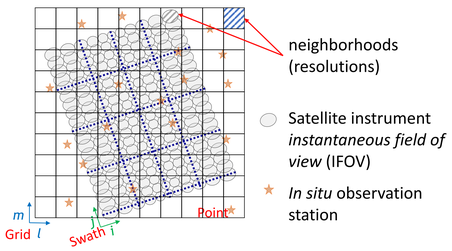
STARE for scalable unification of diverse data within Earth, Space, and Planetary Science

M L Rilee[1] K-S Kuo[2] J H R Gallagher[3] J Frew[4] N Griessbaum[4] K Neumiller[3] R E Wolfe[5] Hongfeng Yu[6] P E Clark[7]

[1]Rilee Systems Technologies, LLC, Derwood MD; [2]Bayesics LLC, Bowie, MD; [3]OPeNDAP, Inc., Narragansett, RI; [4]University of California, Santa Barbara; [5]NASA Goddard Space Flight Center, Greenbelt, MD; [6]University of Nebraska, Lincoln; [7]Jet Propulsion Laboratory, Pasadena, CA

The Problem

- Low-level datasets have immense value but are hard to use and integrate.
- High-level datasets reinterpret low-level data, e.g. to a common grid or resolution or to fill in gaps, easing integration, while washing out important details.
- Customized integration aligning diverse data (swath, grid, point) and exploiting high resolution data does not scale.

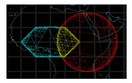


- Native array indexing disconnects data from its spatiotemporal arrangement impeding integration and coalignment.
- We need an index strategy supporting alignment of data in space and time and on distributed/parallel computing resources.

STARE provides an alternative to native-array indexing and longitude/latitude geolocation by combining spatiotemporal location and neighborhood semantics, opening diverse low-level datasets to customized integrative analysis using distributed/parallel computing or Cloud.

STARE-based Tooling

- C/C++ STARE Library
- PySTARE
- budding GeoPandas/Shapely support



Intersecting with Modis data

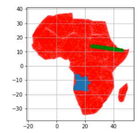
```
In [15]: # Get the granule at:
         # https://ladsweb.modaps.eosdis.nasa.gov/archive/allData/6/MOD09/2019/317/MOD09.A2019317.081
         file_path = 'MOD09.A2019317.0815.006.2019319020759.hdf'
         hdf = SD(file_path)
         lat = hdf.select('Latitude').get().flatten()[0:1354]
         lon = hdf.select('Longitude').get().flatten()[0:1354]
         band4 = hdf.select('1km Surface Reflectance Band 4').get().flatten()[0:1354]
         modis = pandas.DataFrame({'lat': lat, 'lon': lon, 'band4': band4})
         modis = geopandas.GeoDataFrame(modis, geometry=geopandas.points_from_xy(modis.lon, modis.lat
         modis['stare'] = pystare.from_geopandas(modis, 12)
         modis.head()
```

Out[15]:

	lat	lon	band4	geometry	stare
0	13.981338	24.933973	1848	POINT (24.93397331237793 13.981337547302025)	4528191884642127404
1	13.976474	24.978500	1799	POINT (24.97850036621094 13.976473808288857)	4528191929150335436
2	13.971599	25.023054	1943	POINT (25.023054122924805 13.971598625183105)	4528192141154377164
3	13.966847	25.066420	1925	POINT (25.066419601440643 13.966847419738770)	4528191461944221900
4	13.961843	25.111959	1561	POINT (25.111959457397461 13.961843490600586)	4528191340451727692

```
In [16]: fig, ax = plt.subplots()
         ax.grid(True)

         africa.set_geometry('trixels').plot(color='red', ax=ax, linewidth=10)
         africa.set_index('name').loc[['Angola']].plot(ax=ax)
         modis.plot(ax=ax, color='green')
Out[16]: <matplotlib.axes._subplots.AxesSubplot at 0x7fa1a0b84358>
```



Disaster Recovery in Africa (excerpt)
A STARE-based Jupyter Notebook showing STARE/HTM trixels for nations (red), Angola (blue), and a Tropical Rain Mapping Mission (TRMM) granule (green). UCSB.

STARE Use Cases Under Study

- OPeNDAP
- UCSB - Multi-sensor snow cover MODIS/VIIRS/LANDSAT/GOES
- LAADSWEB/MODAP – MODIS, VIIRS
- PMM IMERG – Microwave radiometry
- Tropical land surface – MODIS/VIIRS & LANDSAT TM colocation
- ICESAT/MODIS – Colocation over poles
- CLARREO – Cross calibration

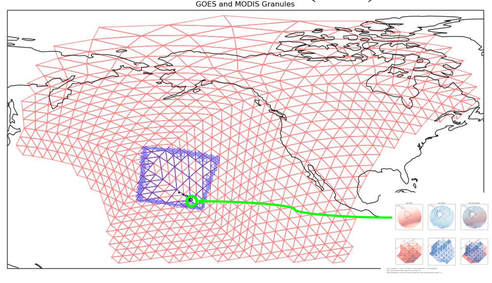
STARE

- **S**patio**T**emporal **A**daptive-**R**esolution **E**ncoding – A hierarchical geo-spatiotemporal reference using **integers**.
- STARE has two elements: The geo-spatial *Hierarchical Triangular Mesh* (HTM) and the temporal *Hierarchical Calendrical Partition* (HCP).
- HTM is a way to index the surface (actually, solid angle) of a sphere using a hierarchy of spherical triangles.
  1. Start with an inscribing octahedron of a sphere.
  2. Bisect each edge.
  3. Project the bisecting points from sphere center to the sphere surface forming 4 smaller spherical triangles.
  4. Repeat quadfurcation from 2.
- The index at the 27th quadfurcation level (Q-level) is used for indexing geolocation with better than 10 cm precision.
- *Neighborhood size* is given by Q-level, e.g Q-10~10km.
- HTM encodes both *location* and *neighborhood* (trixel).
- HCP encoding is similar to HTM, but branching at each "neighborhood" level depends on calendrical subdivision.
- Temporal indexing is built on international standards TAI and UTC using the International Astronomical Union's *Standards of Fundamental Astronomy* (via *ERFA).*

STARE Unified Indexing

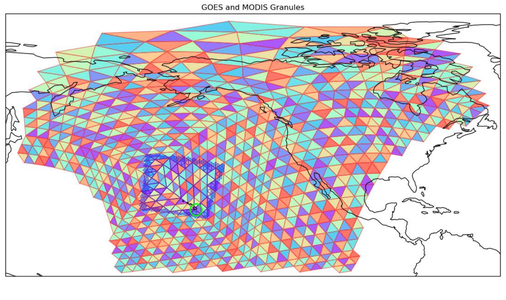
GOES footprint (red)

MODIS Granule (blue)

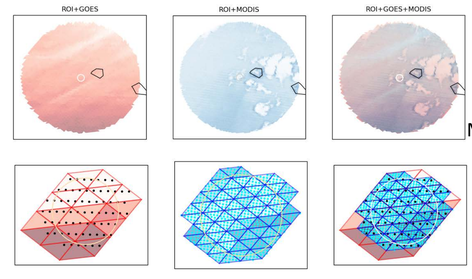


Distributed Coalignment

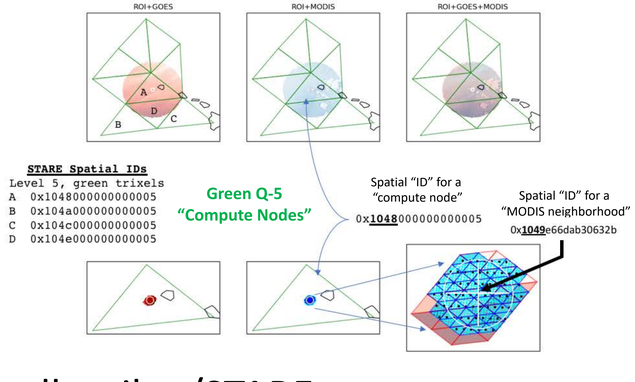
Distribute across 16 "nodes" or tasks.



Easy Low-Level Data Integration



Spatial and Cyber Coalignment



STARE Future/Take-away

- STARE is maturing, becoming usable for combining diverse data by early adopters.
- A growing variety of functions in convenient APIs are being developed.
- STARE provides a foundation for searching, processing, and packaging data on distributed/parallel systems and Cloud.
- Improvements including radial indexing, alternative root polyhedron, and non-spherical bodies are in study.

https://github.com/michaelleerilee/STARE

Acknowledgement.
NASA ACCESS program provided partial support to this effort, award ID 80NSSC18M0118.