4th International Workshop on Climate Informatics
September 25-26, 2014
Hosted by the National Center for Atmospheric Research in Boulder, CO

SCI-WMS:PYTHON BASED WEB MAPPING SERVICE FOR VISUALIZING GEOSPATIAL DATA

Brandon A. Mayer[1,2], Brian McKenna[2], Kelly Knee[2]

***Abstract*—SCI-WMS is an open-source python implementation of an OGC WMS [1] service for qualitatively assessing society-critical oceanagraphic applications including: forecasting, risk assessment, model comparison and algorithmic/parameter selection. The modular cross-platform implementation of SCI-WMS allows the service to keep pace with the rapid developments in the geospatial data science community, able to produce visualizations for numerous types of model outputs with transparent support for both structured and unstructured geo-referenced topologies. This abstract outlines the implementation and technology stack for visualizing geospatial coastal forecasting (CF) data using SCI-WMS[1] and details the deployment of SCI-WMS for visualizing model data and simulations within the scope of the U.S. IOOS Coastal and Ocean Modeling Testbed project. [2]**

I. MOTIVATION

The U.S. Integrated Ocean Observing System (IOOS) Coastal and Ocean Modeling Testbed (COMT) was formed to unify otherwise disparate entities in government, academia and industry to leverage the proliferation of oceanagraphic data and modeling techniques to combat natural and man-made coastal stressors by accelerating the turnaround from research and development to operational application of society-critical applications including: forecasting, model comparison, model skill assessment, and algorithmic/parameterization improvements [2]. Key to the U.S. IOOS COMT mission is an extensible and universally available tool for quickly visualizing and assessing a diverse set of coastal modeling data. SCI-WMS is a general OGC WMS solution for serving rasterized visualizations of geospatial data which has been deployed for the COMT project to provide visualizations of a wide range of scientific data.

II. SCI-WMS

SCI-WMS is an open-source implementation of the Open Geospatial Consortium's (OGC) Web Map Service (WMS) standard which specifies an HTTP interface for generating rasterized visualizations of geospatial data [1]. SCI-WMS is implemented in Python using the Django[2] web framework and standard cross-platform numerical software, NumPy and Matplotlib [3], [4] for generating visual content. Additionally, the open-source python implementation provides a cross-platform WMS solution which can leverage the suite of tools developed by the geospatial data analysis community such as pyugrid[3] to maintain pace with the latest geospatial software and standards developments such as unstructured grid support and CF-UGrid Compliance[4].

Vital to the efficiency of SCI-WMS is the abstraction of an oceanagraphic dataset into two entities: a topology, defined as a geo-referenced spatial structure and numerical model output as visualized in figure 2. SCI-WMS creates a local topology cache for efficiently computing spatial neighborhoods with respect to topology structure. For storage efficiency, model output is not replicated locally but referenced via OGC compliant web-services. Because geospatial WMS requests are commonly restricted to a subset of the earth surface, SCI-WMS uses the local topology to compute the subset of model data needed to fulfill each request prior to accessing the external data. Furthermore, by classifying topologies as either regular or unstructured, efficient algorithms and data structures are exploited to optimize the computation of relevant model data subsets.

III. DEPLOYING SCI-WMS FOR THE U.S. IOOS COMT TESTBED

While SCI-WMS is a general software solution for geospatial visualization, it is a key component in realizing the U.S. IOOS COMT mission, facilitating qualitative model comparisons and aggregation through a unified visualization framework. Figure 1 outlines the cyberinfrastructre behind the deployment of SCI-WMS for the COMT project.

Brown University School Of Engineering[1], RPS-Applied Science Associates, South Kingston RI[2], Brandon_Mayer@brown.edu, BMcKenna@asascience.com, KKnee@asascience.com

[1]https://github.com/asascience-open/sci-wms

[2]https://www.djangoproject.com/

[3]https://github.com/pyugrid/pyugrid

[4]https://github.com/ugrid-conventions/ugrid-conventions/blob/v0.9.0/ugrid-conventions.md

4th International Workshop on Climate Informatics
September 25-26, 2014
Hosted by the National Center for Atmospheric Research in Boulder, CO

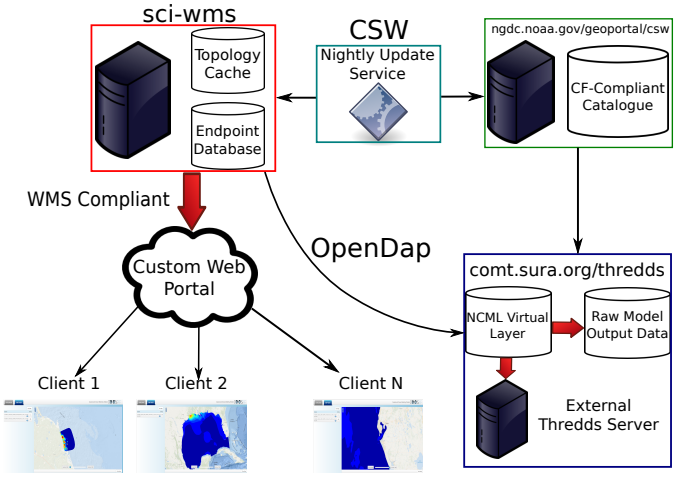

Fig. 1. Overview of the SCI-WMS deployment for the U.S. IOOS COMT project. SCI-WMS updates its topology and endpoint database via a nightly service which queries CF-Compliant datasets cataloged by NGDC. Model data is hosted on an external web server through an NcML facade accessible to SCI-WMS through OpenDAP as a single NetCDFdata structure. SCI-WMSthen responds to http requests made simultaneously by multiple clients interfacing through a custom built web-portal.

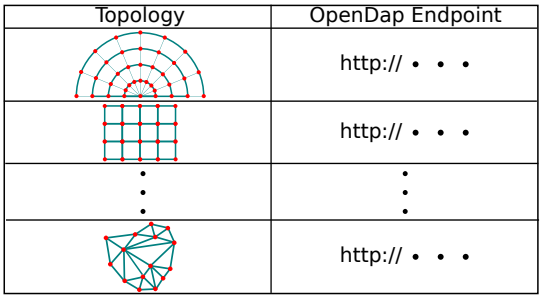

Fig. 2. SCI-WMS topology and endpoint data store. Typologies are classified as c-grid and u-grid for efficient geospatial queries and remote model data access.

Raw coastal data is hosted by the Southeastern Universities Research Association (SURA) on a dedicated server for the COMT project [5]. Each data set may consist of multiple files in different formats, and may be the result of very different models run by various institutions with disparate computing resources. However, accompanying each dataset is an NcML virtual layer which exposes each dataset as a single NetCDF object which may be accessed via OpenDAP[5]. Furthermore, the NcML facade presents a consistent set of meta information in accordance to CF-Conventions [6] so services like SCI-WMS can access the data through a uniform interface.

The NOAA[6]–NGDC[7] Geoportal indexes public geophysical datasets and provides an OGC Catalogue Web Service (CSW) to query datasets by their metadata attributes. Sci-wms queries the NGDC Geoportal at regular intervals updating both the topologies and OpenDAP links for all new or modified datasets.

SCI-WMS is designed to be scalable and responsive to handle queries for visualizations of any registered dataset by multiple users simultaneously.

SCI-WMS is used to visualize data from the first phase groups of COMT program: *estuarine hypoxia, shelf hypoxia and coastal inundation* [2]. For each modeling group, SCI-WMS successfully generates visualizations from ADCIRC, FVCOM, SELFE and SLOSH models run by scientists at MDL, UND, USF, VIMS, UMASS, DAL, TAMU and NOAA. The IOOS COMT use case is an example of how SCI-WMS can be leveraged as an effective tool for delivering consistent visualizations of scientific data to a diverse community utilizing various sophisticated modeling technologies.

IV. CONCLUSION

SCI-WMS is an Python implementation of the OGC WMS standard leveraging the suite of tools and utilities the geophysical science community has developed and continues to develop for advanced data analysis. The rapid development and integration of unstructured grid support into a WMS compliant service is demonstrated throught the efforts to display and analyze the IOOS COMT program model output. Ongoing development is in progress for sci-WMS to support emerging geophysical datasets such as enesemble model output as well as provide clear visual support for the assessment and quantification of model skill and performance metrics.

[5]http://www.opendap.org/

[6]National Oceanic and Atmospheric Administration

[7]National Geophysical Data Center

REFERENCES

[1] Open Geospatial Consortium http://www.opengeospatial.org/standards/wms [Accessed: 2014-07-24], "OpenGIS Web Map Service."

[2] R. A. Luettich, L. D. Wright, R. Signell, C. Friedrichs, M. Friedrichs, J. Harding, K. Fennel, E. Howlett, S. Graves, E. Smith, G. Crane, and R. Baltes, "Introduction to special section on the U.S. IOOS Coastal and Ocean Modeling Testbed," *Journal of Geophysical Research: Oceans*, vol. 118, no. 12, pp. 6319–6328, 2013.

[3] S. v. d. Walt, S. C. Colbert, and G. Varoquaux, "The numpy array: A structure for efficient numerical computation," *Computing in Science & Engineering*, vol. 13, no. 2, pp. 22–30, 2011.

[4] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing In Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[5] R. A. Luettich, L. D. Wright, and S. Elizabeth, "SURA Final Report: A super-regional testbed to improve models of environmental processes on the U.S. atlantic and gulf of mexico coasts," tech. rep., SURA, May 2012.

[6] CF Conventions Document, http://cfconventions.org/latest.html.