Roger Moussalli, Mudhakar Srivatsa, Sameh Asaad

05 May 2015

Fast and flexible conversion of **geohash** codes to and from **lat./long.** coordinates

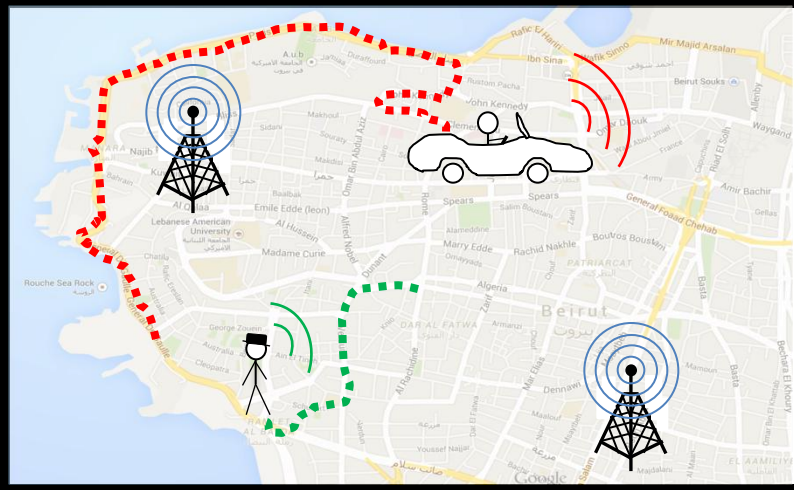
FCCM 2015
c2b2qkded

Spatiotemporal analytics

- Ubiquitous availability of location sensing devices:
  - GPS, RFID, mobile devices.

- Overwhelming majority of geospatial data is being produced by the **Internet of Things that Move** – geospatial data is increasingly spatiotemporal data:
  - Complex pattern queries can be posed on objects moving in time and space.

- Opportunities for business intelligence:
  - Traffic congestions prediction.
  - Targeted advertising.
  - Epidemic spread characterization.
  - Insurance pricing.
  - Crime pattern analysis.



25+ TBs daily logs

12+ TBs daily tweet data

1.75 Billion smartphone users worldwide

100s of millions GPS-enabled devices sold annually

- CPUs unable to keep up with online and offline processing of skyrocketing amounts of data.

Geohash: hierarchical geocoding

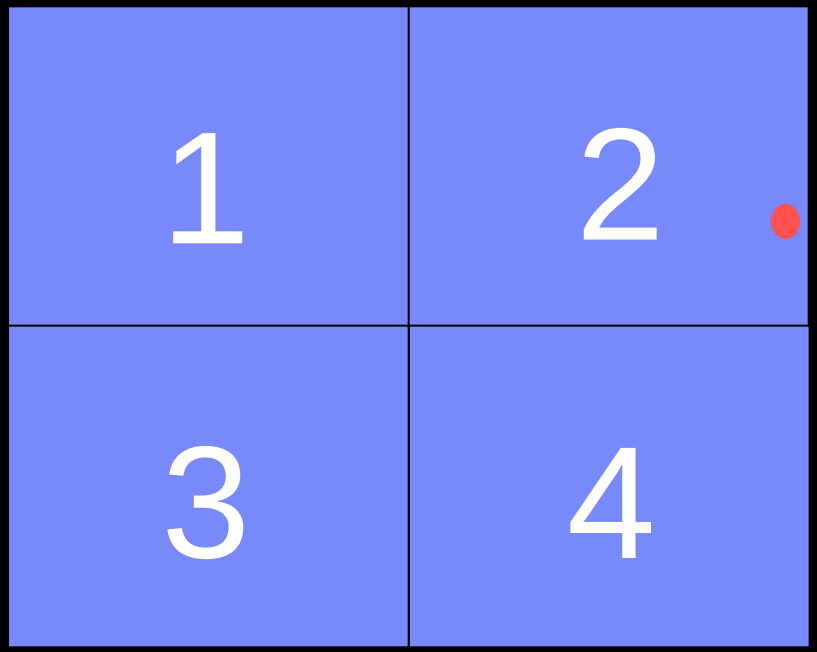
- Algorithm introduced in 2009:
  - http://geohash.org (/c2b2qkded)

- Hierarchically divide space into grids.

Geohash: hierarchical geocoding

- Algorithm introduced in 2009:
  - http://geohash.org (/c2b2qkded)

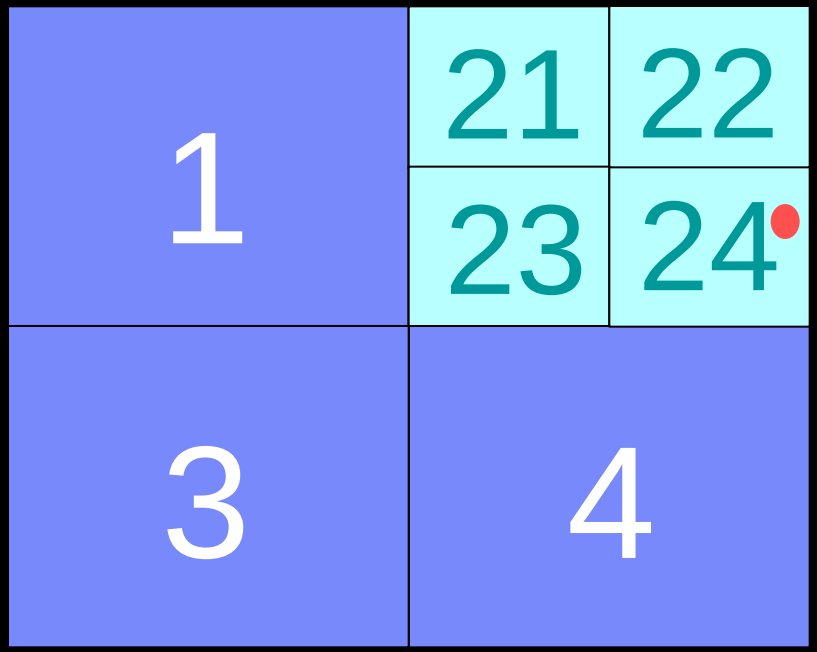
- Hierarchically divide space into grids.



Geohash: hierarchical geocoding

- Algorithm introduced in 2009:
  - http://geohash.org (/c2b2qkded)

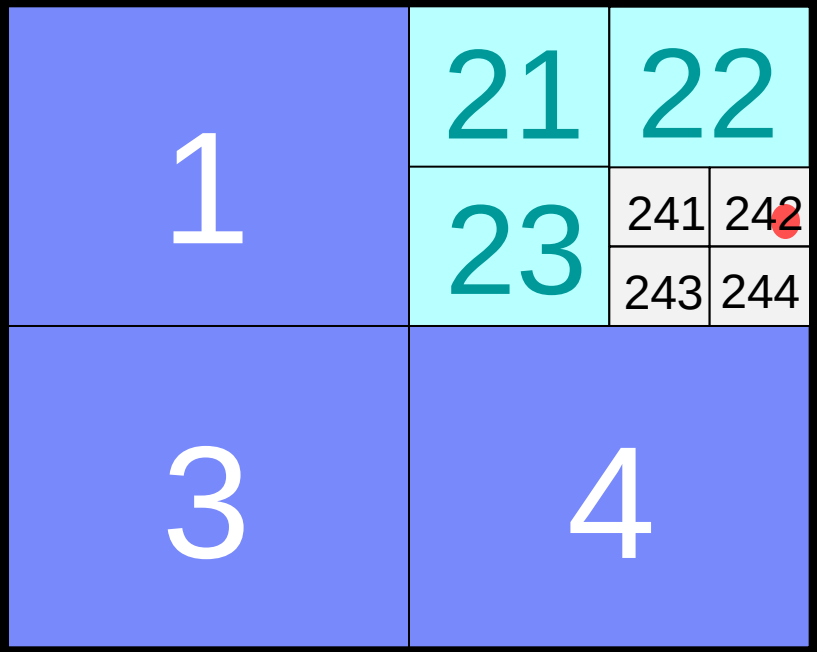
- Hierarchically divide space into grids.



Geohash: hierarchical geocoding

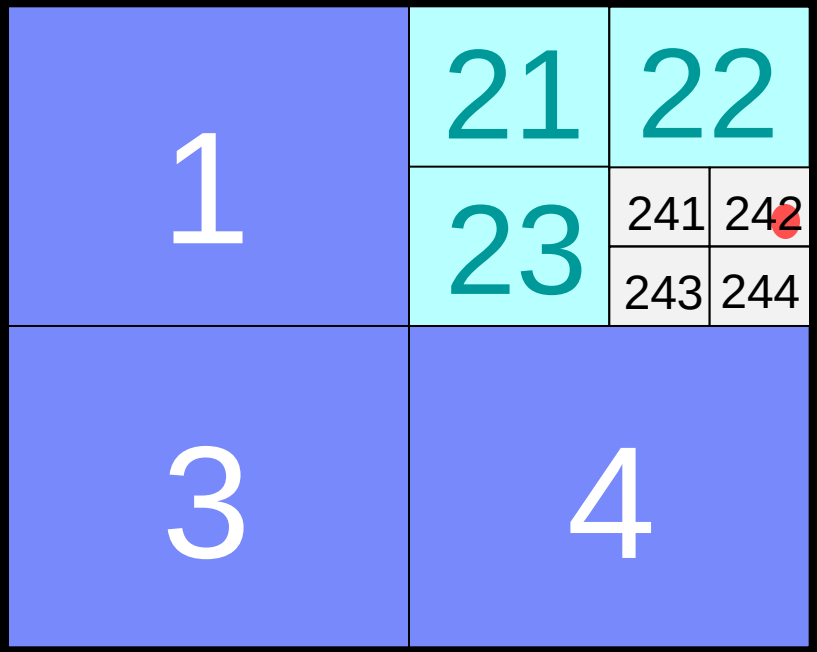
- Algorithm introduced in 2009:
  - http://geohash.org (/c2b2qkded)

- Hierarchically divide space into grids.



Geohash: hierarchical geocoding

- Algorithm introduced in 2009:
  - http://geohash.org (/c2b2qkded)
- Hierarchically divide space into grids.
- **Geohash codes properties:**
  - Support for hierarchical regions.
  - Arbitrary precision: remove characters from the end of the code to reduce size and precision.
  - Near places often (but not always) present similar prefixes (simple proximity estimation).
- **Main uses of geohashes:**
  - Geotagging.
  - Efficient database indexing.



Problem statement

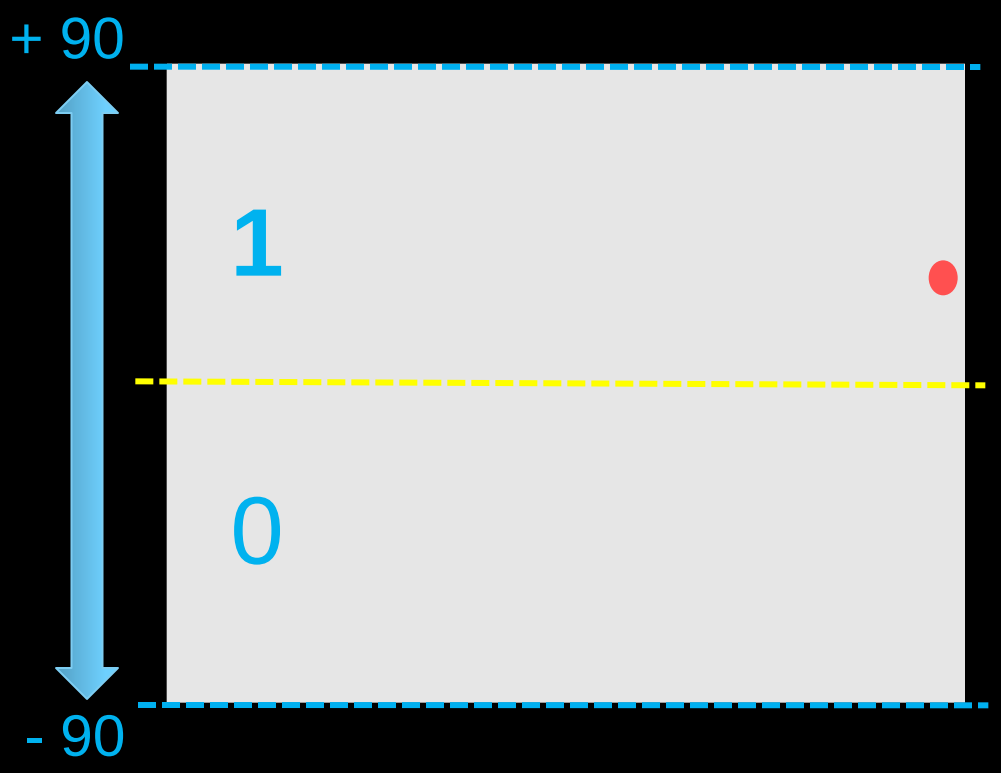
- **Geohash codes do not replace lat./long. coordinates:**
  - Format of disseminated data.
  - Dependency of legacy algorithms on lat./long. coordinates.

- **New algorithms are developed to operate in the geohash domain.**

- **The (bit-serial) conversion of geohash codes to and from lat./long. is a highly frequent problem, and is computationally demanding** (double-precision FP)**:**
  - MongoDB, Apache Accumulo, IBM Streams and SPSS (among others).

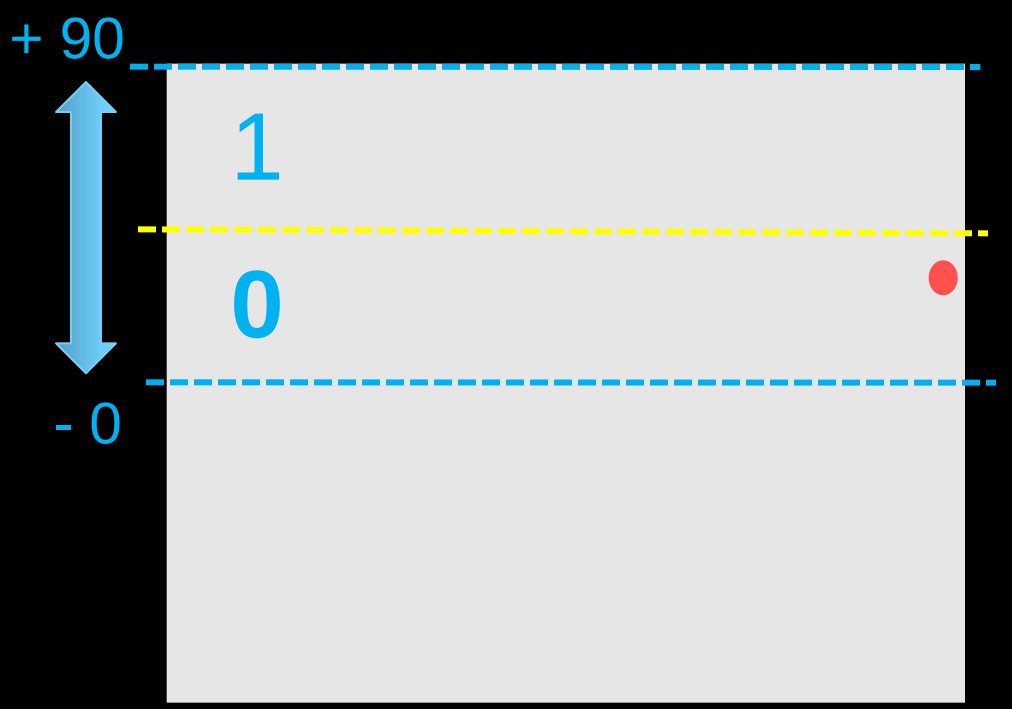
Geohash code generation (latitude)

- Geohash latitude bits:
  1



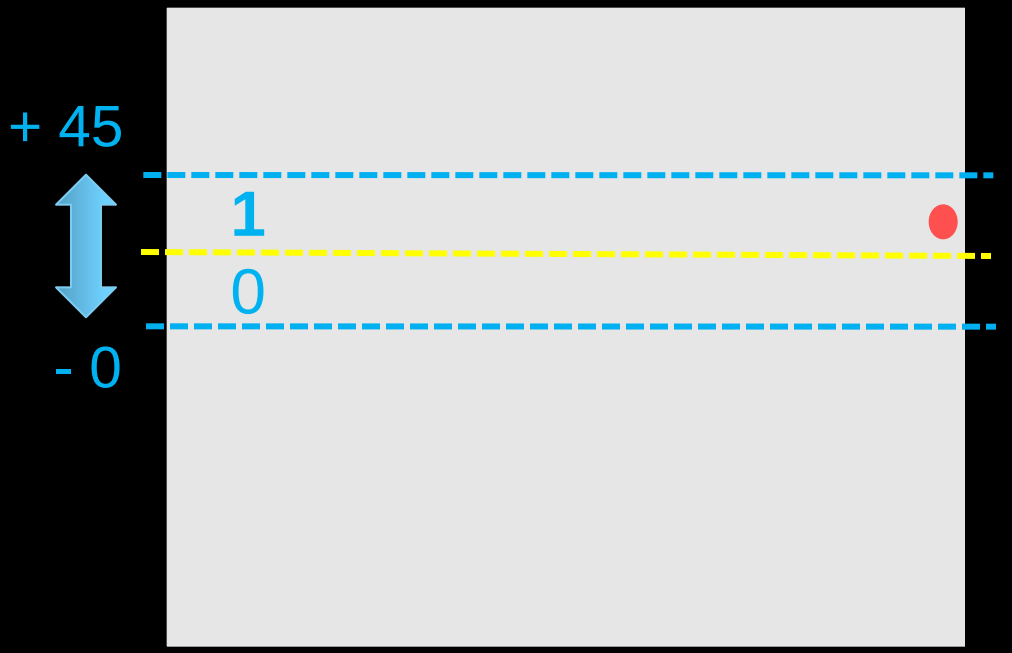
Geohash code generation (latitude)

- Geohash latitude bits:
  1, 0



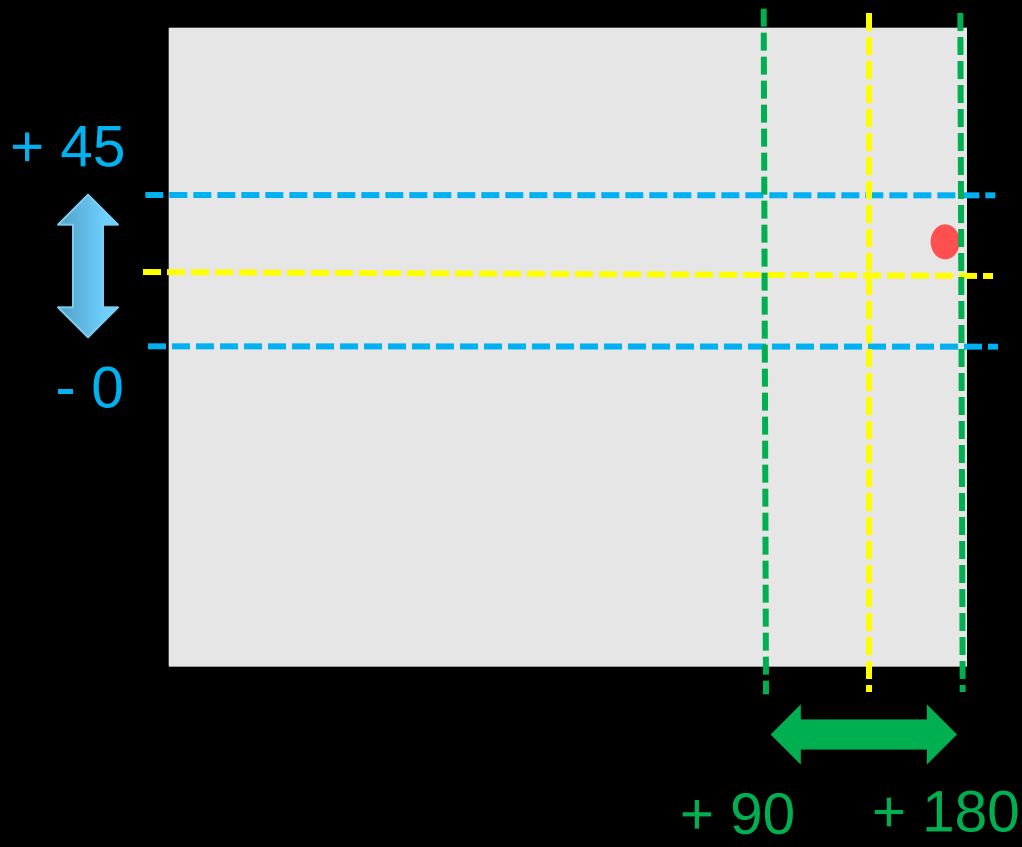
Geohash code generation (latitude)

- Geohash latitude bits:
  1, 0, 1



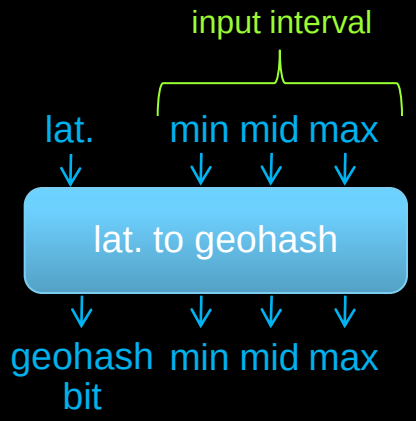
Geohash code generation

- Geohash latitude bits:
  1, 0, 1
- Geohash longitude bits:
  1, 1, 1
- Geohash code:
  111011

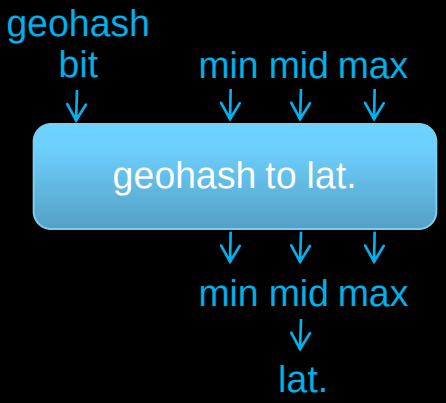


Single step conversion block

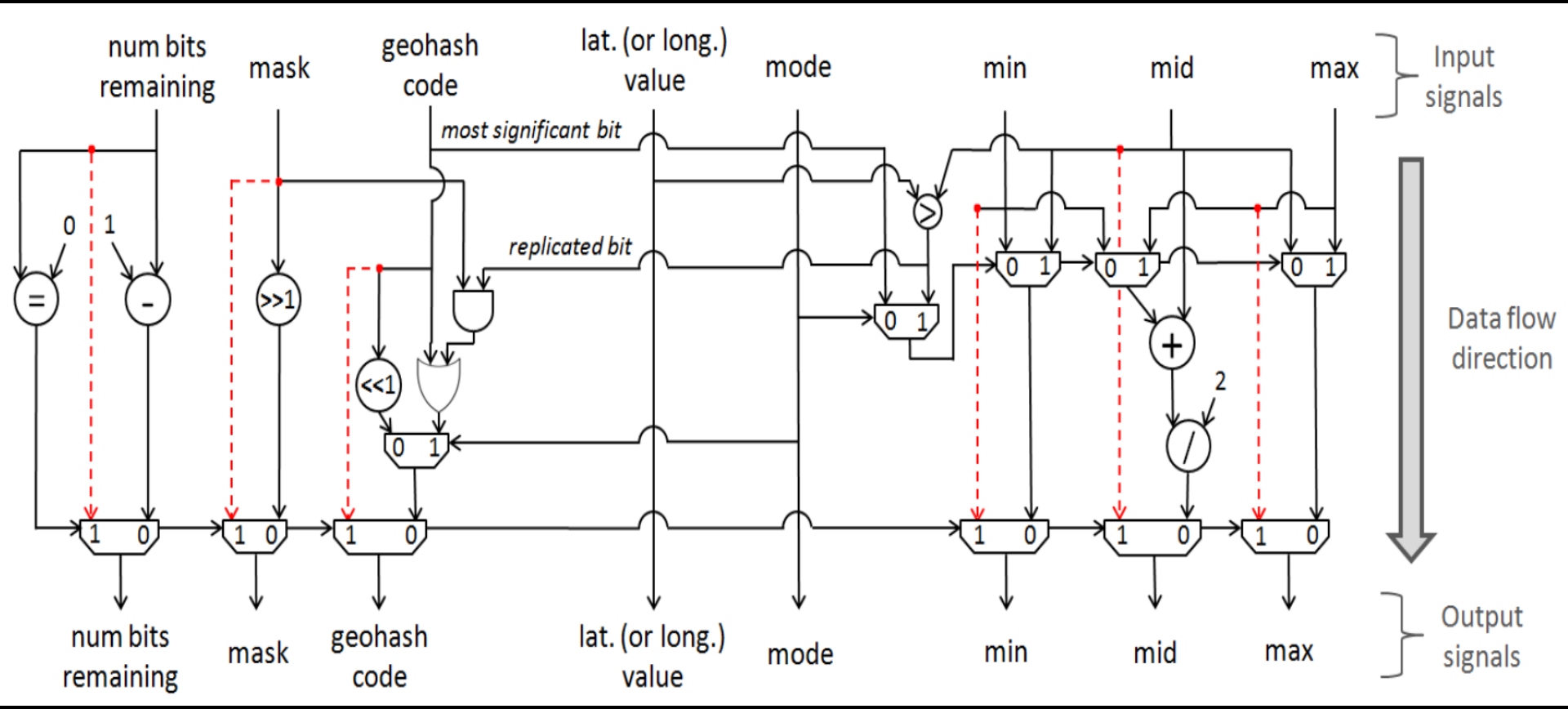
- **Latitude (or longitude) to geohash**
  - Input: Latitude (or longitude).
    Interval (min, mid, max).
  - Output: Geohash bit.
    New interval.



- **Geohash to latitude (or longitude)**
  - Input: Geohash bit.
    Latitude (or longitude) value.
    Interval (min, mid, max).
  - Output: New interval.
    Latitude (or longitude) value = new interval mid.



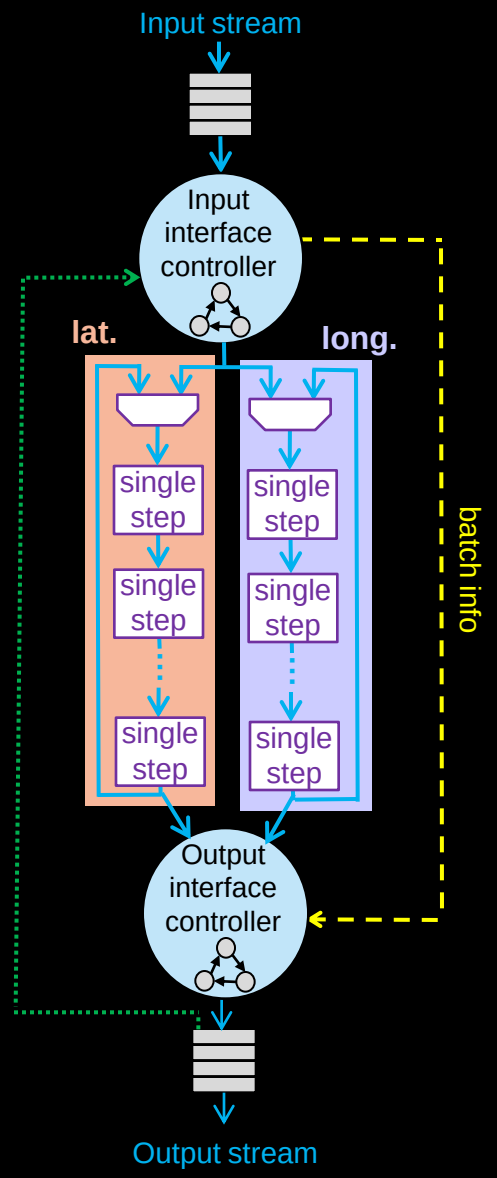
Single step conversion block



Flexible, high-throughput converter

- Design-time flexibility:
  - Max geohash precision.
  - Num deployed stages (single step blocks).
  - Supported transfer-side geohash sizes.
  - Engineering details:
    - Stream interface width.
    - Further Pipelining.

- Run-time flexibility:
  - Mode: geo-to-ll, ll-to-geo.
  - Geohash precision.
  - Geohash transfer size.



Conversion by lookup

- Pre-meditated (offline) conversion table.
- Relax requirements on compute resources (DSPs).
- Table size = $\sum_{i=1}^{N} 2^i = 2^{N+1} - 2$
- Address in table =

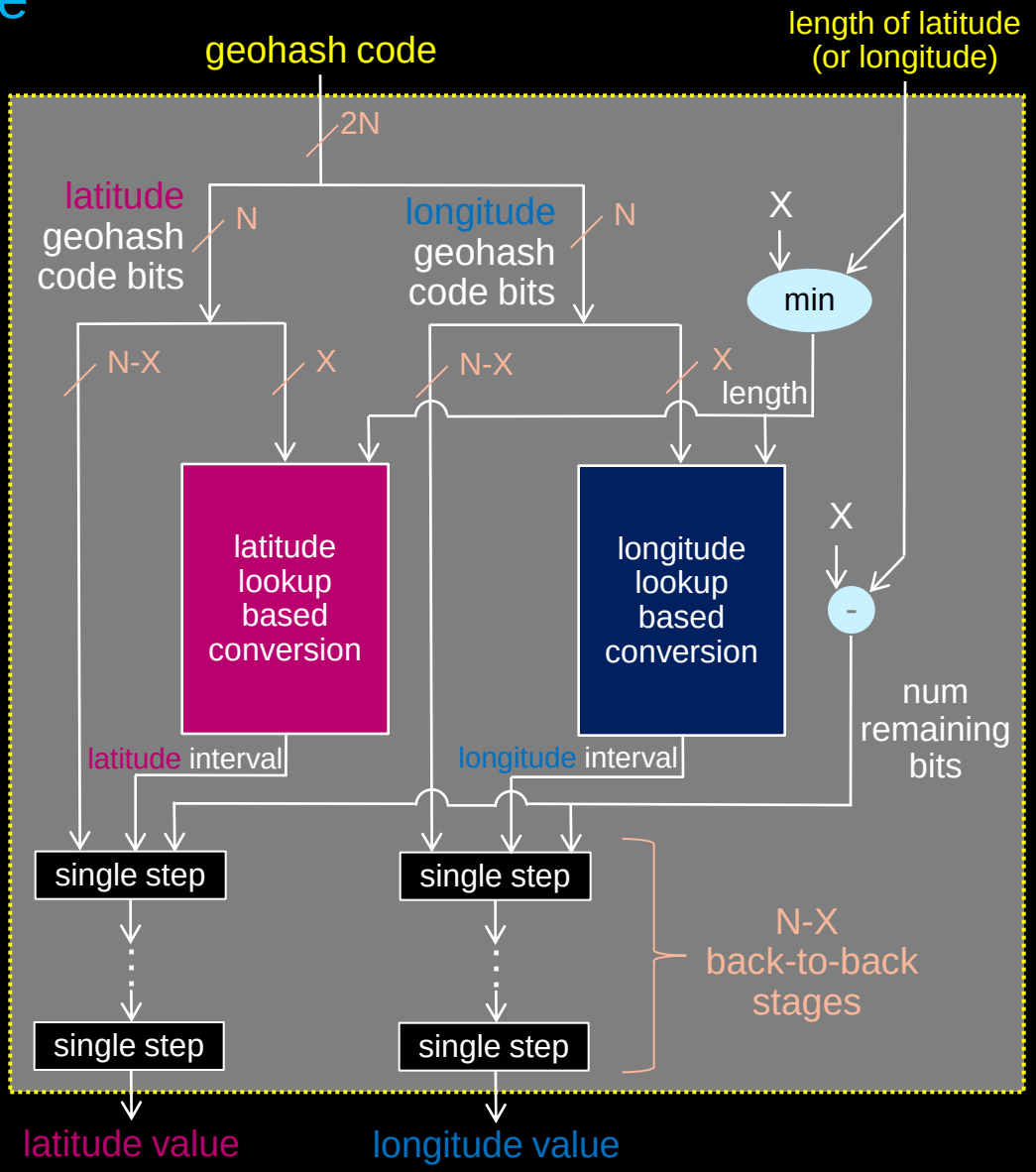
  longitude geohash code+offset
- Offset = $\sum_{i=1}^{L-1} 2^i$

longitude

address	longitude geohash	min	mid	max	
0	0	-180	-90	0	conversions for longitude of length **1 bit**
1	1	0	90	180	
2	00	-180	-135	-90	conversions for longitude of length **2 bits**
3	01	0	45	90	
4	10	-90	-45	0	
5	11	90	135	180	
6	000	-180	-157.5	-135	conversions for longitude of length **3 bits**
7	001	0	22.5	45	

Combining lookup with compute

- Flexible resource-friendly solution.
- Lookup tables process at most X geohash bits (each).
- N-X bits are processed by compute single step blocks:
  - Multiple passes through the compute stages is allowed.



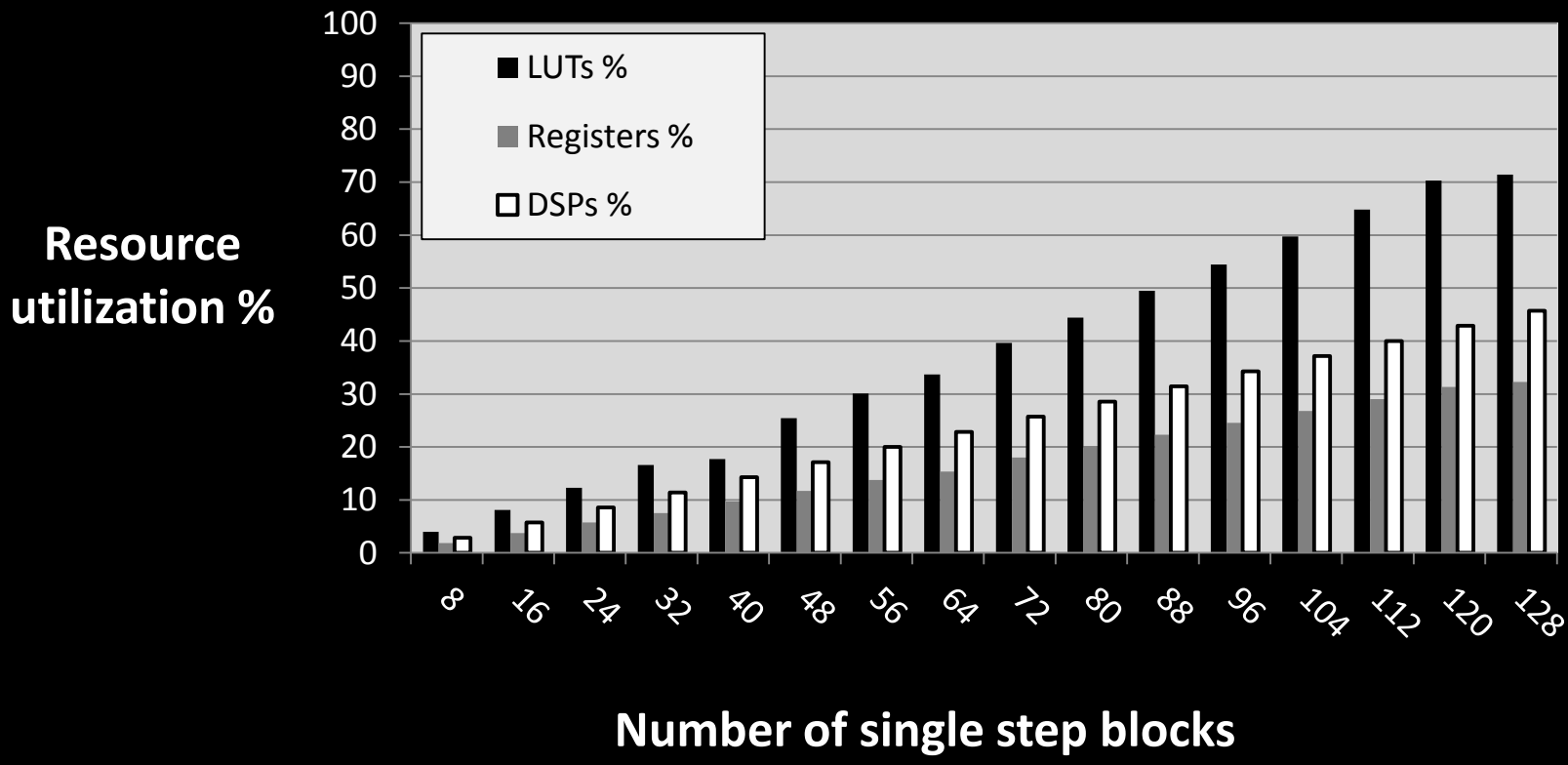
Experimental evaluation

- Hardware platform:
  - Pico M505 board.
  - PCIe gen2 x8.
  - Xilinx Kintex 7 325T @250MHz.
  - Vivado 2014.2 .
  - **End-to-end performance**: host RAM -> FPGA -> host RAM.

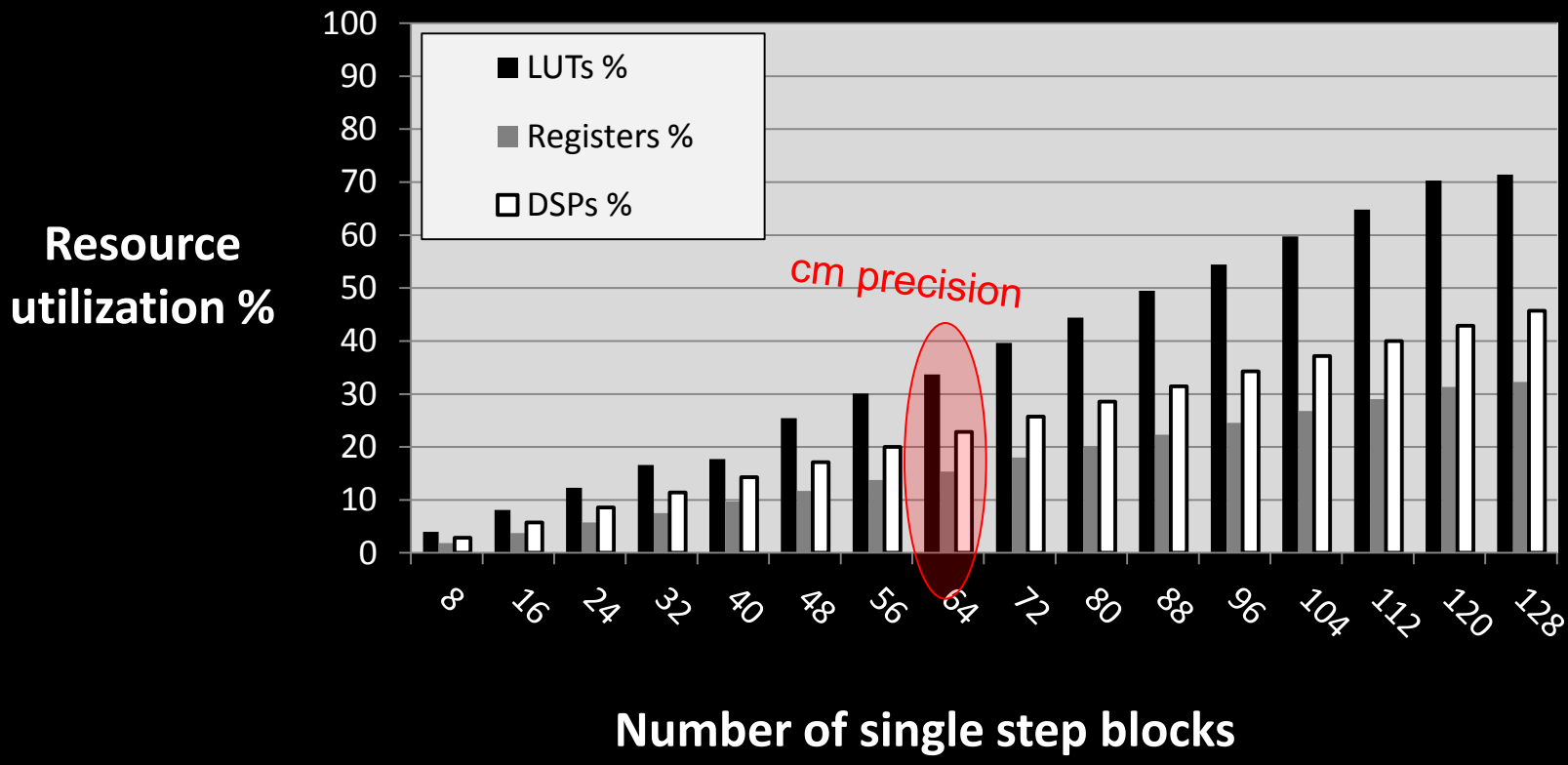
- Software framework:
  - Multi-threaded converter from IBM Streams and SPSS.
  - Single socket 8-core x2 Hyper-Threads Intel Xeon @2.5GHz.
  - 32GB RAM.

- Metric: $\frac{MConversions}{s}$

Design space exploration

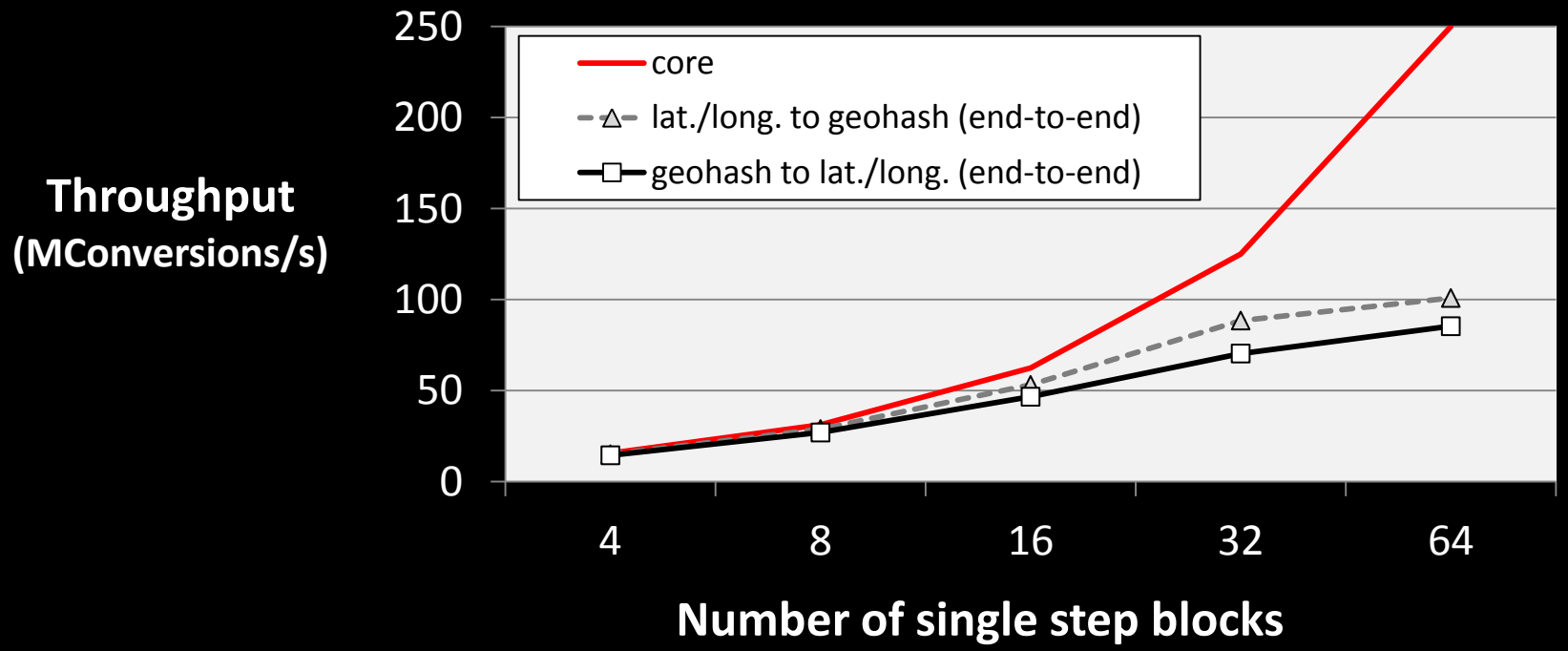


Design space exploration



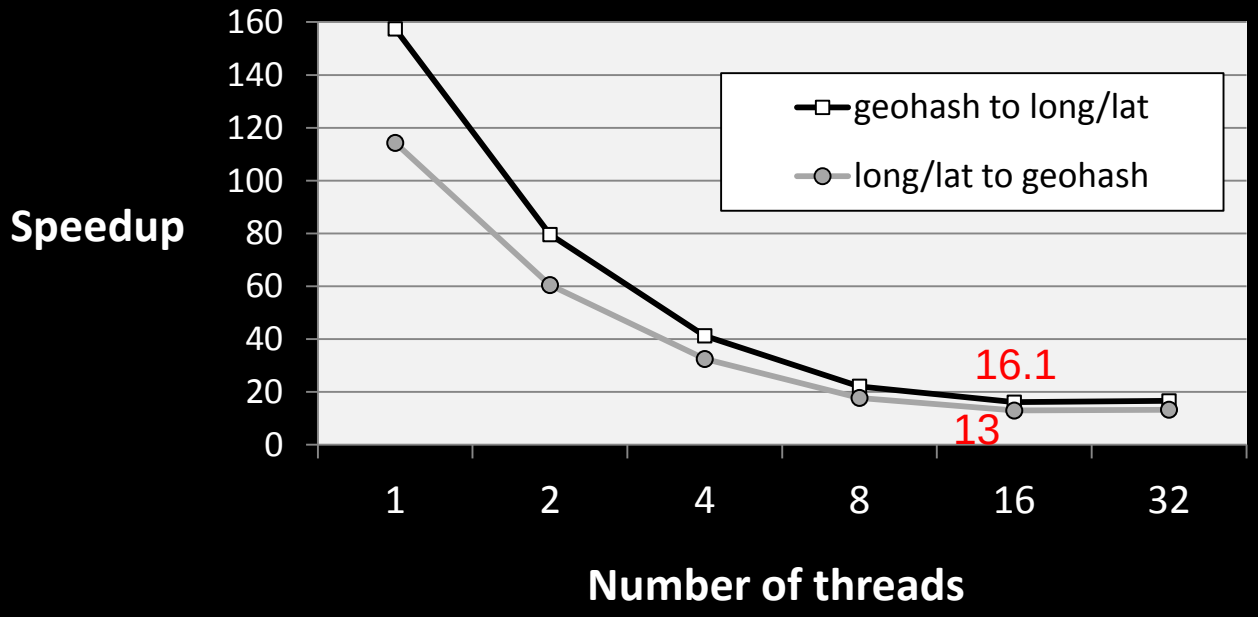
End-to-end throughput: PCIe gen2 x8 limitations

- 64-bit geohashes.
- Batch of 100M conversions.



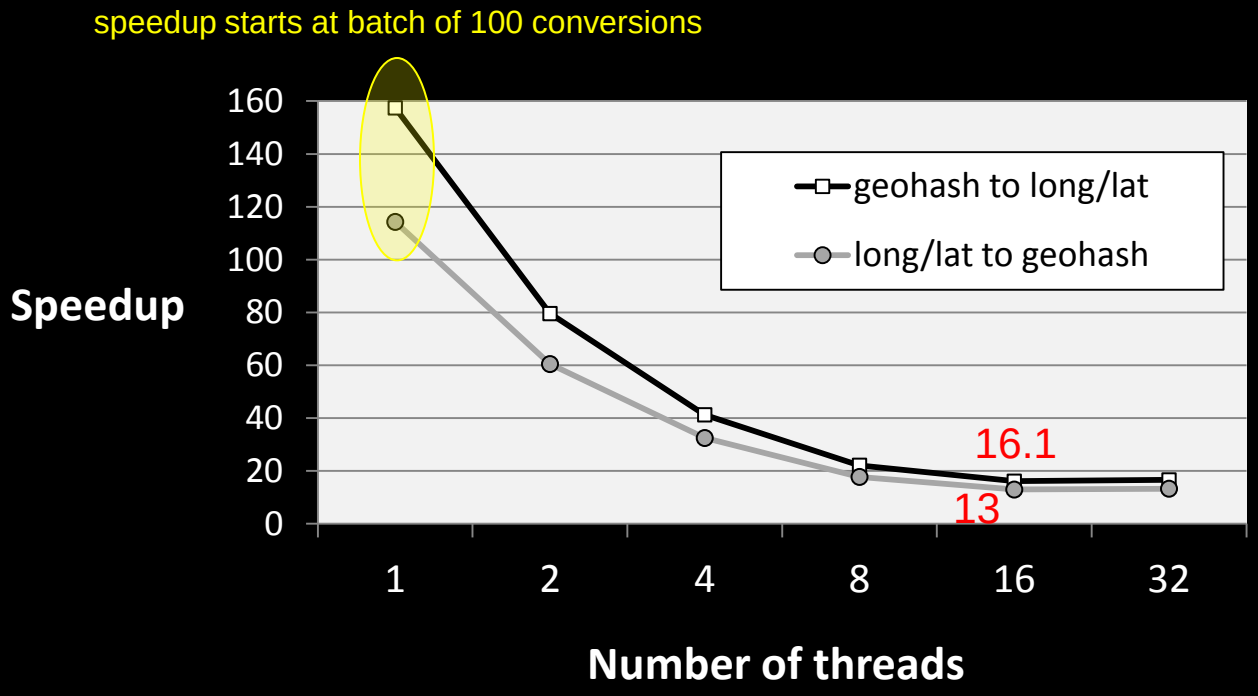
Speedup vs multi-threaded software: a socket-to-socket comparison

- 8-core x2 Hyper-Threads Intel Xeon @2.5GHz.
- 64-bit geohashes.
- 100M conversions.



Speedup vs multi-threaded software: a socket-to-socket comparison

- 8-core x2 Hyper-Threads Intel Xeon @2.5GHz.
- 64-bit geohashes.
- 100M conversions.



Conclusions

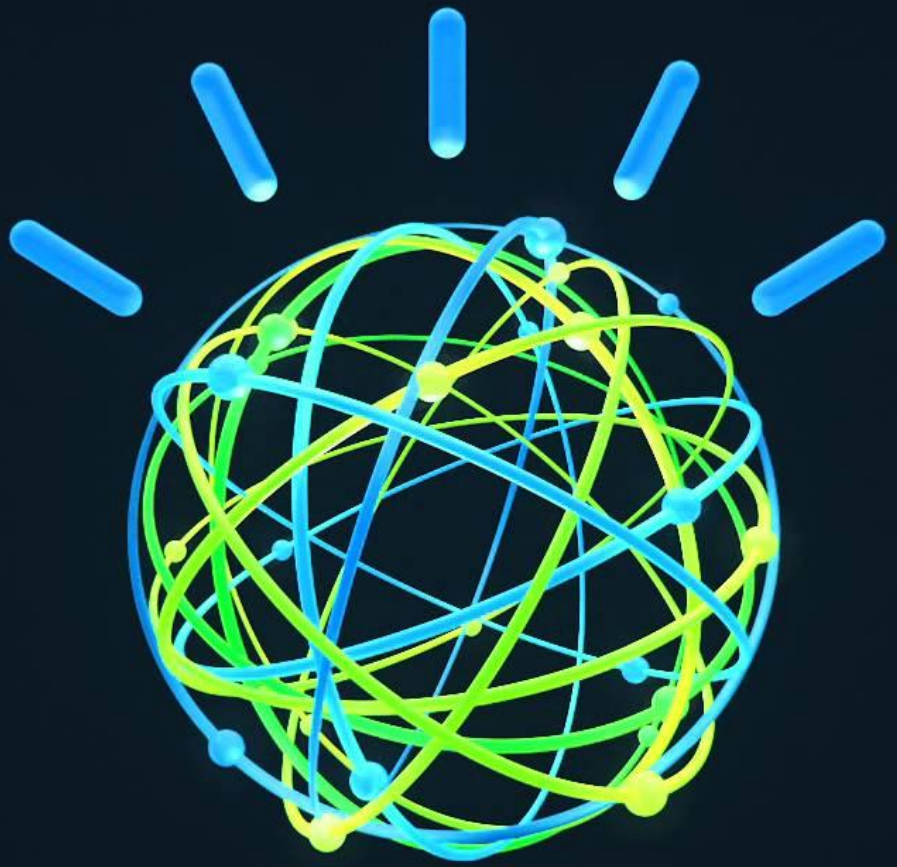
- First hardware implementation of a geohash conversion engine operating at wire speed.

- Flexibility:
  - Design time.
  - Run time.

- Performance:
  - >13X vs multi-threaded software.
  - Limited by the available PCIe bandwidth.

- Future work:
  - Accelerating compute-intensive queries in the geohash domain.

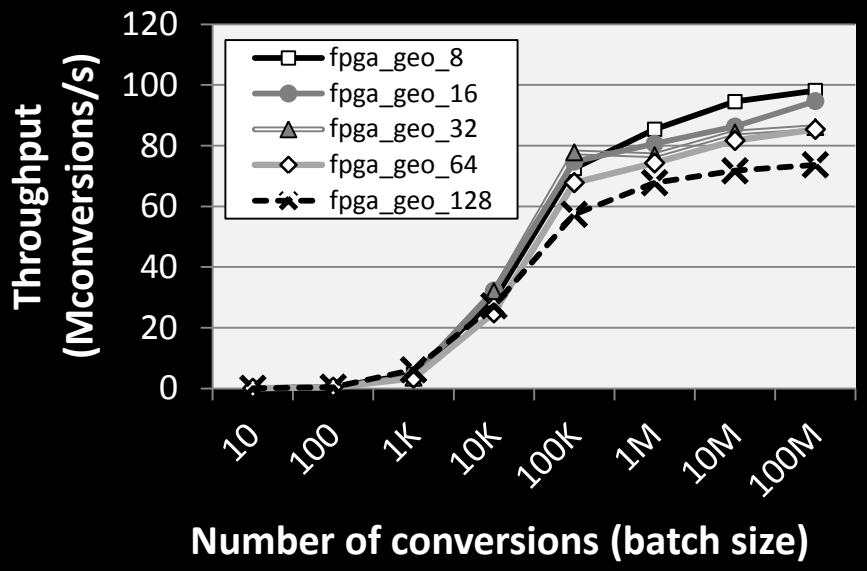
THANK YOU!



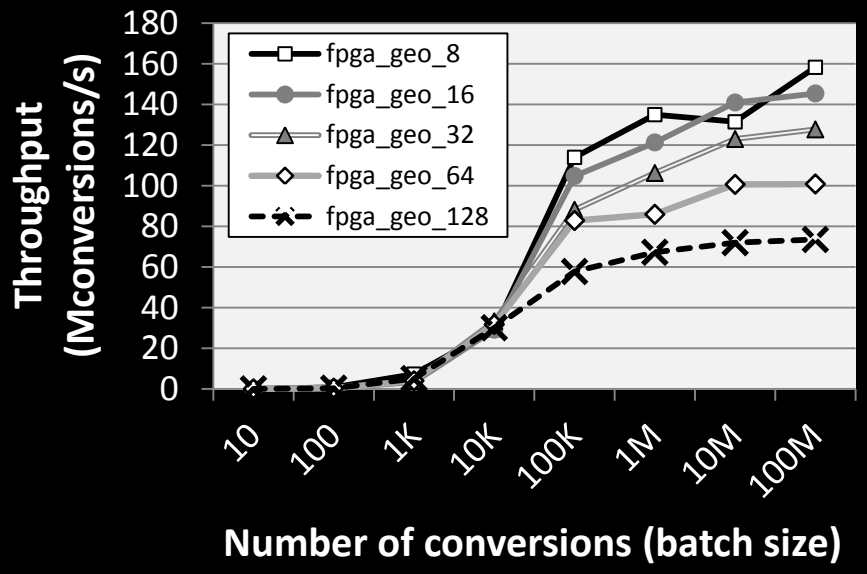
BACKUP

End-to-end throughput

geohash-to-lat./long.

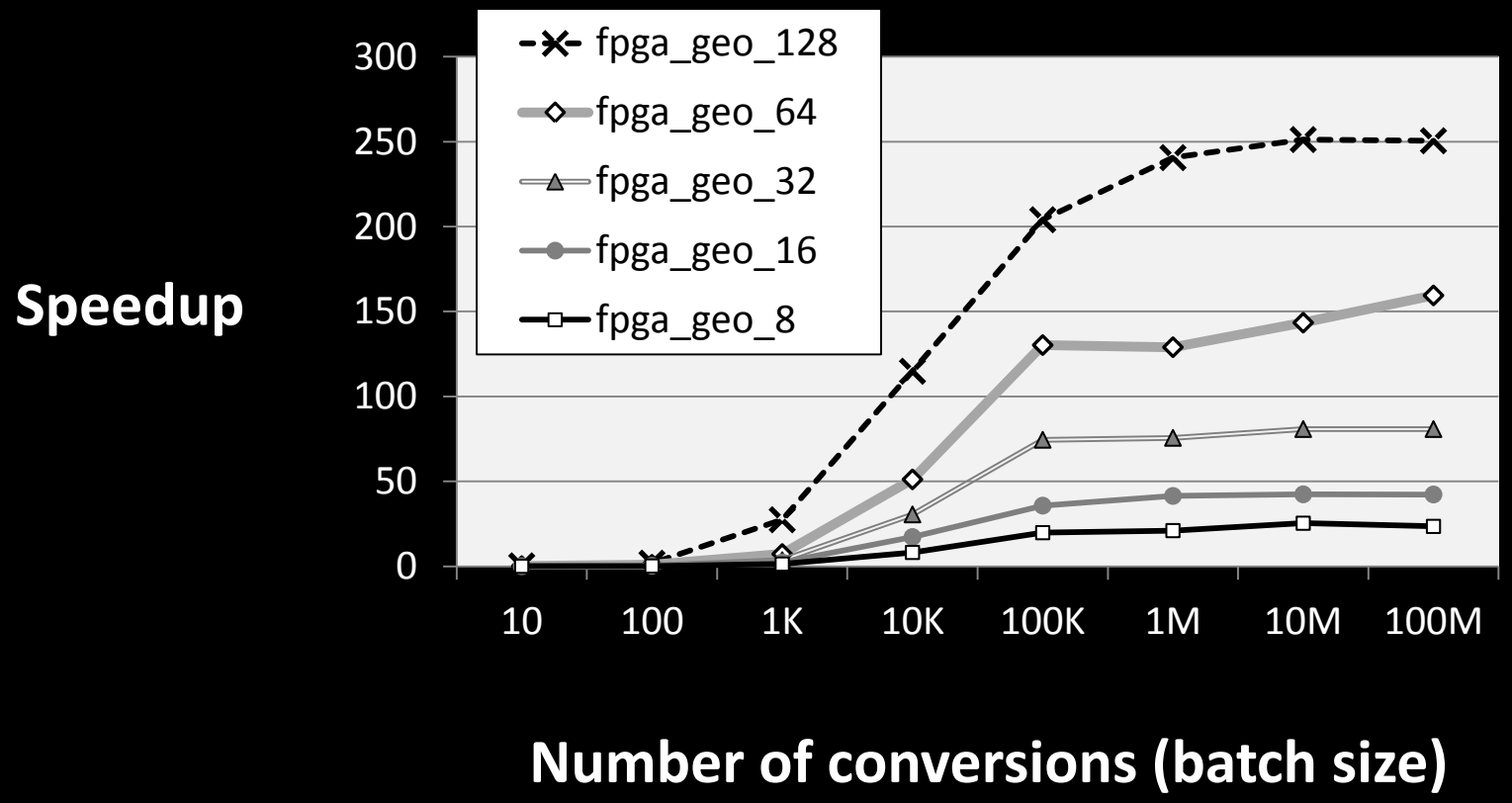


lat./long.-to-geohash



Speedup vs single-threaded software

- Geohash-to-lat./long.



Speedup vs single-threaded software

- Geohash-to-lat./long.