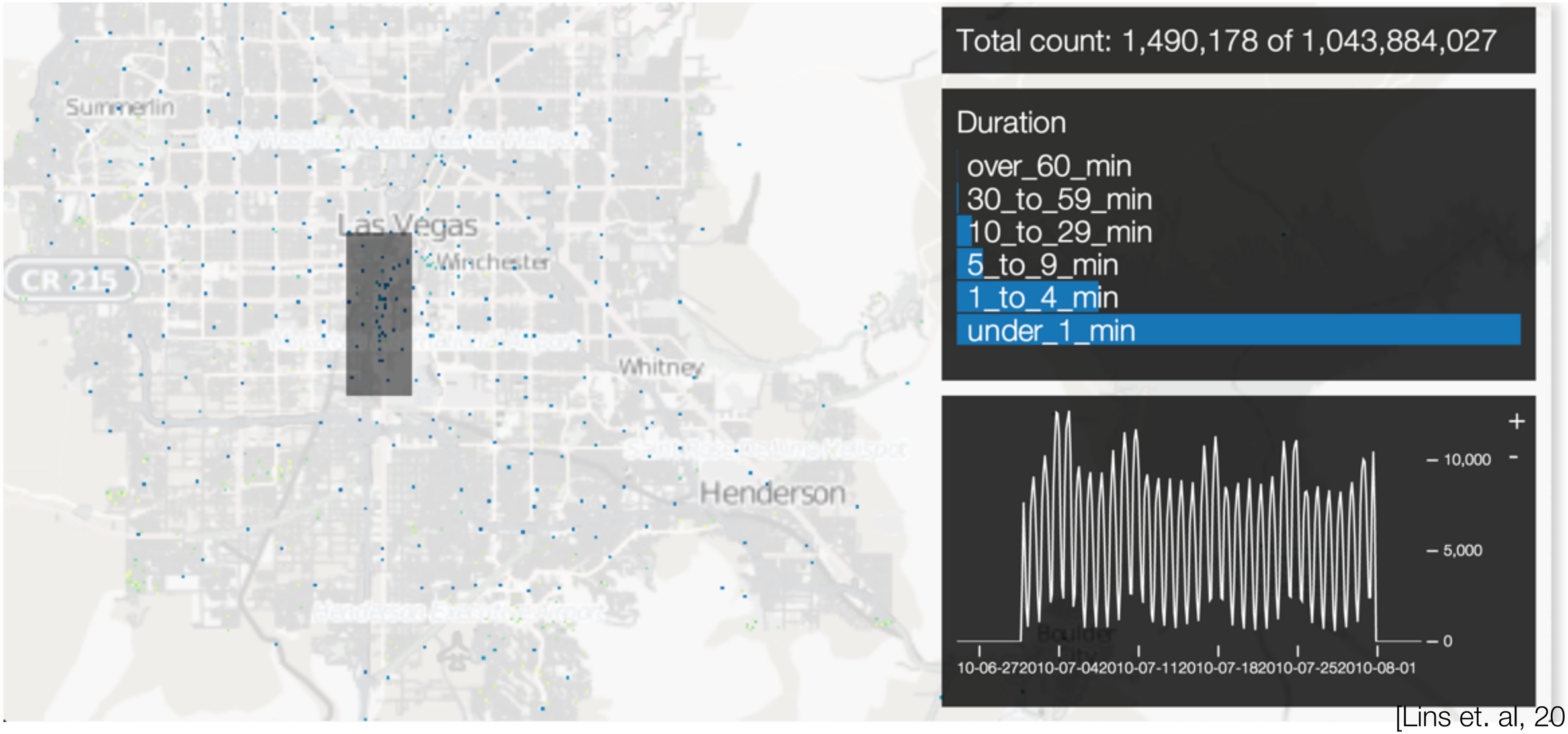
[Lins et. al, 2013]

Example: Cell Data Records



[Lins et. al, 2013]

Example: Cell Data Records

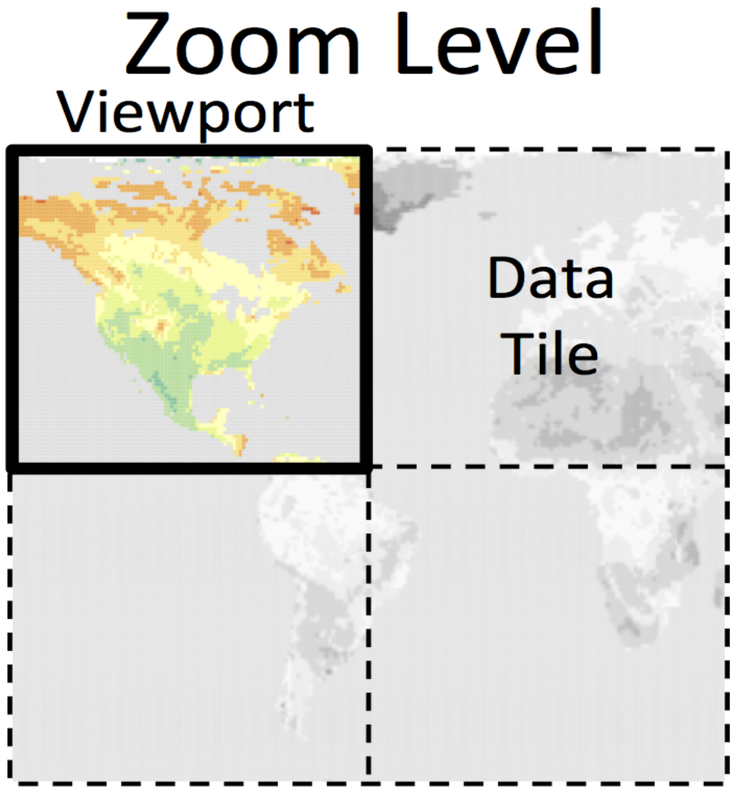


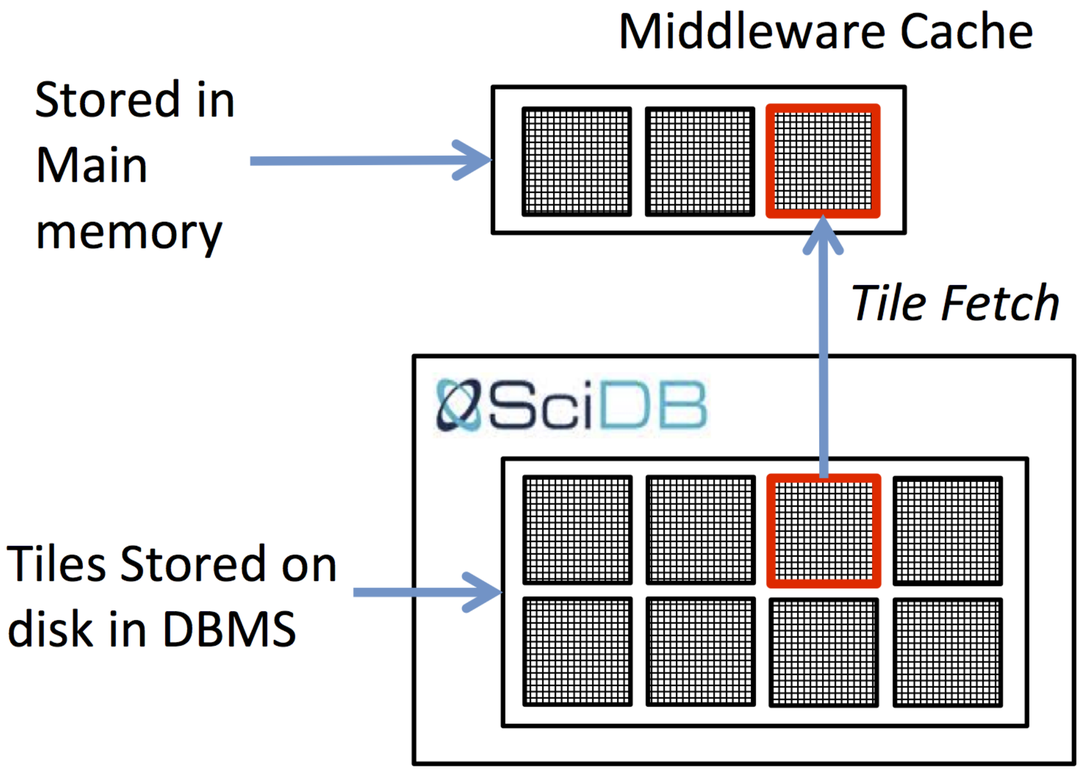
[Lins et. al, 2013]

ForeCache

L. Battle

Tile Storage in ForeCache

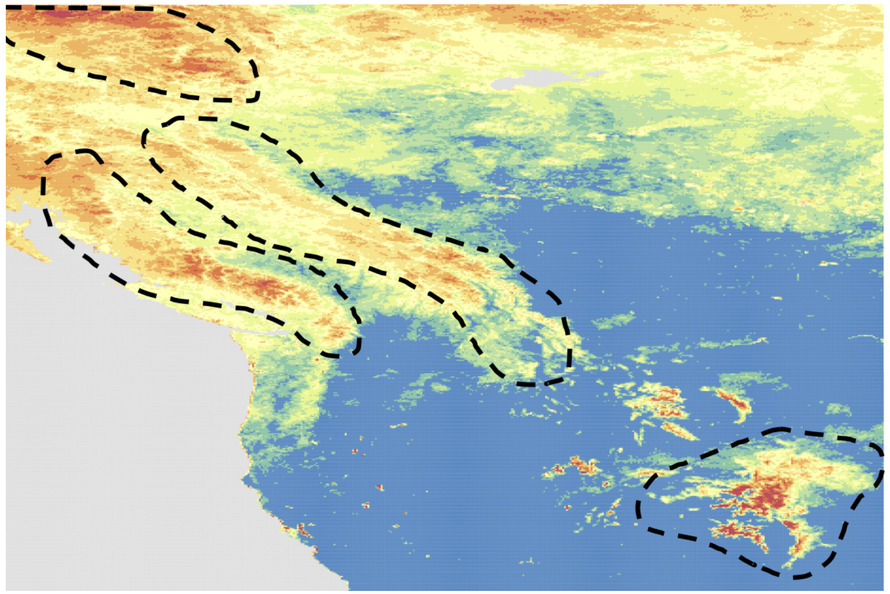


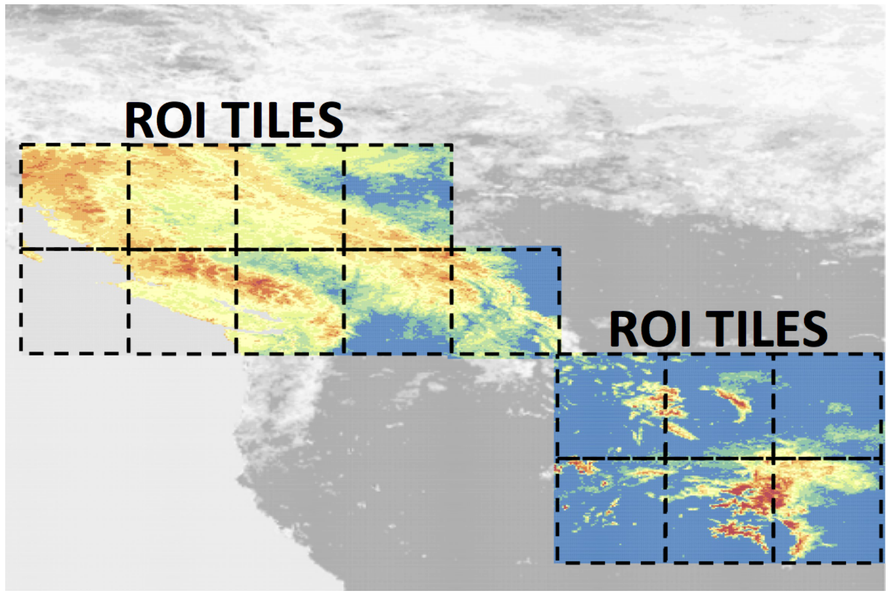


[Battle et al., 2016]

ForeCache

- Predict which tiles a user will need next and prefetch those
  - Use common patterns (zoom, pan)
  - Use regions of interest (ROIs)





[Battle et al., 2016]