AdcircViz: A Visualization Application for Distributed ADCIRC-based Coastal Storm Surge, Inundation, and Wave Modeling

A JHT 2013 Project
Middle of Second Year Progress Report

Abstract

High-resolution predictions of coastal storm surge, inundation, and waves generally require substantial computational resources, particularly in a forecast, real-time mode. Considering that a given storm only threatens a portion of the US coastline, high resolution is only needed in the coastal areas threatened by active storms and simultaneous and comprehensive spatial coverage of the entire US East and Gulf of Mexico coasts is unnecessary. Regional forecast systems using the ADCIRC tide, storm surge, and wave model have been operating in several risk-prone areas of the US East and Gulf coasts for the past several years. The ADCIRC grids are high-resolution regional discretizations of the topography, bathymetry, critical hydraulic structures, and land cover embedded in a larger scale coarser ADCIRC grid to simplify offshore/open boundary conditions. These systems use high-performance- computing (HPC) resources that are often locally provisioned but can be located at any available facility (e.g., NOAA's "jet" HPC systems in Boulder, CO).

ADCIRC developers at the University of North Carolina (UNC) have created the ADCIRC Surge Guidance System (ASGS) to provide end-to-end ADCIRC forecasts. However, the distributed nature of ASGS instances requires an efficient way to discover, gather, organize and visualize the distributed results from multiple ASGS instances and other models. Therefore, UNC's Renaissance Computing Institute (RENCI) has developed a "data grid" to federate ASGS instances and provide uniform access to forecasts. To access and visualize results, we have developed AdcircViz, a MATLAB-based visualization application that provides robust access to ASGS forecasts in the data grid. In this two year project, we will deploy, extend, and support AdcircViz and work with the National Hurricane Center (NHC) storm surge team and JHT staff to make AdcircViz readily available for their operational environment. We will demonstrate the preliminary application version, gather requirements for and implement new capabilities, and provide documentation and support during active storms for which ASGS instances are running. While our project focus is on ASGS based results, the data grid methodology and AdcircViz application can be applied to other models if providers use recently developed community standards for storing and serving model results. This document reports on the project's accomplishments to date and anticipated activities in the next 6-month period.

During the prior 6-month period, the application was renamed StormSurgeViz. However, the project name still references AdcircViz, and AdicrcViz is used below when referring to work performed prior to the name change.

1) Overview of the AdcircViz Project

The community of expert ADCIRC system operators is organizing around an ADCIRC Data Grid that effectively creates a single collection of ADCIRC results. This is illustrated in Figure 1. The content of each THREDDs server is harvested into a master catalog file that AdcircViz retrieves and presents to the user through the GUI.
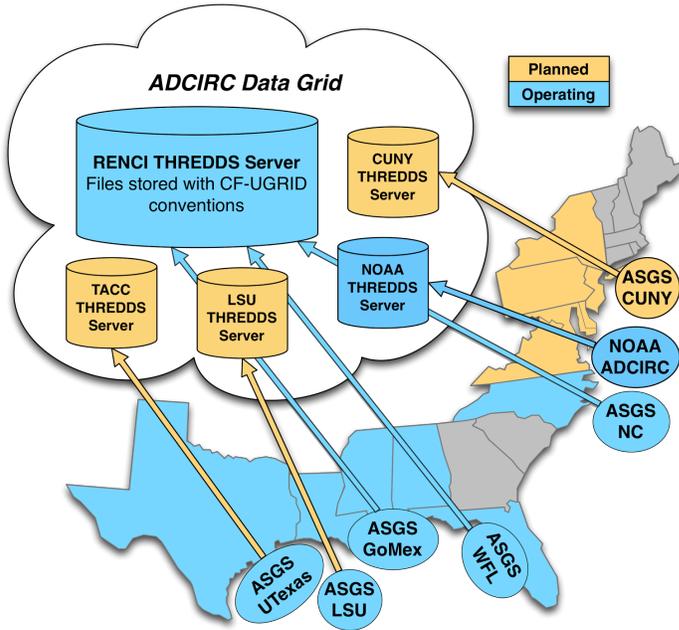

Figure 1: Current set of THREDDs servers (blue) that comprise the ADCIRC Data Grid. NOAA Coast Survey Development Laboratory has configured a server to host their experimental ADCIRC simulations. These simulations, computed for both tropical events and daily weather conditions, can be directly accessed and visualized with AdcircViz.

AdcircViz relies on making connections to an THREDDS Data Server (TDS) using the OPeNDAP data transfer protocol through the MATLAB toolbox nctoolbox. This toolbox, which uses the Unidata netCDF java class as the data access layer, is used extensively to open netCDF data files and access metadata, without accessing the data components until they are needed (called *lazy loading)*. Each graphic is generated using MATLAB functions and tools "on the fly". The color map, color limits and color increment are specified by the user with default values appropriate to the scalar field being rendered. The default colormap is one similar to the NOAA colormap currently used for NHC surge maps (Figure 2).

Low   High

Figure 2: Default AdcircViz Colormap based on the colormap used for NHC storm surge graphics.

For the original proposal to the JHT program, we developed a prototype AdcircViz implementation to demonstrate the feasibility of the approach. This version of AdcircViz (Figure 3) included the ability to visualize several scalar fields from ADCIRC (namely maximum water level, time-dependent water level, maximum significant wave height, time-dependent wind speed, and the maximum wind speed) for all ensemble members contained in an ASGS simulation directory. The data and metadata for populating the application are extracted from ADCIRC files posted on a THREDDs server (opendap.renci.org:1935). The files are in netCDF format with strict adherence to the CF-UGRID conventions and standards, and in a specific directory (folder) hierarchy that defines information such as

storm name and advisory number. This prototype version was extensively tested during the 2012 Atlantic hurricane season, particularly during Hurricane Isaac. We have also tested AdcircViz with large Hurricane Katrina simulations in a hindcast mode, and this has proven to be a useful way to examine existing non-forecast simulations, as long as the files adhere to the CF-UGRID conventions and are manually put onto a THREDDs server.
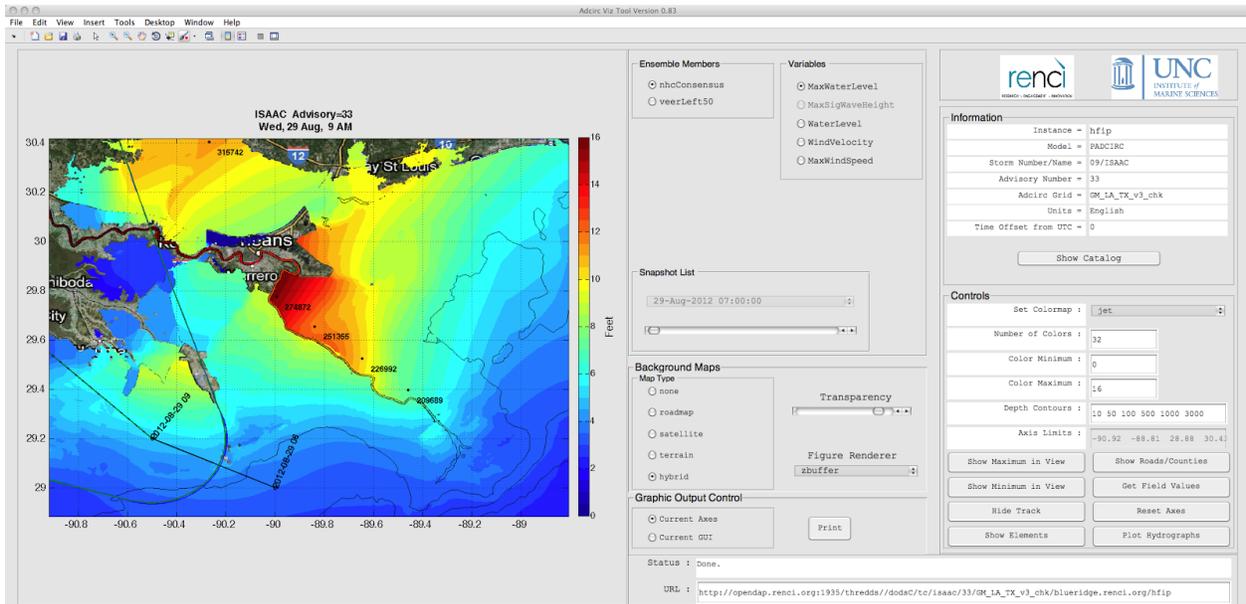

Figure 3: Prototype AdcircViz application developed for the original proposal to JHT. This version was extensively tested during the 2012 Atlantic hurricane season and deployed to test/evaluation users for 2013.

2) Status, Advancements and Progress to Date

Status and remarks relative to the previous progress report are made in bold italics, succeeding each relevant paragraph.

The current version of AdcircViz was made available to the NHC Storm Surge Unit (SSU) for deployment and initial evaluation prior to the 2014 Atlantic hurricane season. A presentation describing the project was made at the recent 2014 Interdepartmental Hurricane Conference, and afterward we met with the SSU to discuss specific requirements for AdcircViz, including specific features, code access, machine/operating system dependencies, and how to optimally engage for bug fixes and feature requests.

During initial testing at the NHC, it was noted that the GUI response can be sluggish. To explain this behaviour, note that the underlying MATLAB toolbox nctoolbox exploits the Java netCDF "lazy-loading" feature. This means that when an ADCIRC data file is opened in MATLAB, only the metadata in the netCDF file is initially transferred to MATLAB. If the ADCIRC grid contained in the data file has not been previously downloaded in the current application version, then the grid variables are also extracted from the netCDF file. Lazy-loading functionality is critical to overall application performance and is *in lieu* of downloading the entire netCDF data structure, most of which is not needed. ***[Note that this issue has been identified and resolved. See below.]***

This means that AdcircViz is already downloading only the minimum amount of data to make display images, and that some amount of slow response is expected due to the size of the ADCIRC model

3

grids and any local internet speed issues.   In an effort to mitigate the impact of local internet speed issues, we have established a virtual machine at RENCI into which NHC can remote-desktop and operate AdcircViz. This alternative puts AdcircViz in direct connection with the THREDDS server (the VM and THREDDS servers are on the same internal network). This option has not been tested by NHC yet, but this will be evaluated between hurricane seasons.
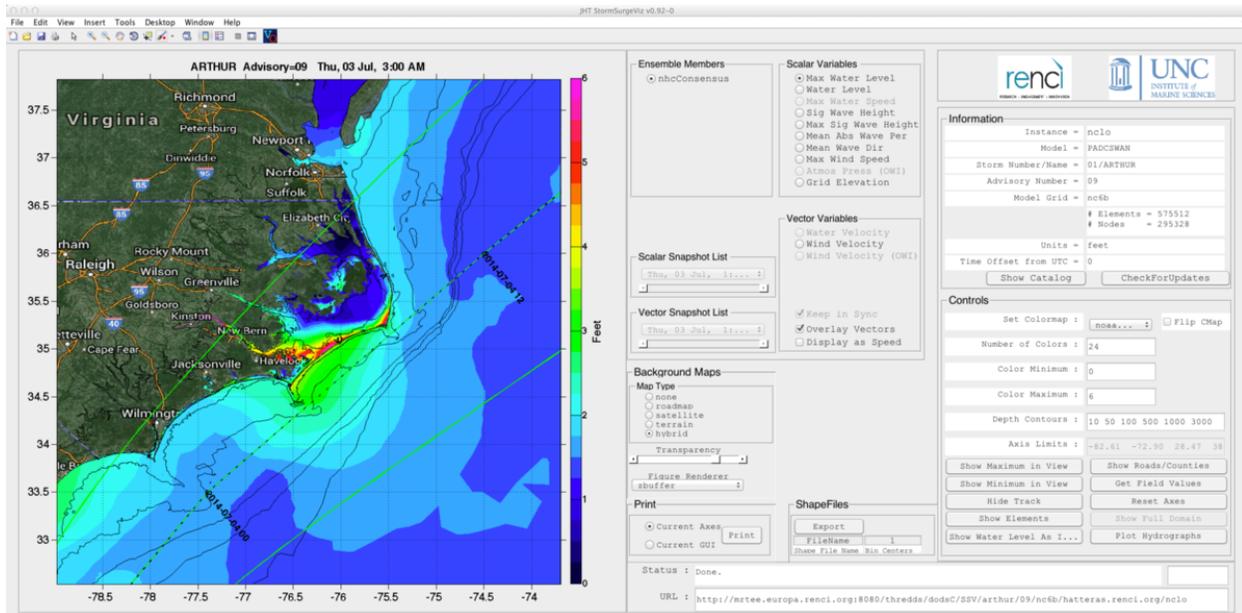

Figure 4: AdcircViz showing the RENCI Adcirc Surge Guidance System forecast for Hurricane Arthur, Advisory 09. The colors are the maximum water level, including tides and wave setup, over the duration of the forecast simulation. The dashed line is the forecast storm track, with the uncertainty lines in green.

Due to a relatively quiescent Atlantic hurricane season, there have been few "real time" opportunities to rigorously evaluate AdcircViz as a tool within the NHC's SSU. However, Hurricane Arthur, which made landfall on the North Carolina coast on August 27, provided an opportunity to exercise AdcircViz. We (RENCI) were able to monitor the status of the forecast system during the event. Each new advisory's simulation results were detected by AdcircViz and were immediately available to AdcircViz. Figure 4 shows the maximum water level [ft MSL] for Hurricane Arthur, Advisory 09.

After the submission of the end of year progress report, Hurricane Gonzalo occurred in mid October 2014. This hurricane did not pose a threat to US coasts, instead veering north and east toward Bermuda. The ADCIRC grids used in most ASGS forecasting systems extend eastward to the 60-deg meridian, thus providing coverage of Bermuda, albeit not at substantial resolution. RENCI contacted NHC/SSU about using the Gonzalo ADCIRC-based forecasts on RENCI's THREDDS server as an opportunity to experiment with StormSurgeViz (the application formerly known as AdcircViz) in a low-stress manner (NHC/SSU does not have operational responsibilities for Bermuda).

At the time of this test/experiment, the NHC/SSU reported substantial delays in the response of StormSurgeViz , noting that 10's of seconds elapsed when the application requested data slices from the THREDDS server. The perception was that StormSurgeViz would not be efficient enough operational use. Subsequently, RENCI determined that the sluggishness was not due to the data servers, and in general the data transfer speeds between the servers and other remote sites (not NHC) were consistent with expected data speeds.

The PI (Blanton) attended the 69th IHC, with the JHT projects attending in Miami. [Note that this meeting date is outside of the reporting period, but it is essential to capture some activities that impact the assessment of StormSurgeViz.] During that meeting, 2-4 March 2015, we attempted to recreate the performance problems noted during the Gonzola test. We could not repeat the slow response. It was determined that network switching equipment had been upgraded in the intervening period, and the poor response of StormSurgeViz may have been due to a network equipment issue (either outdated/old or faulty equipment). During the IHC/JHT visit, the StormSurgeViz application performed as expected by both the PI and the SSU. The network transfer speeds were at least as fast, if not faster, than when StormSurgeViz is run at RENCI, where the data servers are located in close proximity to the workstations used by the PI for development and testing of the application.

Project accomplishments in September 2014 through February 2015:

Documentation: The user's guide for StormSurgeViz has been substantially expanded and refined, providing overviews and details of its capabilities and features. *It now includes a detailed description of how other storm surge model output can be described and posted such that they are available to StormSurgeViz, and thus to the SSU, in a consistent way.*

Test/Evaluation Opportunities: *Hurricane Gonzalo (mid-October 2014) presented an opportunity to test StormSurgeViz without typical operational pressures in the SSU. RENCI's Adcirc Surge Guidance System was running during this storm, with wind and pressure forcing from the ATCF advisory sequence. Figure 5 shows the maximum significant wave height for Advisory 20, in meters. The ADCIRC grid is also shown, to indicate the relative coarseness of the grid in deep water.*
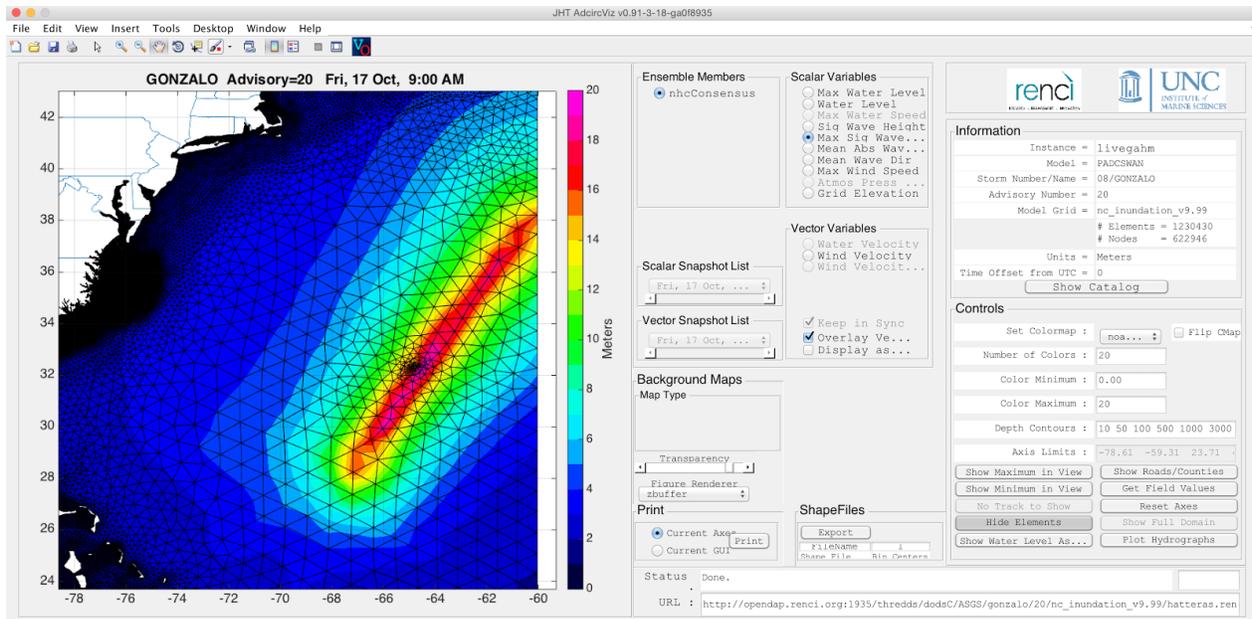

Figure 5: StormSurgeViz showing the RENCI Adcirc Surge Guidance System forecast for Hurricane Gonzalo, Advisory 20. The colors are the maximum significant wave height in meters. The triangles are ADCIRC's finite element grid.

Rebranding: In the *prior* 6-month period, we renamed the AdcircViz application to StormSurgeViz and refactored some of the MATLAB code to present a model-neutral, UGRID visualization application. The

project GitHub page has been updated to reflect the rebranding. The new project GitHub landing page ([here](#)) is shown in Figure 6.
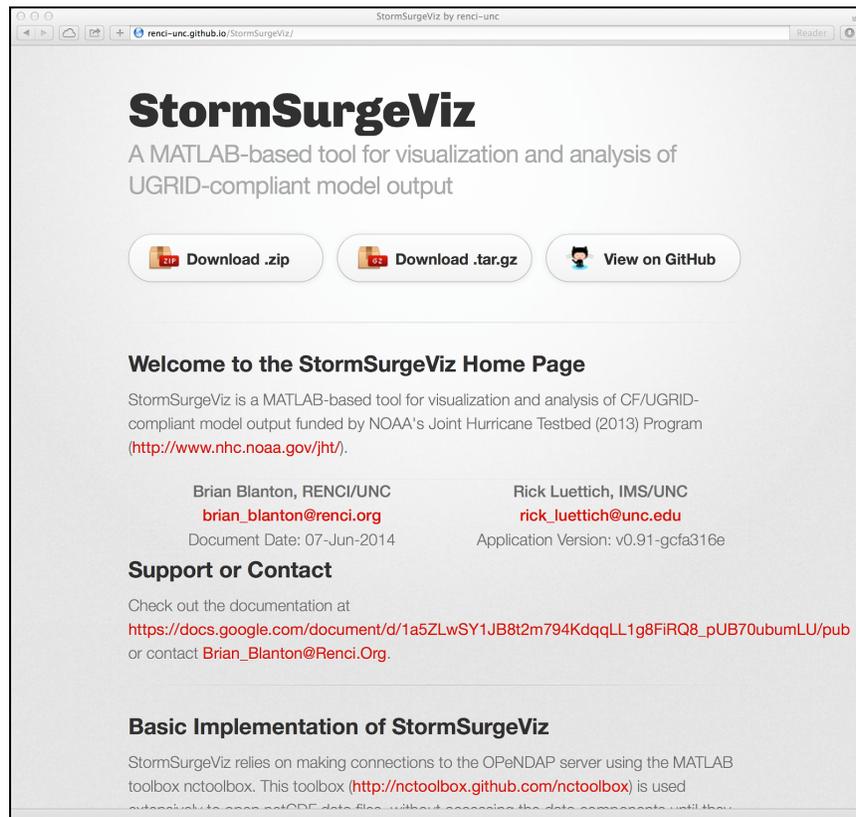

Figure 6: StormSurgeViz project GitHub landing page. This page provides easy access to the code, as well as links to documentation and support.
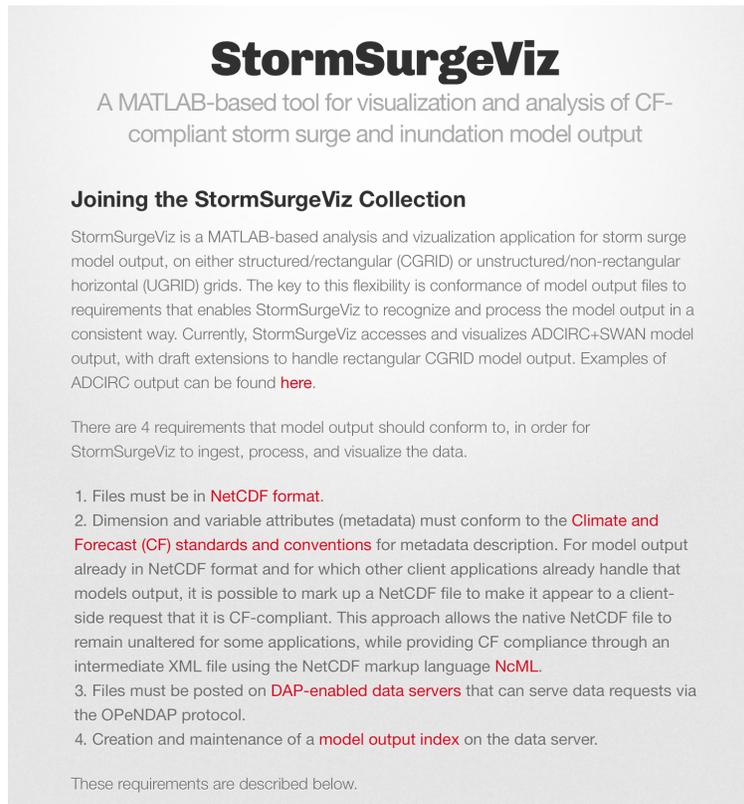
3) Year 2 Activities and Goals

There are several aspects of the StormSurgeViz under development for year 2 of the project, focused on enabling the NHC to access and visualize storm surge model output beyond ADCIRC results available on THREDDS servers.

Community Outreach: In conversations with the NHC and JHT program, there is a need to enable the storm surge modeling community at large to provide storm surge hazard assessments to the NHC SSU during active tropical cyclone periods. StormSurgeViz and the underlying THREDDS Data Server mechanics could be used for this "community outreach" purpose. RENCI/UNC will devote resources to enabling this in between hurricane seasons. This past summer, the NHC drafted a letter for distribution to the community that describes the need for a consistent interface to the variety of storm surge models. Specifications for this interfacing already exist, as contained in the CF-UGRID conventions and related work of the NOAA IOOS Coastal and Ocean Modeling Testbed (COMT). RENCI/UNC will work with the NHC to refine both the letter and the documentation needed for other groups to implement the THREDDS Data Server and the required metadata in the simulation output files. This is further described below.

RENCI has recently provided to NHC the draft community participation guidelines. The guidelines are specified in a new section to the GitHub-based project documentation, reached from the "Participation

6

in StormSurgeViz Activities" section of the document landing page. The four components to participation, from the model output perspective, are netCDF formatted files, adherence to the CF variable attribute and metadata conventions, file posting to THREDDS data servers, and maintenance of an indexing file on the THREDDS server. These requirements are fully described on the community "on-boarding" documentation page, http://renci-unc.github.io/StormSurgeViz/participate.html, shown in Figure 7. RENCI will also host other model data on our THREDDS servers, if useful.

Figure 7: StormSurgeViz project GitHub page for the community onboarding process.

NcML: NcML is an xml markup language for netCDF files. It allows for augmenting netCDF metadata, aggregation of multiple netCDF files, and other features *without* modifying or altering the actual netCDF files. NcML can be used on existing netCDF files to make the files appear to be CF compliant. It can also be used to aggregate multiple netCDF files so that they appear to be one, larger file. This is typically used to aggregate across time (if separate netCDF files are output for each output time) or if different dependent variables are output into separate files (as is the case for ADCIRC). NcML has proven to be essential to "marking up" the metadata for compliance without any changes to the underlying model output files. Our THREDDS server configuration document includes coverage of NcML, leveraging the experiences and outcomes of the IOOS testbed. *RENCI will also work with other modelers to construct NcML files for their model output.*

StormSurgeViz modifications: StormSurgeViz will be modified (extended) to work from the standardized NcML descriptions. The mechanics of handling the UGRID data structures will not change.

This has been started, but not yet complete. We anticipate that it will be completed by the start of the 2015 Atlantic hurricane season.

4) A note about SLOSH visualization

Since SLOSH uses a rectangular grid, it is straightforward to "convert" SLOSH to CF-UGRID. We have recently demonstrated this capability by mapping a SLOSH output file from the NOAA IOOS Coastal and Ocean Modeling Testbed into a UGRID-compliant triangular description. We are developing a workflow that can be used to perform this conversion "on-the-fly" so that the existing internal StormSurgeViz mechanics can be used for SLOSH. Figure 8 shows a Hurricane Ike (2008) maximum water level result from SLOSH within StormSurgeViz. However, in order for StormSurgeViz to access SLOSH results, the solution files need to be posted on a THREDDS server in the appropriate directory structure and with an accompanying catalog file. RENCI can host the results and make the catalog file, but we need to identify how the solutions will get from where the results are computed to RENCI's THREDDs server.
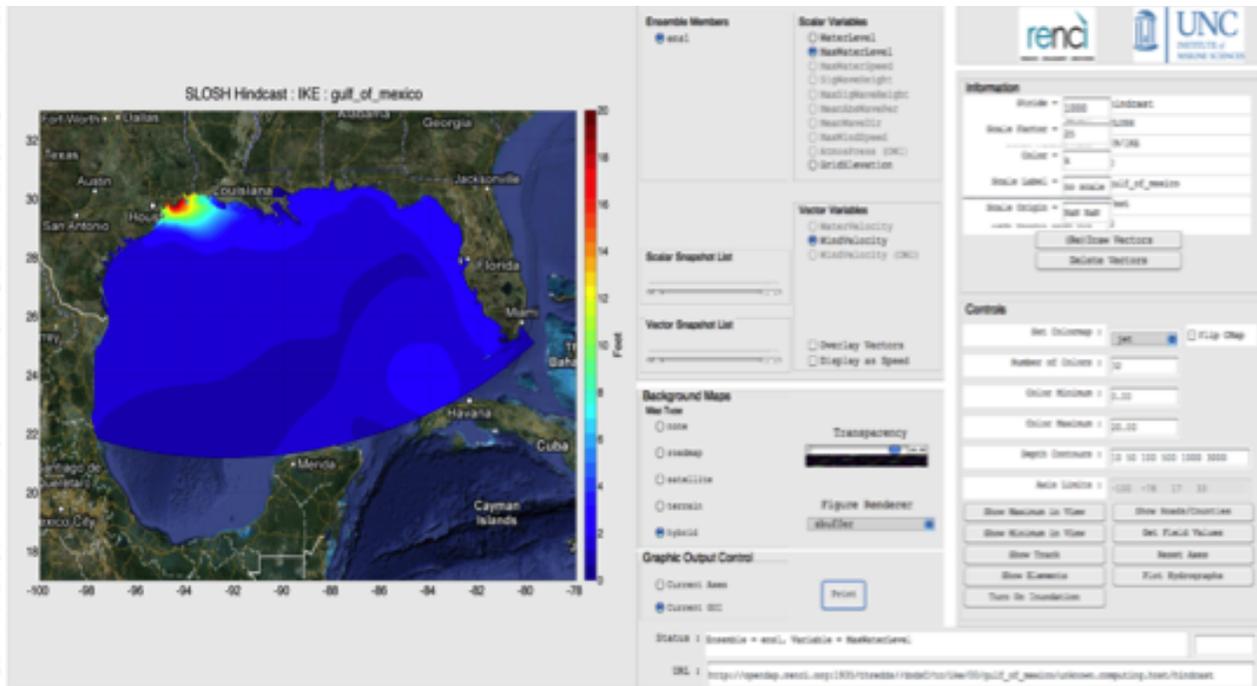
Figure 8: Demonstration visualization of SLOSH output in AdcircViz. A SLOSH Hurricane Ike (2008) simulation from the NOAA IOOS Coastal Ocean Modeling Testbed project was converted to CF-UGRID and posted on a THREDDs server.

5) URLs

THREDDS: http://www.unidata.ucar.edu/software/thredds/current/tds/
NcML: http://www.unidata.ucar.edu/software/thredds/current/netcdf-java/ncml/
NCTOOLBOX: http://nctoolbox.github.com/nctoolbox

StormSurgeViz GitHub webpage: http://renci-unc.github.io/StormSurgeViz
StormSurgeViz GitHub Community Participation webpage:
        http://renci-unc.github.io/StormSurgeViz/participate.html

RENCI/UNC THREDDs server: http://opendap.renci.org:1935/thredds
NOAA/CSDL THREDDs server: http://coastalmodeldev.data.noaa.gov/thredds