Addressing the Variety Challenge to Ease the Systemization of Machine Learning in Earth Science

K-S Kuo[1,2,3], M L Rilee[1,4], A O Oloso[1,5],

K Doan[1,3], T L Clune[1], and H-F Yu[6]

[1]NASA Goddard Space Flight Center
[2]Bayesics LLC
[3]ESSIC, University of Maryland-College Park
[4]Rilee Systems Technologies, LLC
[5]SSAI, Inc.
[6]University of Nebraska-Lincoln

The Vision

Data-Intensive Analysis System

- Meant for the great majority of Earth scientists
  - Not a specialized one for a particular type of data, a particular kind of analyses, or a particular subdomain/sub-community
- Generic Big Data analysis system optimized for the most prevalent class(es) of analyses
  - But still does well with other ones
- Good scalability in *Variety* to facilitate system science data analysis
  - A system science requires the collaboration of subdomains/sub-communities
  - Thus, integrative analysis involving varieties of data should be the norm
- Apt to streamline data preparation and facilitate automated machine learning

It is a Big Data problem!

- All the "*V*" challenges of Big Data apply:
  - Volume, Variety, Velocity, Veracity, …
- The ultimate determinant is ***Value***.
  - The approach offering the best value is the logical choice.
- The crucial capability is good scalability in ***Variety***.
  - Scaling ***Volume*** is relatively easy: Parallelization!

Current Practice

- A 2-step practice
  - Package data granules in “standardized” files.
  - Catalogue metadata for discovery and distribution.
- Perils of the practice – Needless waste in its wake
  - Copious data movements
  - Duplicated compute/storage resources and efforts
  - Inevitable barriers to collaboration
  - Compromised reproducibility
  - Impossible to systemize for machine learning

Technology Choice: SciDB

(To get away from the 2-step practice…)

- Full ACID DBMS that stores data in multidimensional arrays with strongly typed attributes (aka fields) within each cell.
  - Search and work directly on data, not just metadata
- Integrated advanced analytic suite for array analysis.
  - Extensible (but not trivial).
- Open-source community version
  - backed commercially by Paradigm4 who offers a licensed Enterprise version.
- Core of its data storage management is Paradigm4's Multidimensional Array Clustering (MAC™):
  - Data close to each other in the user-defined coordinate system are stored in the same chunk in the storage media.
  - Data are stored in the storage media in the same order as in the coordinate system

Considerations for Optimizing *Value*

- Simply using an Array DBMS does not necessarily yield best *Value*!
- In the following we describe some issues and solutions.

Data Placement Primer

- Simplistic data placement (or layout)
- For better load balance

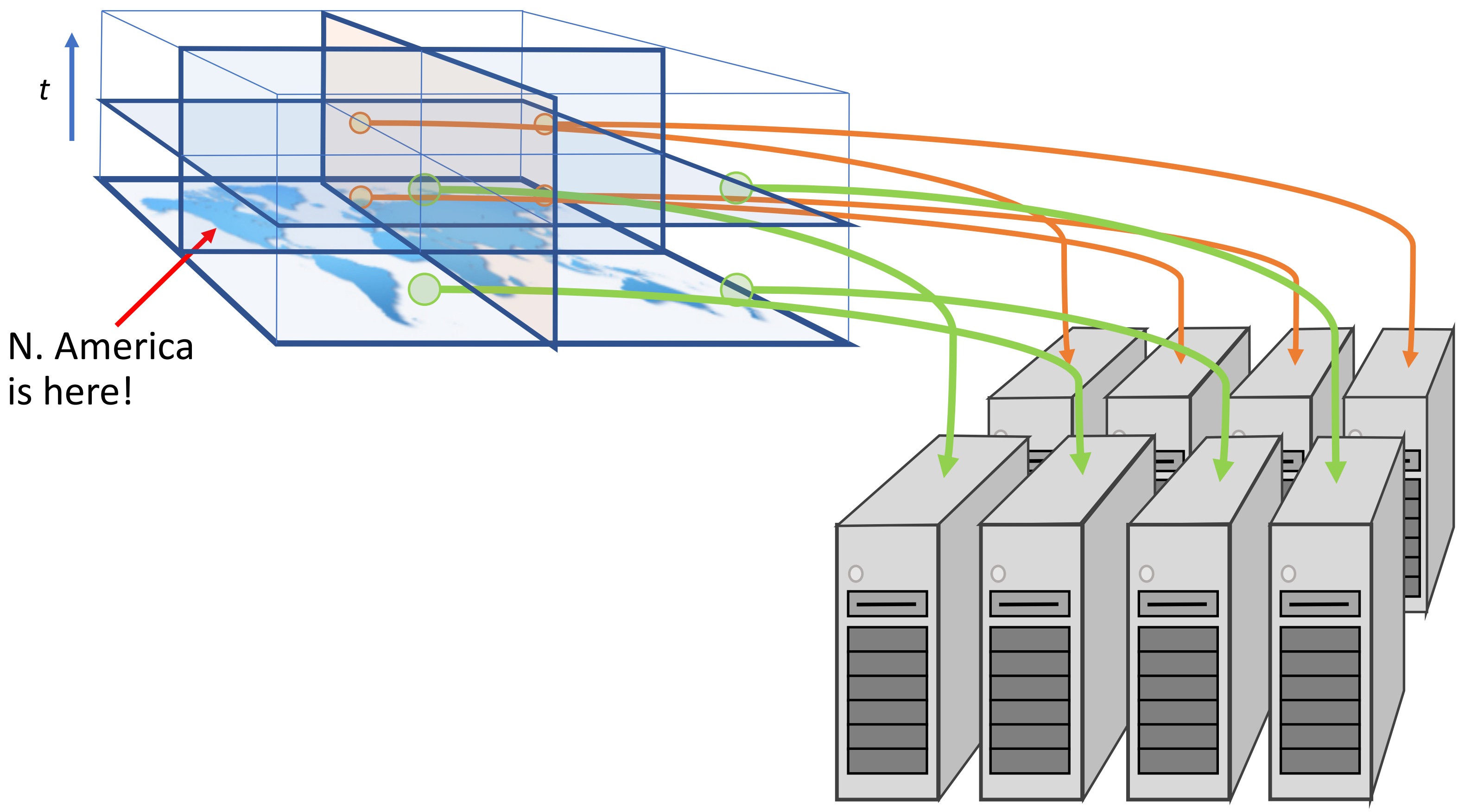
Importance of Data Placement

The coupling of the *analysis framework* and the *storage system*

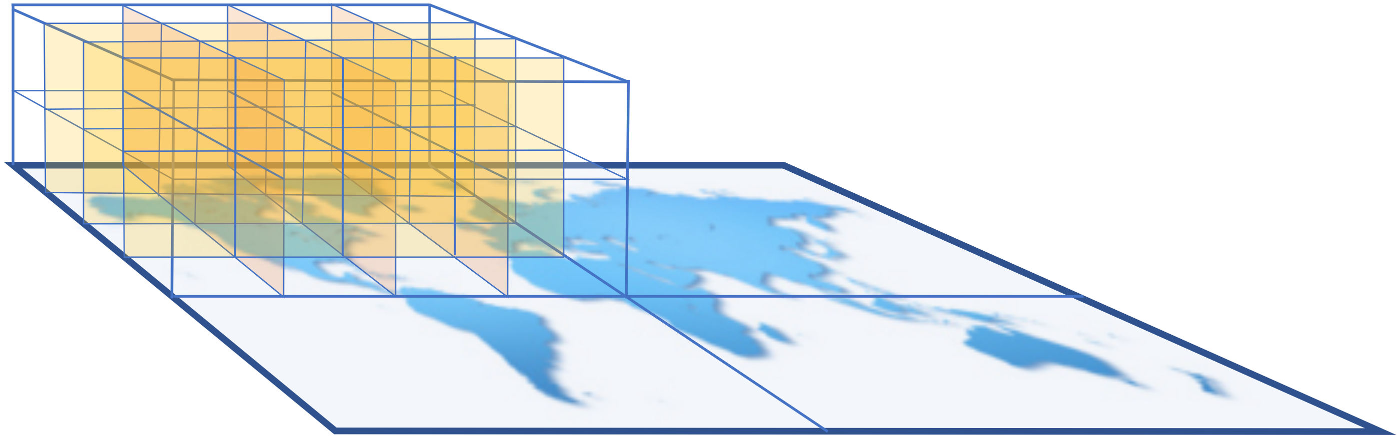
- Data placement is especially important for technologies in which the two are tightly coupled, such as DBMSs.
- MapReduce or Spark are analysis frameworks, which are only loosely coupled with the storage system (e.g. HDFS, Cassandra, …)
  - Loosely coupled technologies provide more flexibility or *elasticity* but less performance or *efficiency* (with equivalent resources).

Simplistic Data Placement (Layout)

– For *Variety* = 1



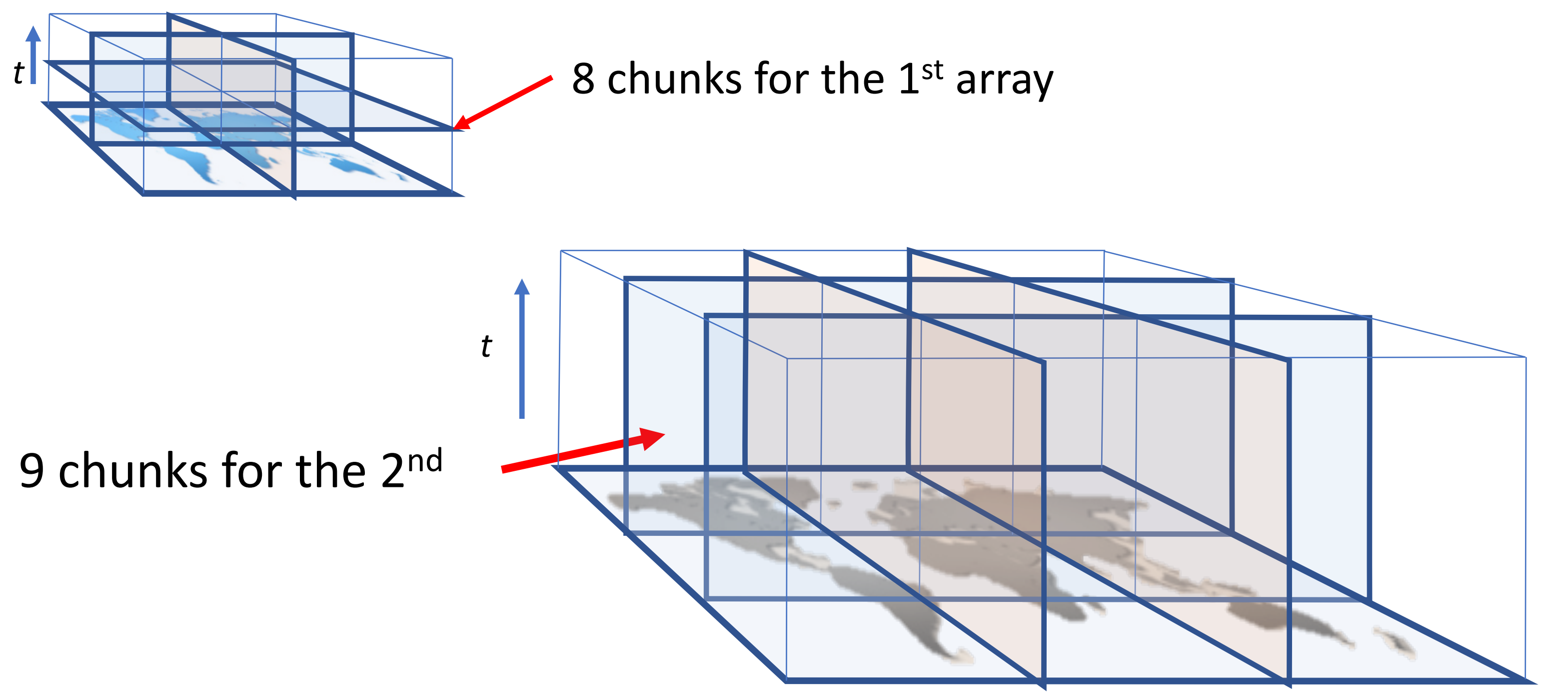
Better Load Balance
– Smaller Chunks



Data Placement (Mis-)Alignment

- Arrays of the same *shape** but partitioned differently
  - Arrays with the same dimensionality and dimensions (i.e. index ranges in each dimension) are said to have the same ”shape”!
- Increasingly more challenging cases (*Variety* > 1)

Arrays of the Same *Shape* but Different Partitions



Their chunks (or partitions) will not be aligned when they are placed on the cluster nodes!!

Consequences of Misalignment

- When misaligned arrays need to be processed together for *Integrative Analysis*, **they need to be aligned first!**
  - Expensive on-the-fly *repartitioning* becomes necessary.
    - Array elements of the same spatiotemporal subdomain must be moved to the same nodes.
    - Loosely coupled analysis-storage must (always) do on-the-fly repartitioning!
- Common **Integrative Analyses** include:
  - *Conditional subsetting*: Subset an array based on the filtered result of another array.
    - E.g. find in MERRA hourly array(s) where GPM DPR indicates presences of precipitation.
  - *Comparing between arrays*: Compare the same geophysical quantities between 2 arrays.
    - E.g. compare the precipitation rates between MERRA and GPM DPR
  - (And more…)
- Short paper presented at IEEE Big Data 2016 conference:
  - Doan, K., A. Oloso, K.-S. Kuo, T. Clune, H. Yu, B. Nelson, J. Zhang*: Evaluating of the Impact of Data Placement to Spark and SciDB with an Earth Science Use Case.*

Data Placement Alignment Matters!

- For technologies with tightly coupled analysis and storage, e.g. DBMSs, data placement is extremely important!
- SciDB out performs <u>Spark+HDFS</u> and <u>Spark+Cassandra</u> by a factor of **~3-10** for equivalent *integrative analyses*.

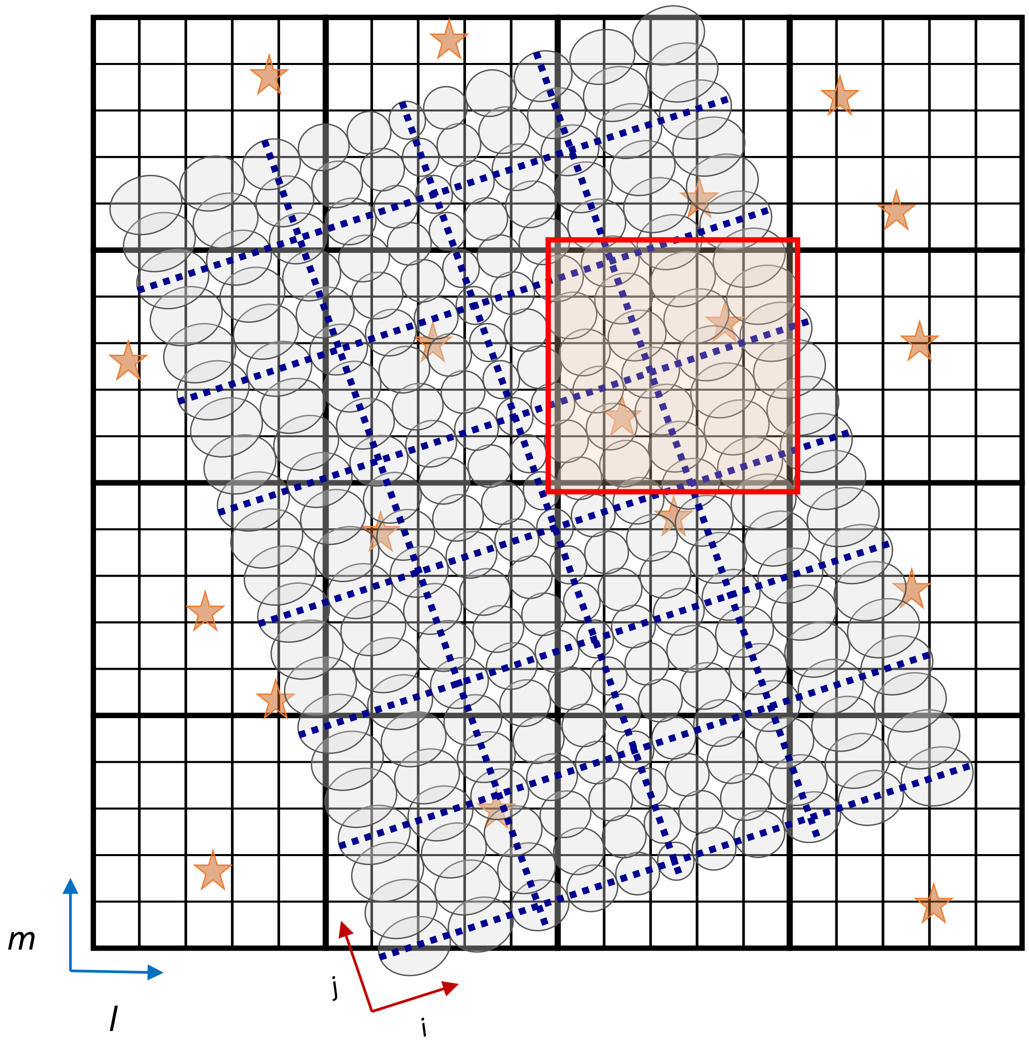
Increasingly More Challenging Cases

- Challenging!
  - It becomes challenging to align *arrays of different shapes*, i.e. different dimensionality and/or index ranges.
    - For example, both MERRA and MERRA-2 are regularly gridded arrays in longitude-latitude, but with slightly different resolutions, resulting in arrays of different shapes.
- Even more challenging!!
  - *Arrays of different data models* are even more challenging, even when they have the same shape!
    - Some subsets of NMQ/MRMS and GPM DPR products may have the same shape but because they have different geolocation representations, (NMQ/MRMS in Grid whereas GPM DPR in Swath). Processing efficiency for Integrative analysis of the two is difficult, even when the arrays are of the same shape!

Optimal Partitioning

- **Insight**: Since most of our integrative analyses require spatial and/or temporal coincidences, data should be partitioned so that data for the same *spatiotemporal subspace* reside on the same node!
    - A demonstration is given on the next few slides.
- **Grand Question**: With the diversity of Earth Science data, is there a generalized indexing scheme that can make the optimal partitioning possible, even simple and systematic?

Common Earth Science Data Models



Grid

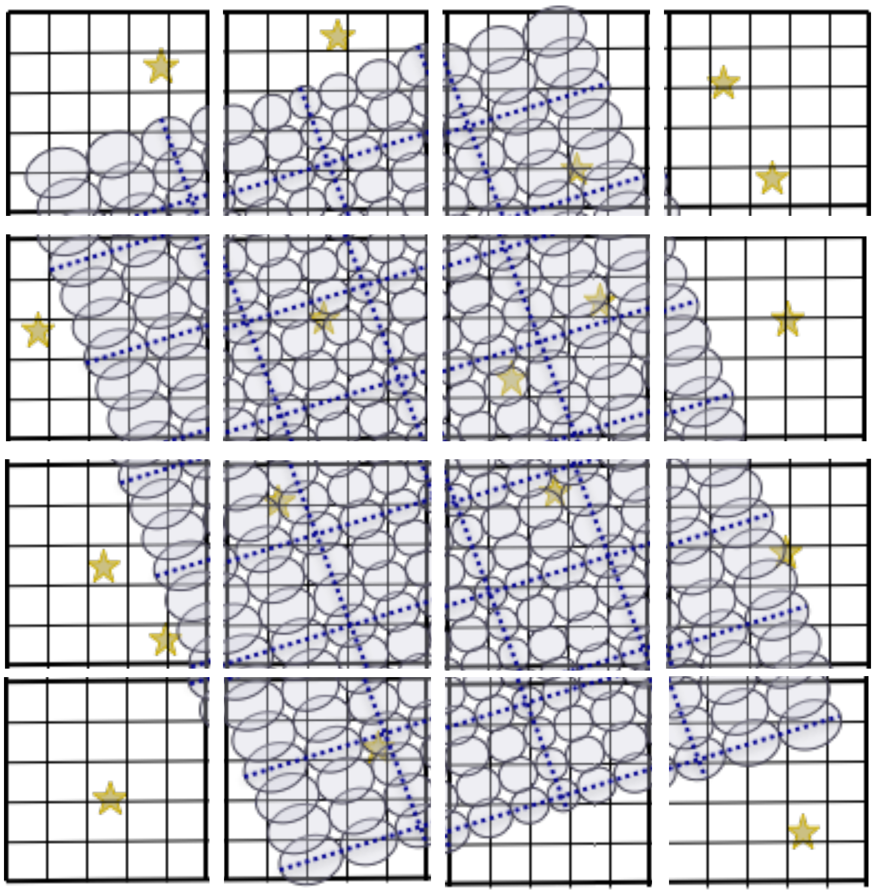
Swath

Point

Observation Station

Sat. Instrument IFOV

Aligning Different Models



As one can see, it would be tedious to partition datasets in the case-by-case manner as illustrated on the left.

How can we generalize the partitioning, so it may be simplified and automated to guarantee placement alignment?

HTM as a Solution to Alignment Challenges

Hierarchical Triangular Mesh (HTM) provides a good solution to the all the data placement challenges.

Introduction

- HTM is primarily used as an *indexing scheme* to address any piece of real estate on the surface of Earth (to a predefined uncertainty).
  - It is not used as a grid like those of numerical models for our purposes.
  - It indexes the representative geolocations of the data values and does not imply the value(s) associated with an index represents the spherical triangle area of that index.
- Short paper presented at IEEE Big Data 2016 conference:
  - Rilee, M., K-S Kuo, TL Clune, AO Oloso: *Addressing the Big-Earth-Data Variety Challenge with the Hierarchical Triangular Mesh*.

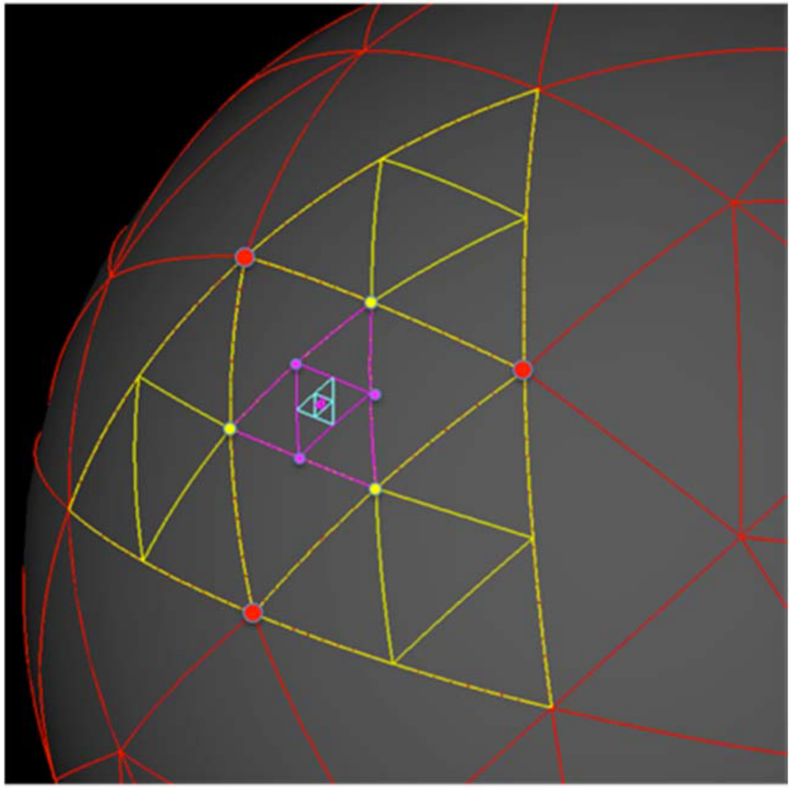
Addressing *V* in *Variety*

- Scaling volume is relatively easy – parallelization!
- High performance computing coupled with MPI has been successful in parallelizing numerical simulations – tightly coupled problems.
    - This is the reason HPC+MPI is among the first solutions considered for Big Data.
- HPC is expensive and, with the current 2-step practice, it does not scale variety well.
- Shared nothing architecture can leverage commodity compute resources – less expensive.
    - More suitable for loosely coupled problems.
- Need good strategy to homogenize the differences among different data models to realize the maximum value – **Scaling variety!!!**

Hierarchical Triangular Mesh – HTM

HTM is a way to address the surface of a sphere using a hierarchy of spherical triangles.

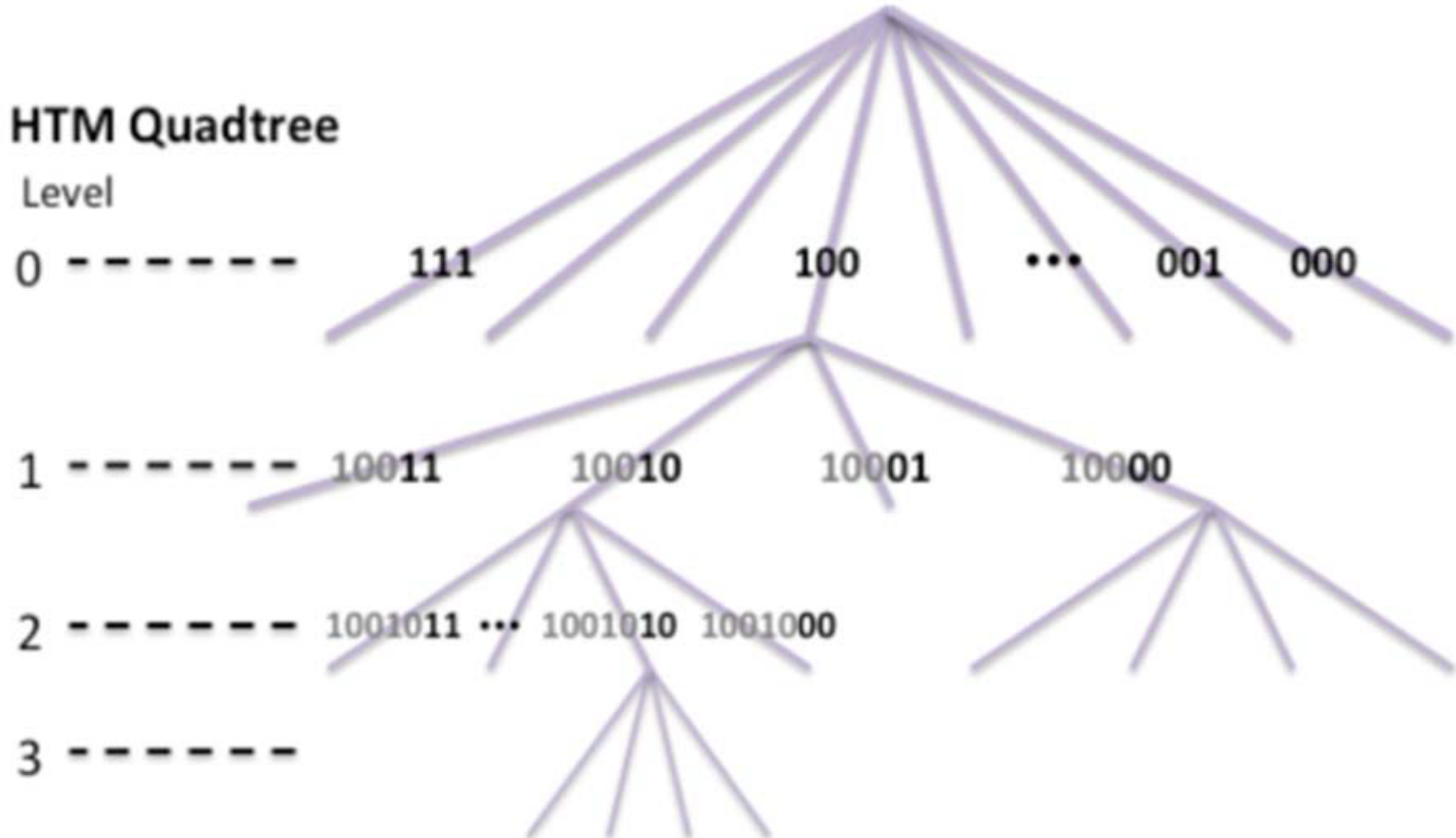
- Start with an inscribing octahedron of a sphere.
- Bisect each edge.
- Bring the bisecting points to inscribe the sphere to form 4 smaller spherical triangles.
- Repeat



Advantages of HTM

- Earth's surface is indexed with 1D indices and forms a **quad tree**.
    - Each piece of real estate, down to a chosen resolution, can be assigned an HTM index, HID.
- The resolution reaches $\lesssim$**1 m**, at the 24^{th} level (requiring 49 bits).
- When a level of the quad tree is chosen to be the chunk length for partitioning, all levels below it (same approximate geographical neighborhood) will reside on the same node.
- Thus, using the HTM index allows us to simultaneously
    - **Geo-reference different data representations in a uniform way,** and
    - **Ensure placements of data from diverse datasets may be aligned.**
- Specifying spherical regions is very efficient with HTM.
    - Every arc edge of a spherical triangle is a segment of a great circle.

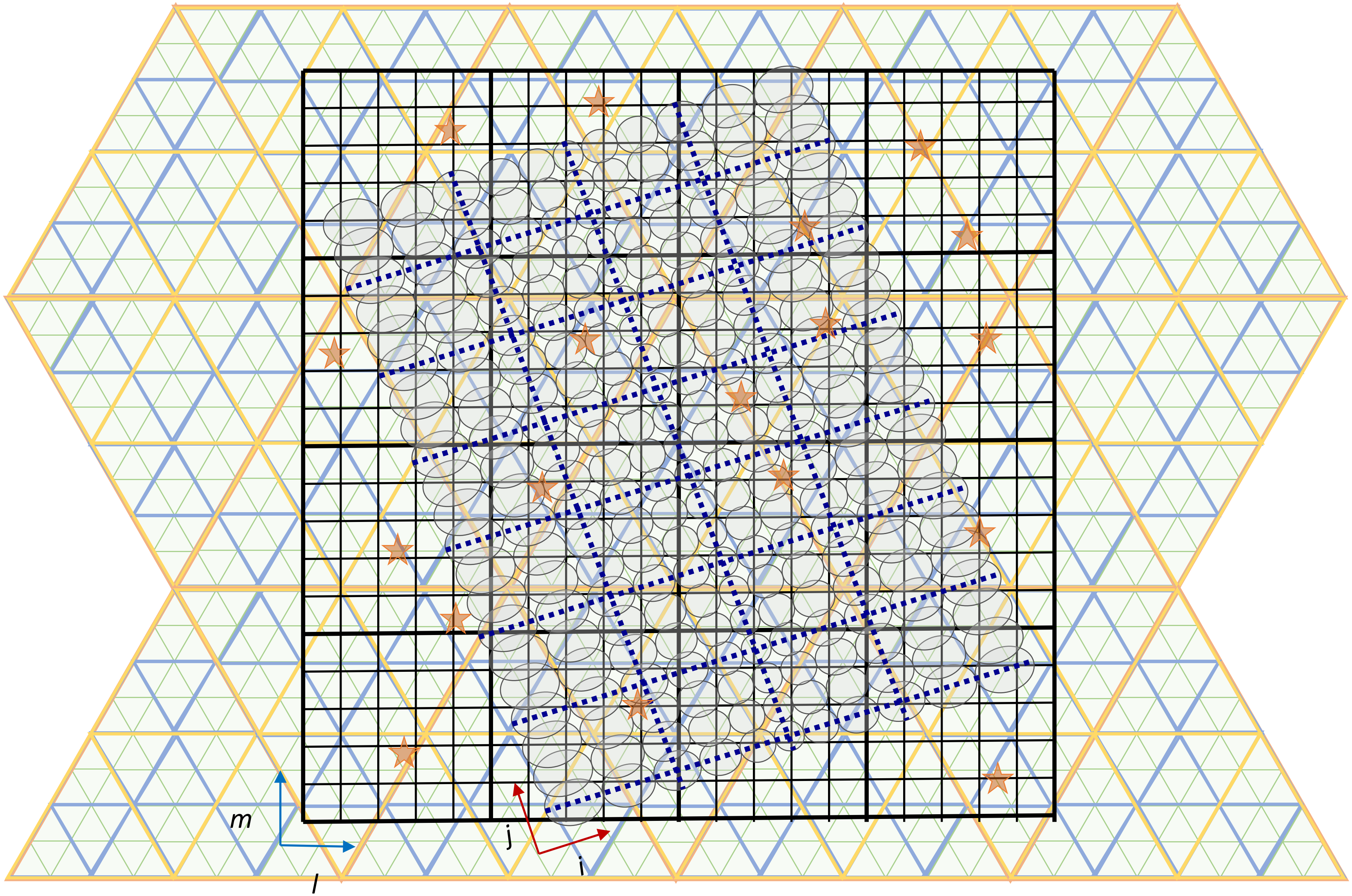
HTM Quad-tree



HTM Quad-tree

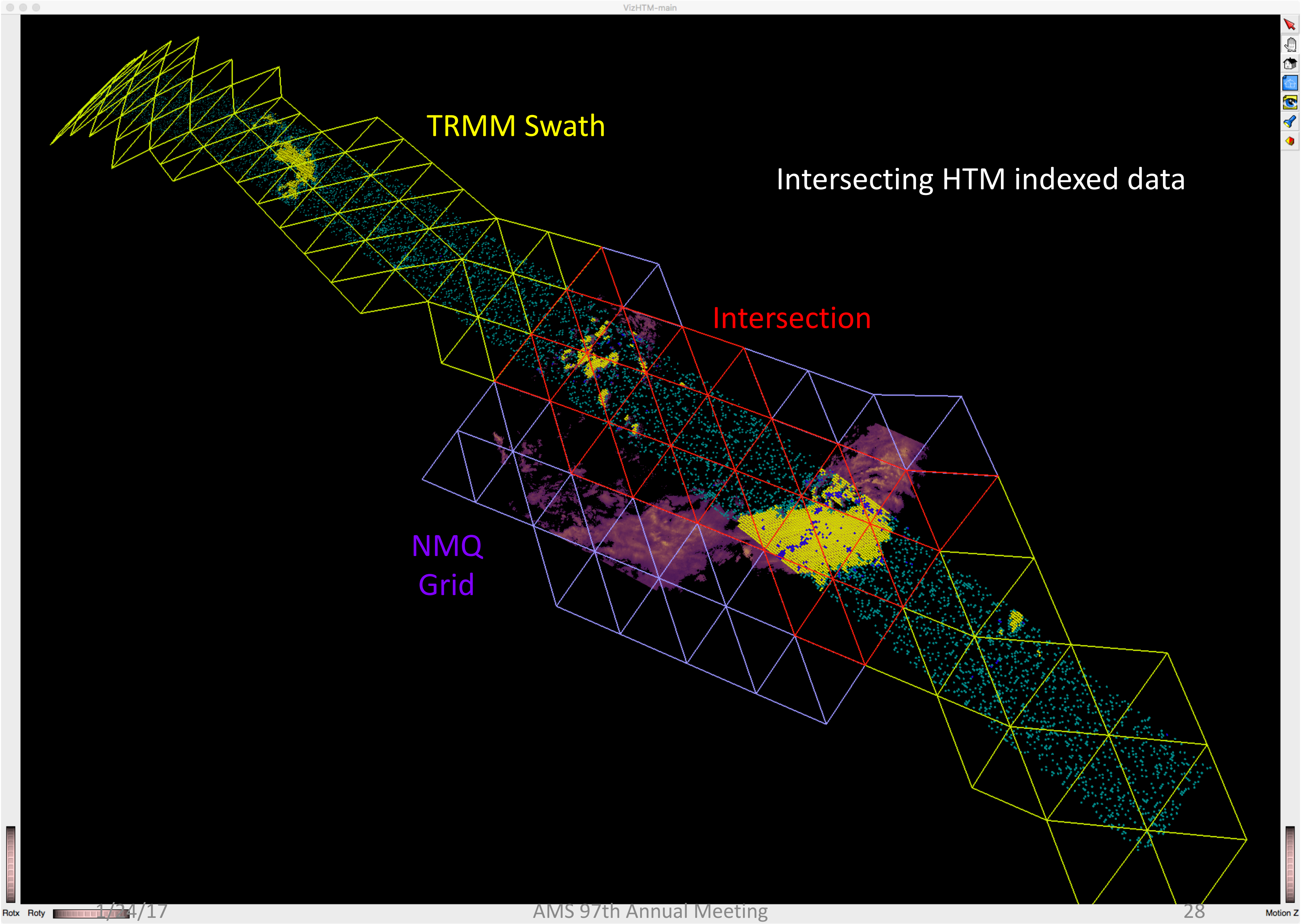
- 8 spherical triangles at level 0, i.e.
  - 000, 001, 010, 011, 100, 101, 110, 111
- After 3 quadfurcations, there are
  - 8x4 depth level 1 triangles
  - $8x4^2$ depth level 2 triangles
  - $8x4^3$ depth level 3 triangles
- Table on the right shows sub-triangle indices for the 4th one of the depth level 0 triangles.

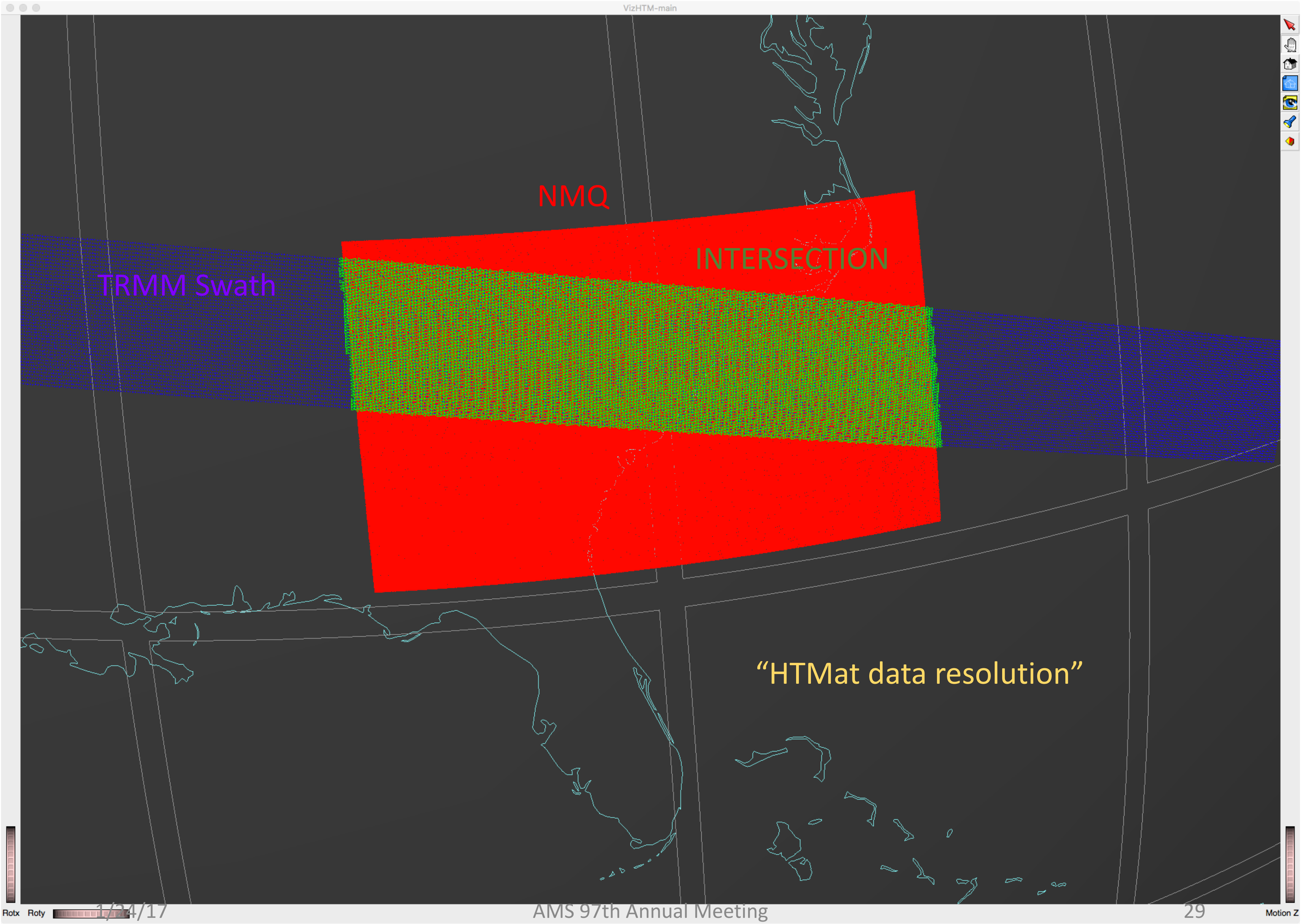
0	1	2	3	decimal
100	00	00	00, 01, 10, 11	0, 1, 2, 3
100	00	01	00, 01, 10, 11	4, 5, 6, 7
100	00	10	00, 01, 10, 11	8, 9, 10, 11
100	00	11	00, 01, 10, 11	12, 13, 14, 15
100	01	00	00, 01, 10, 11	16, 17, 18, 19
100	01	01	00, 01, 10, 11	20, 21, 22, 23
100	01	10	00, 01, 10, 11	24, 25, 26, 27
100	01	11	00, 01, 10, 11	28, 29, 30, 31
100	10	00	00, 01, 10, 11	32, 33, 34, 35
100	10	01	00, 01, 10, 11	36, 37, 38, 39
100	10	10	00, 01, 10, 11	40, 41, 42, 43
100	10	11	00, 01, 10, 11	44, 45, 46, 47
100	11	00	00, 01, 10, 11	48, 49, 50, 51
100	11	01	00, 01, 10, 11	52, 53, 54, 55
100	11	10	00, 01, 10, 11	56, 57, 58, 59
100	11	11	00, 01, 10, 11	60, 61, 62, 63


m
l
j
i

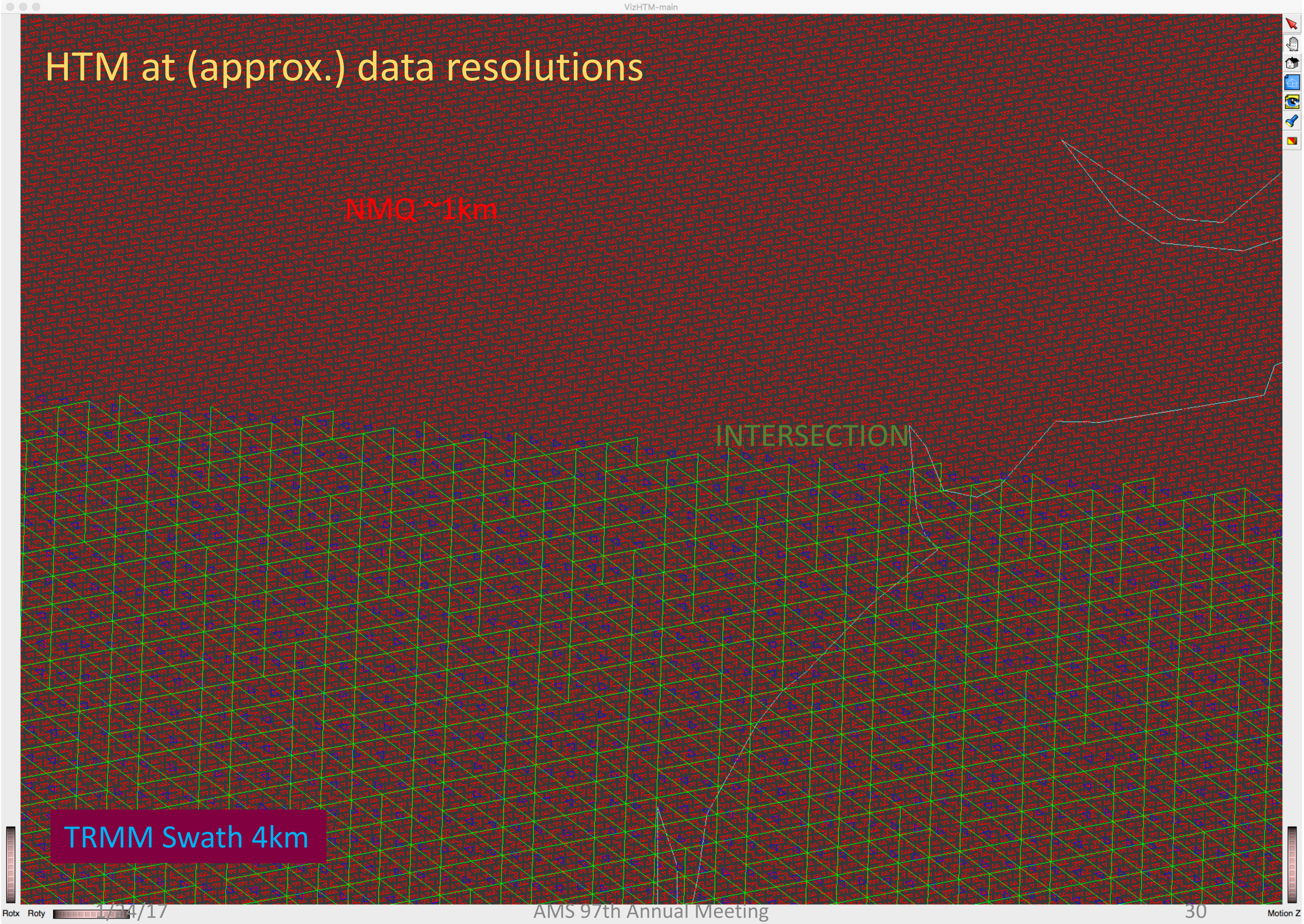
HTM Indexing

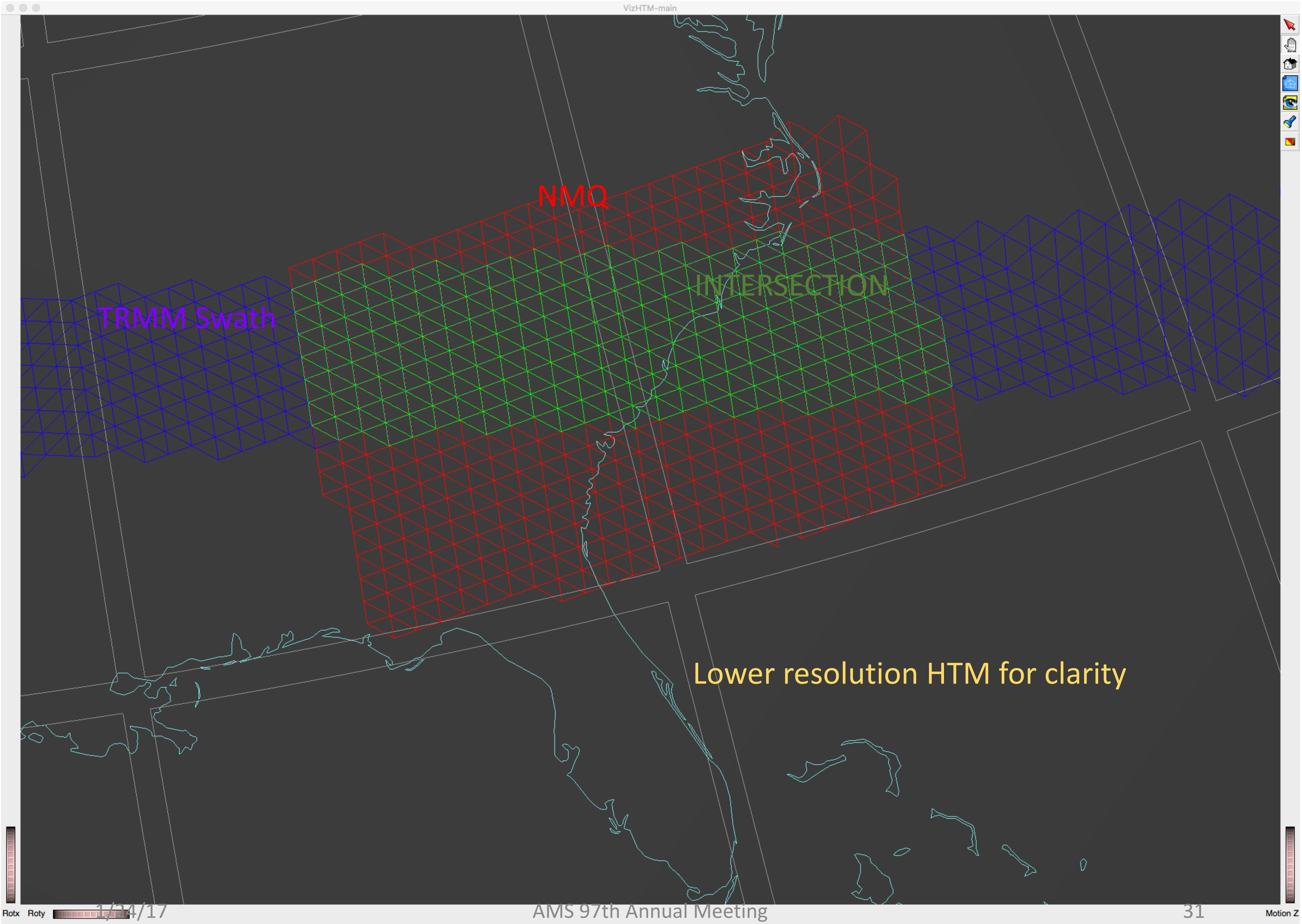
- An HTM index is assigned to
  - Every cell of Grid,
  - Every IFOV of Swath, and
  - Every location of Point.
- Every Earth Science dataset is indexed the same way.
  - Same index is the same place on Earth for all data arrays!
- This results in very sparse arrays!
- SciDB is efficiently with sparse arrays, fortunately!

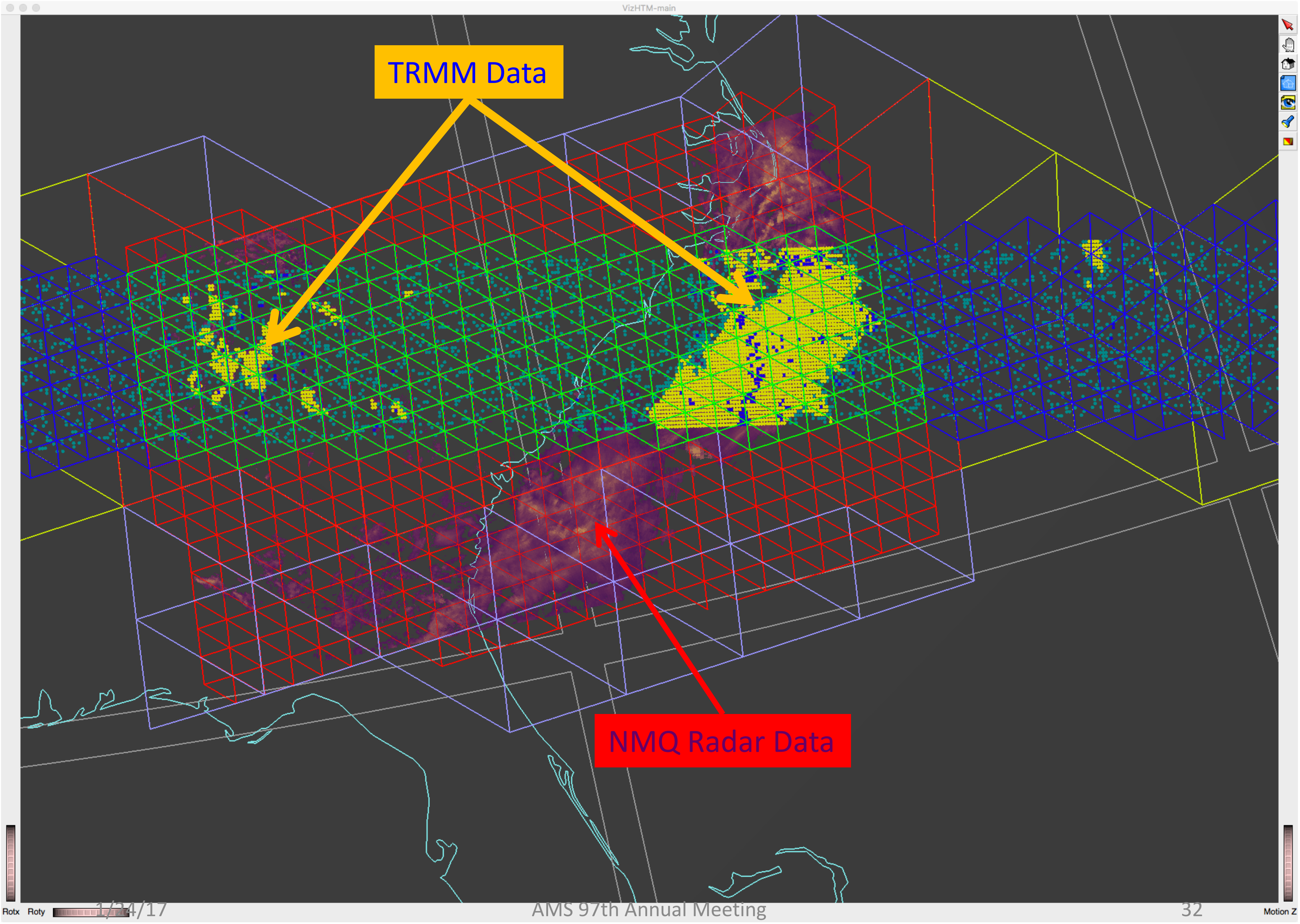

VizHTM-main
TRMM Swath
Intersecting HTM indexed data
Intersection
NMQ
Grid
Rotx
Roty
Motion Z


VizHTM-main
NMQ
INTERSECTION
TRMM Swath
“HTMat data resolution”
Rotx
Roty
Motion Z

HTM at (approx.) data resolutions

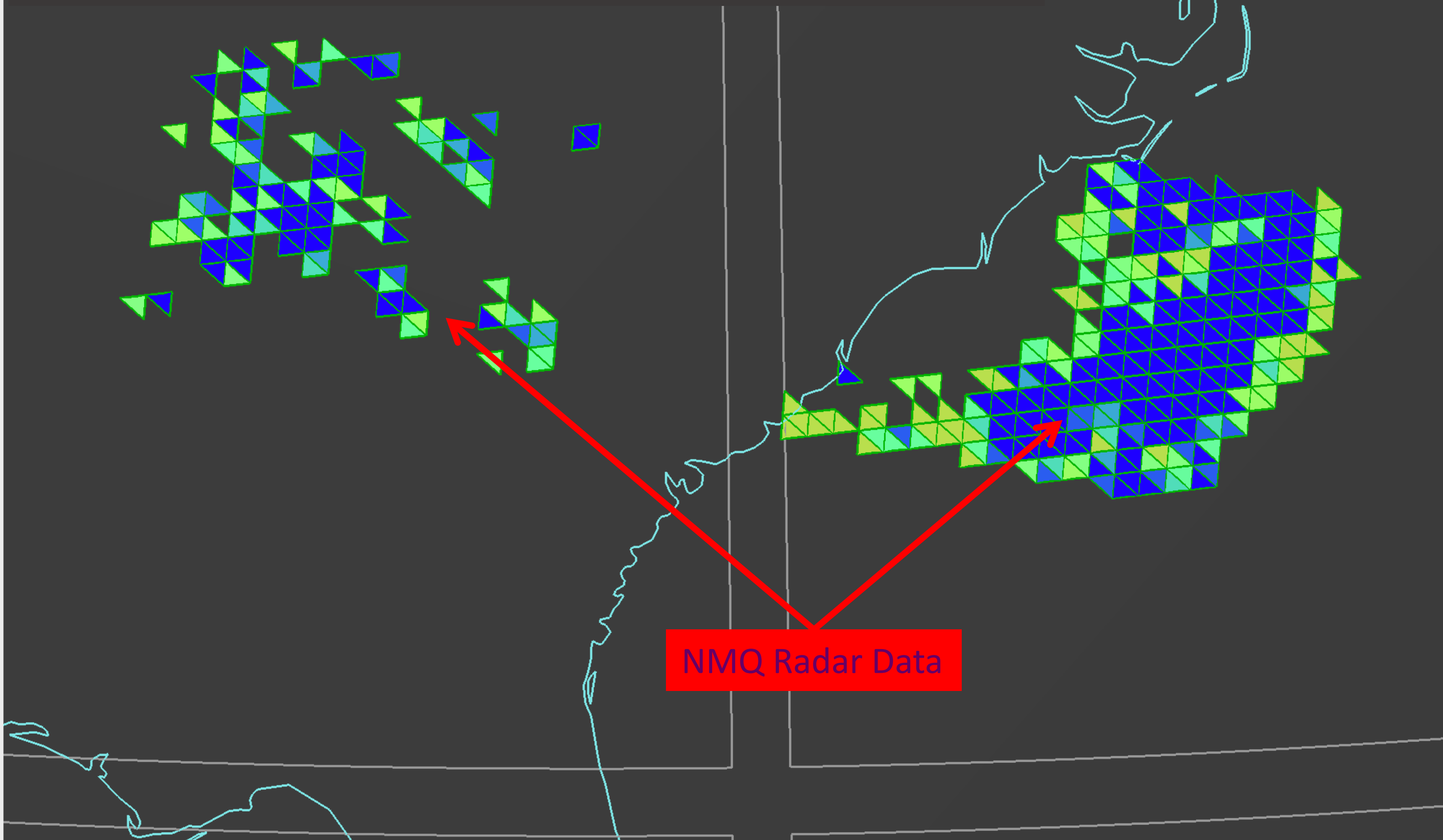



VizHTM-main
NMQ
INTERSECTION
TRMM Swath
Lower resolution HTM for clarity
Rotx
Roty
Motion Z


VizHTM-main
TRMM Data
NMQ Radar Data
Rotx
Roty
Motion Z

Integrative Analysis

- *Conditional subsetting* by a simple query
  - Choosing NMQ data based on TRMM via HTM intersection



Summary - HTM

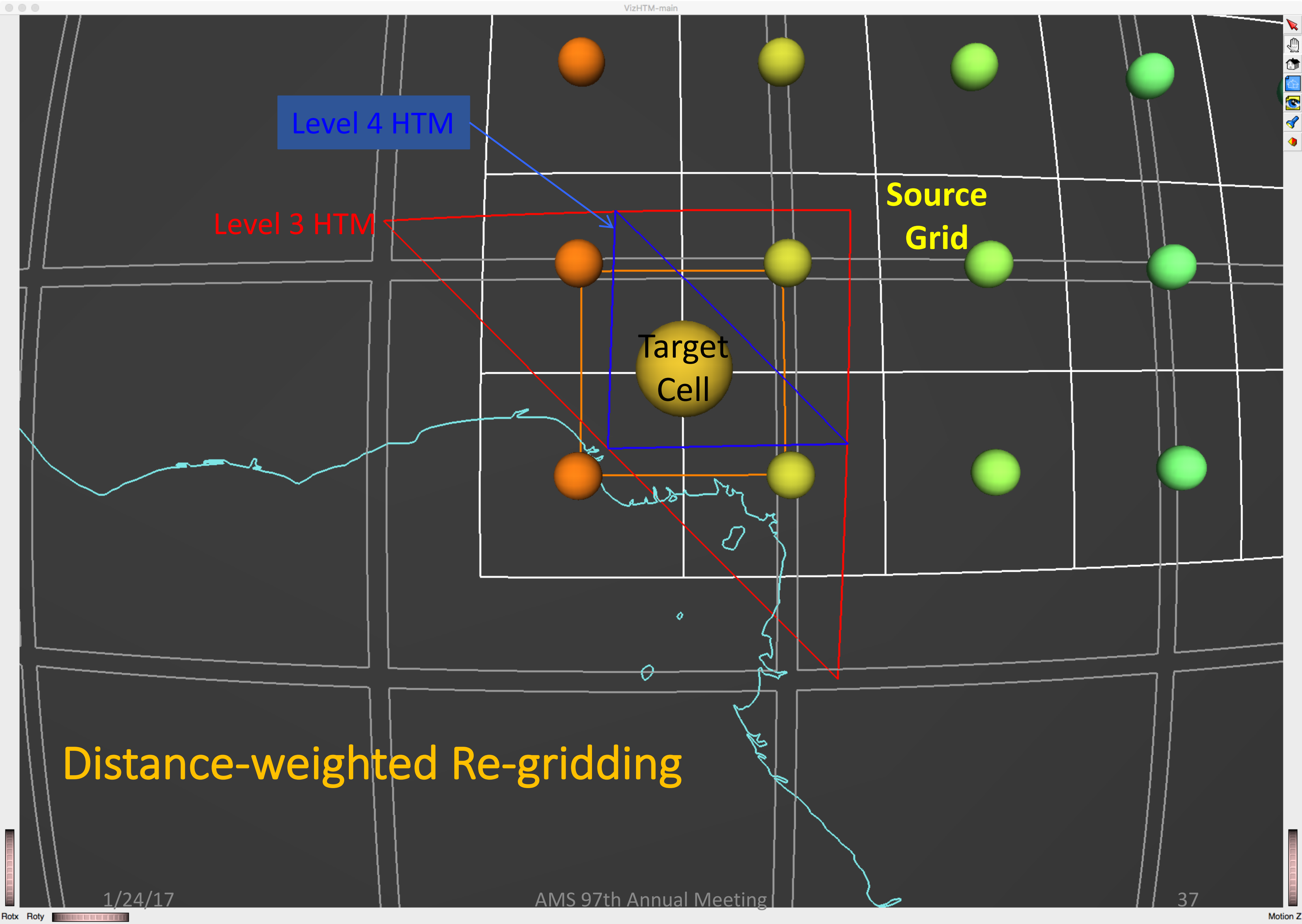
- Presents uniform representation for different data models
  - i.e. Grid, Swath, and Point
- Provides mechanism for efficient *conditional sub-setting*.
  - Identify subsets in one dataset based on criteria applied to another dataset.
- Supports data placement alignment for common analysis/queries.
- Facilitates efficient spatial set operations.
- **Serves as a key to scaling variety in Earth Science data analysis.**

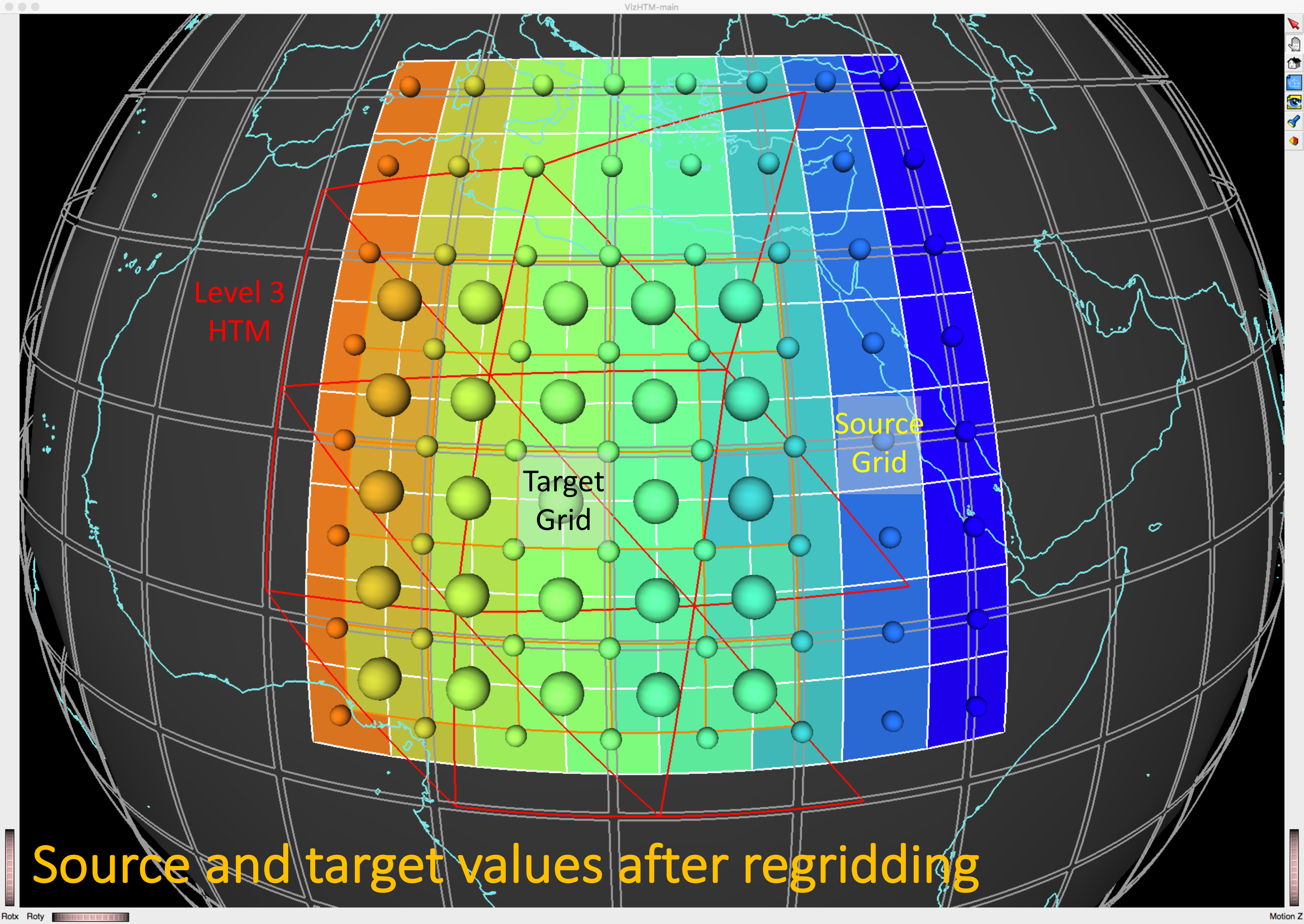
Re-gridding/Re-mapping

Systemizing a necessary, often duplicated, repeated process

Integrative Analysis

- *Integrative analysis* uses more than one dataset together for analysis.
- HTM allows us
  - to address all data arrays (sets) uniformly and
  - to perform set operations efficiently.
- We need re-gridding/remapping to *compare* data values of the same geophysical quantities but obtained by different means.
  - NMQ has a spatial resolution of 0.01°×0.01° (~1 km).
  - TRMM PR has a spatial resolution of ~4 km.
- Meaningful comparisons (or integrative analyses) require:
  - Re-grid *A* to *B* (i.e. re-grid NMQ to TRMM PR), or
  - Re-grid *B* to *A* (i.e. re-grid TRMM PR to NMQ), or
  - Re-grid *A* and *B* to *C* (i.e. re-grid both to a 3rd common geometry).


VizHTM-main
Level 4 HTM
Level 3 HTM
Source
Grid
Target
Cell
Distance-weighted Re-gridding
Rotx
Roty
Motion Z


VizHTM-main
Level 3
HTM
Source
Grid
Target
Grid
Source and target values after regridding
Rotx Roty
Motion Z

Conclusions

- Array-based DBMS, e.g. SciDB, provides better performance and value when data placements are aligned for the most prevalent class(es) of analyses.
- Hierarchical Triangular Mesh (HTM) provides a unified way to index most of (if not all) Earth Science data.
- Integrated re-gridding/remapping tool box homogenizes integrative data analysis of different varieties.
- Systemized machine learning becomes within reach.

Acknowledgement

We are grateful for funding provided, primarily, by **NASA** Earth Science Technology Office (**ESTO**) through its Advanced Information Systems Technology (**AIST**) program and by the National Science Foundation (**NSF**) through the **EarthCube** program. The excellent computing support provided by NASA High End Computing (**HEC**) program through NASA Center for Climate Simulations (**NCCS**) is also acknowledged.

Thank you!