Enhancing SAMOS Data Access in DOMS via a Neo4j Property Graph Database IN21C-1745

Shawn R. Smith, **Adam Stallard**, Jocelyn Elya
Center for Ocean-Atmospheric Prediction Studies, Florida State University, Tallahassee, FL, USA
srsmith@fsu.edu ; **aps10d@my.fsu.edu** ; jelya@coaps.fsu.edu

Shipboard Automated Meteorological Oceanographic System

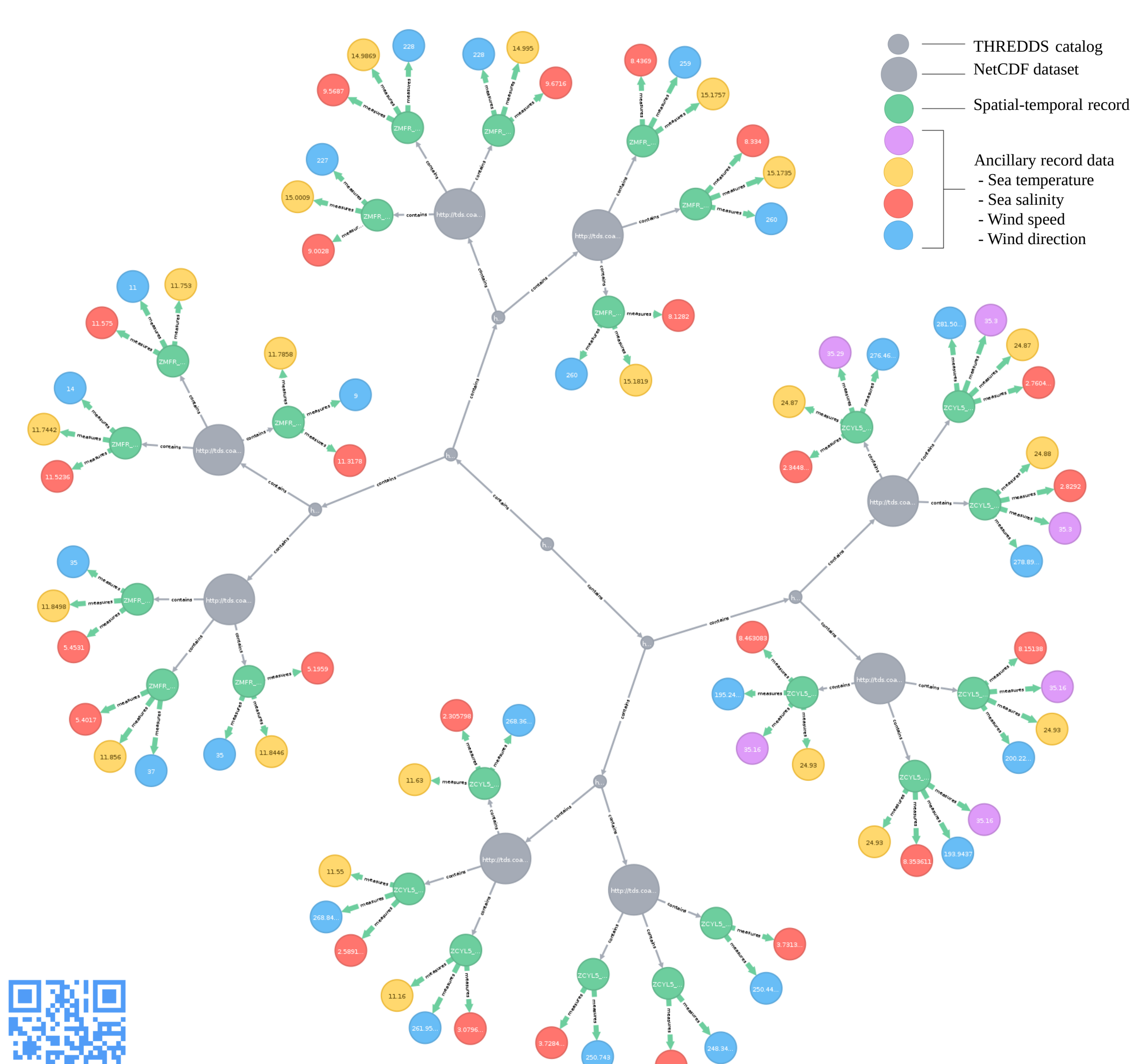
- Provides accurate, high-quality marine meteorological and near-surface oceanographic observations from research vessels and select voluntary observing ships.
- ~30 vessels for 2005 – present
  - Vessels operated by WHOI, SIO, U. Hawaii, U. Washington, U. Alaska, BIOS, NOAA, USCG, USAP, IMOS, SOI, LUMCON
  - ~30-40K one-minute observations per month, per vessel
- All data undergo scientific quality control
- http://samos.coaps.fsu.edu/html/index.php

Provenance

- SAMOS data storage lies on a Thematic Real-time Environmental Distributed Data Services (THREDDS) server.
- Data files are presented as network Common Data Forms (NetCDF) which are designed for high throughput of large array like data, especially multidimensional.
- A common problem with high resolution data sets is finding efficient chunking patterns such that users are able to query and subset the entire data set efficiently.
- For SAMOS, the original data is broken down into the following hierarchy:
  Quality Level -> Ship -> Year -> Day
- While quality and ships are at the top of this hierarchy, the time component of SAMOS is a primary divisor that can be used to chunk data at various scopes of resolution. For this reason, we chose to take advantage of the Time-tree model in Neo4j, provided by Graph Aware technologies.

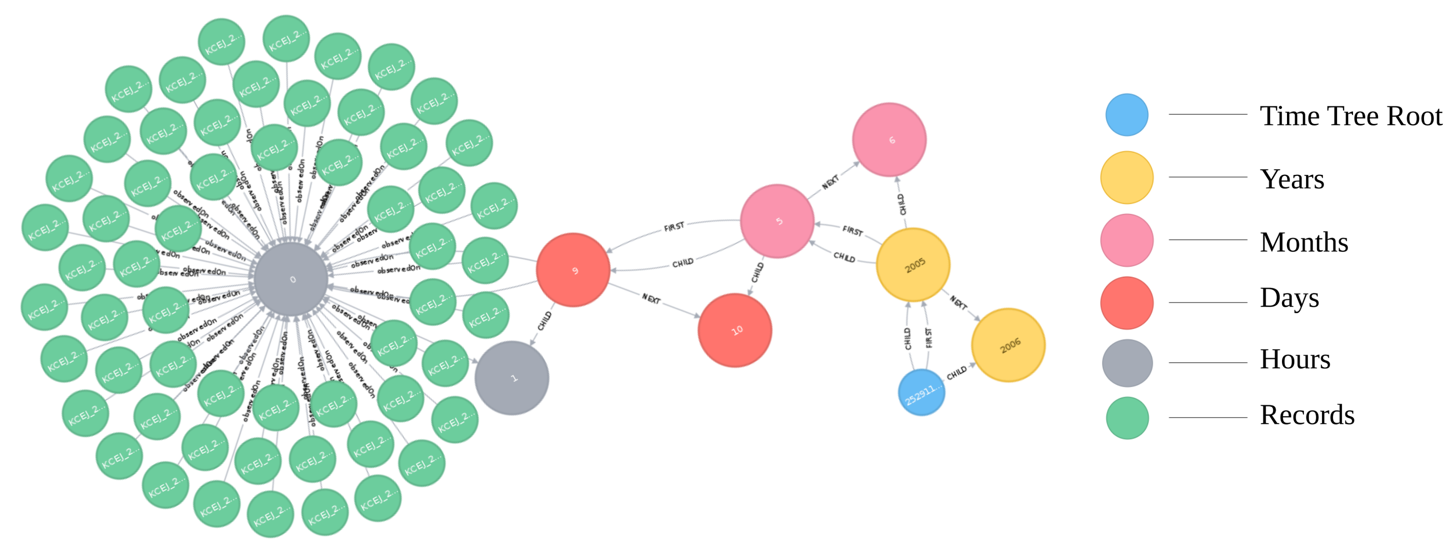
Data Import

- To facilitate this process various python libraries are exploited including:
  - Siphon by Unidata
  - NetCDF4-python by Unidata
  - Neo4j bolt driver
- Communicating with THREDDS and NetCDF is simple and performant using the libraries provided by Unidata, as these technologies were designed for high throughput of aggregated data and largely backed by a C interface.
- When performing an initial loading procedure, Neo4j's CSV bulk import tool is used to send ~28 GB worth of CSV text files into a hierarchical graph modeled exactly as it is in each catalog on the THREDDS server. The advantage of the hierarchy is to provide a mechanism for operating on the graph in chunks to avoid memory intensive write queries.
- A major structural change lies at the NetCDF data file level. Rather than storing contiguous large arrays for each variable component, each individual observation is given a node for auxiliary spatial-temporal components, and related nodes for ancillary data variables. This is motivated by the fact that SAMOS observations always incorporate space and time, but vary regarding oceanographic measurements.



Spatial-Temporal Modeling

- Graph Aware is a Neo4j consultancy, training, and software development agency that provides off the shelf plugins, including one for generating a Time-tree directly into the graph via Java stored procedures.
- This model is event oriented. An update is made by providing a node, value of time, and resolution of time.
- The SAMOS graph uses a single tree structure, which means that all ships are clustered together.
  - An advantage of this is a reduced graph size. SAMOS ships have similar time frames and each is at 1-minute resolution. Dividing ships into separate Time-tree instances would require a very large set of nodes, and each tree would be highly redundant.
  - One disadvantage is that more complex logic is necessary to process data according to ship identity. How this affects performance is yet to be tested, and largely depends on the memory model of Neo4j.



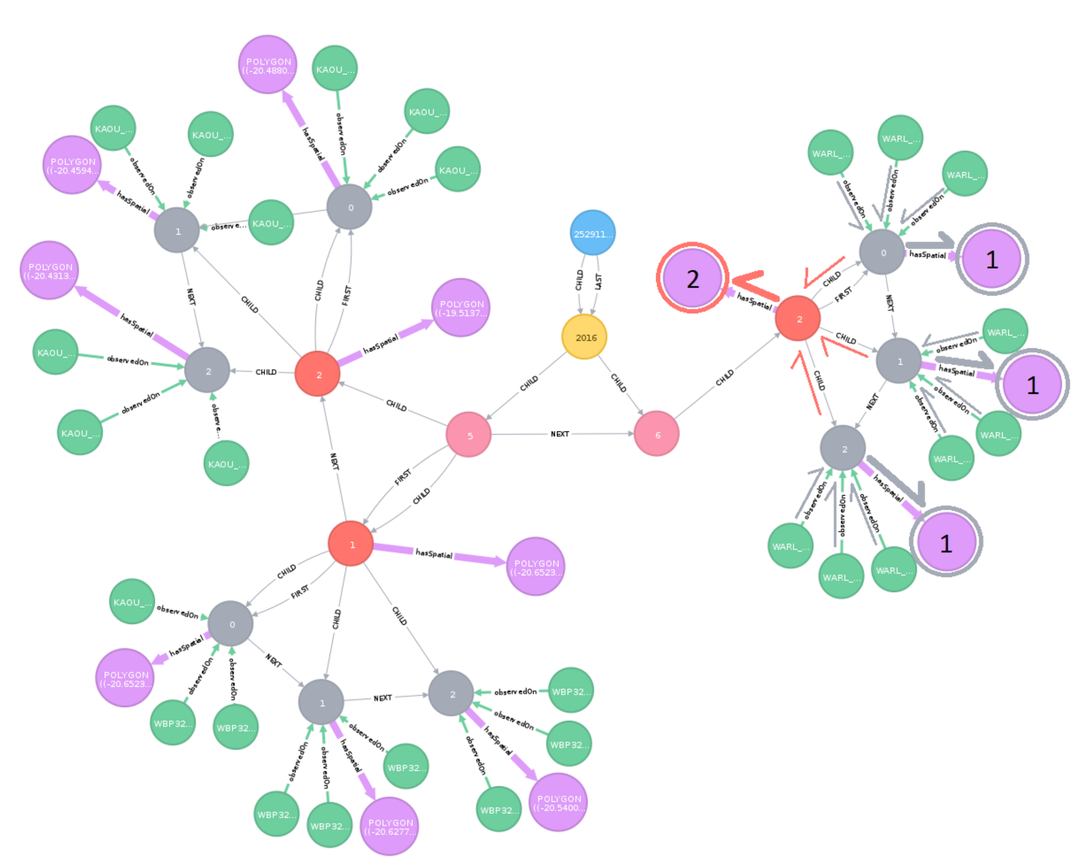
- Neo4j Spatial is a library plugin which is part of the Neo4j open source contribution ecosystem. It provides an array of spatial procedures, as well as a mechanism to deploy an R-tree index directly into a graph.
- This model was deployed and tested with a subset of SAMOS, but demonstrated significant performance problems. This is likely due to the nature of the data, in that each ship has a dense trajectory reporting spatial measurements at very high resolution in a singly linked fashion.

- A large requirement of accessing SAMOS data is querying on space and time. Originally when testing the Time-tree and R-tree models, a CYPHER statement would request disjoint results from each tree, and then intersect each result set. This proved to be unfeasible, however the Time-tree is still a performing candidate. That being said, the plugin is being actively developed and may be useful for it's other spatial operations.

A Time-Space Model

- Many researchers have studied the problem of spatial-temporal structures in different kinds of applications. For querying SAMOS, a structure is needed that can eliminate large portions of the graph during early query processing given some spatial-temporal criteria.
- Examples covered in literature include a 3D R-tree, MV3R-tree, HR-tree, TB-tree, SETI, and others which each have domain specific advantages and disadvantages.
- The TB-tree and SETI models work by aggregating intervals of point data into line polygons for indexing. These will be explored further in future work.

- Currently under development is a model featuring a Time-tree that incorporates metadata to suit the needs of spatial queries. This was inspired by pyramid structures used in computer vision applications.
- When elements are loaded into the tree, an agglomerative process merges spatial features from the event-leaf nodes up to the root node.
- The shapely python package is used to merge point data into polygons using a convex hull. Up to 60 leaf nodes for an hour will be used for each ship. The same process is used when merging polygons in the internal nodes of the tree, however a cascaded union precedes the convex hull computation. Shapely is backed by OSGEO, a C++ library, and therefore is fast.

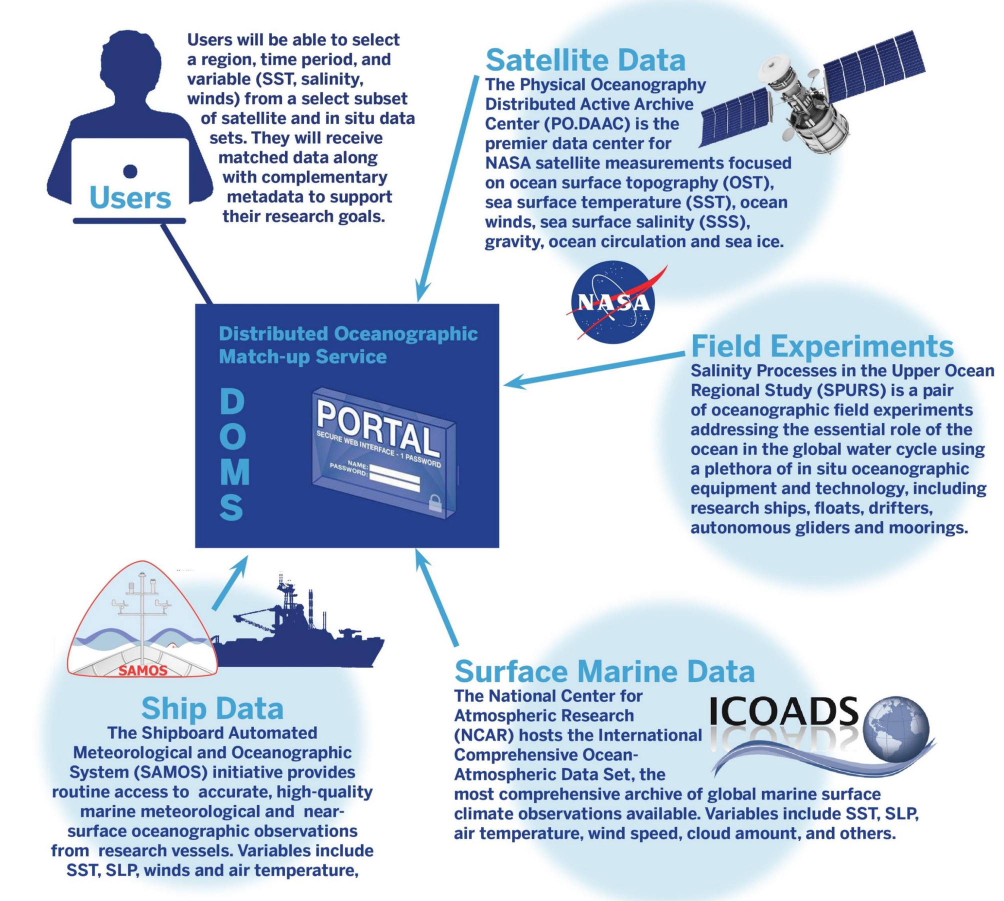


1) Spatial information from a collection of records is merged and assigned to a WKT node attached to the lowest structural nodes in the Time-tree.

2) WKT strings contained in nodes attached to a layer of the Time-tree are merged and assigned to nodes attached at the next layer with lower time resolution.
(Hours -> Days -> Months … etc.)

Distributed Oceanographic Match-up Service

The **Distributed Oceanographic Match-Up Service (DOMS)** is a collaborative effort between COAPS, NCAR, and NASA JPL. The service is currently in the prototype phase and went online in August, 2016. DOMS reconciles satellite and in situ datasets in support of NASA's Earth Science mission. The service provides a mechanism for users to input a series of geospatial references for satellite observations and receive the in situ observations that are matched to the satellite data within a selectable temporal and spatial search domain. DOMS includes several characteristic in situ and satellite observation datasets. It will be used by the marine and satellite research communities to support a range of activities. The service is designed to provide a community-accessible tool that dynamically delivers matched data and allows the scientists to only work with the subset of data where the matches exist.



Apache Solr

- Solr is an enterprise grade document storage and retrieval engine that provides full text search. It is extremely capable and delivers excellent performance on filter queries. The primary use case for DOMS is searching on space and time, and future work may incorporate queries on ancillary variables to the interface as well.
- With the project underway, a Solr index resides at each endpoint in the network. Some are using supplementary technologies such as Apache's Cassandra and Spark, but Solr is the primary ingredient in this ecosystem.
- An interesting question is whether or not similar performance can be achieved when using alternative data retrieval methods, in this case Neo4j.
- A large part of pushing this boundary will depend on the architecture of the graph, how well the Time-Space model handles spatial-temporal queries, and optimizations that exploit the memory model used by Neo4j. This may require additional Java stored procedures in order to exploit parallelism and other optimizations that are not easily achievable directly through CYPHER.

Future Work

- The majority of data engineering work has been done, though there is much to be done for performance testing, and investigation into the internals of Solr and Neo4j.
- Benchmarking is non-trivial since each component is housed on the same machine and competing for resources. This is not an ideal environment, so administration work will be done to normalize each testing suite.
- More research will be covered regarding spatial-temporal databases and other relevant work. It would be interesting to see if other models can be incorporated into the graph for comparison against the model implemented for SAMOS.

- SAMOS is hierarchical in nature and extremely dense. It is possible that graph solutions may not have a competitive advantage for the requirements of DOMS. Graph databases are typically used to discover or represent relationships between objects. While each ship in SAMOS is managed separately in the tree, there may be merit in exploring new relationships between them at various scales of time resolution.

- Finally, it is possible that repurposing this graph structure into a metadata utility might be useful for DOMS, or even other work at COAPS. For example, it might be useful as a mechanism to cache matched data in DOMS, whether it be for individual records or even metadata regarding aggregations of matches. Future work might involve using DOMS to collect matched data and generating metadata using the time space model. Depending on the volume of data which is discovered in DOMS, it may be possible to store a global set of matched records which could be accessed directly for a dramatic improvement in query response time to a user.

Acknowledgements

The DOMS project is supported via NASA's Earth Science Technology Office from the Advanced Information Systems Technology program. The project is funded at FSU (lead institution, NNX15AE29G), JPL, and NCAR (NNX15AG22G) via individual grants to each partner.

ESTO Earth Science Technology Office