Recent Advancements of the COAMPS-OS® Dashboard

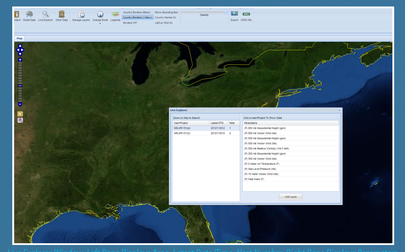
A Web GIS Tool for Military Meteorologists

Chad Hutchins[1] • Dan Martinez[2] • Mike Frost[1] • John Cook[1] • Daniel Geiszler[3] • Gary Love[3] • Qingyun Zhao[1]

[1]Naval Research Laboratory, Monterey CA
[2]Computer Sciences Corporation, Monterey CA
[3]Science Application International Corporation, Monterey CA

New and Improved Client Features

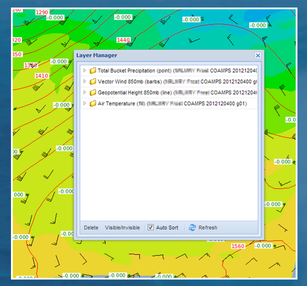
Live Spatial Data Query



Live Explorer Window. Left Pane Displays Area, Latest Date/Time, Nest Number. Right Pane Displays Parameters.

Live spatial data query is the ability to dynamically send a geo-location to a server while a user moves about a map and have it return available/pertinent data. This feature is now available in the Dashboard and called the Live Explorer. As a user moves about the world map, the Live Explorer in real time sends the user's map view to the server and retrieves the data that is pertinent to the map view. A user can then select a particular layer and add it to their map. The Live Explorer enables a user to quickly find and display data for an area of interest.

Layer Order Automation

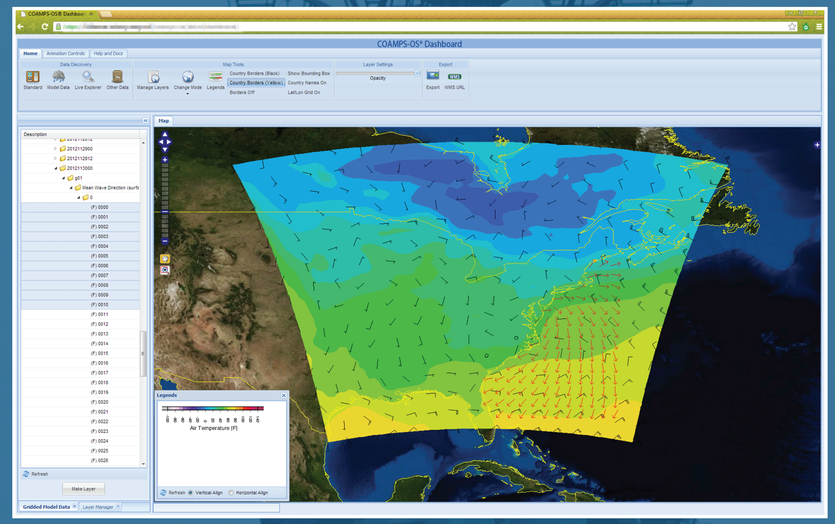


Layer Manager Showing Layers in Order Corresponding to Layer Height, Top Most Parameter Above Others.

In the previous Dashboard, keeping polygon layers below line layers and below point layers was a manual process. Now a one stop Layer Manager handles keeping layers in order, as well as changing the visibility, and permanently removing a layer from the map. The Layer Manager can also be set to all manual drag-and-drop ordering of layers. Layers are automatically organized by the Layer Manager into folders. Editing of individual layers is allowed for maximum flexibility.

Interface Redesign

Ribbons • Windows • Animation



The Dashboard interface was completely redesigned using Sencha® Inc.'s Sencha GXT framework for Google Web Toolkit®. The framework includes many built-in tools and design controls for creating web applications. The redesign focused on quick access, enhancing popular tools, and creating new features.

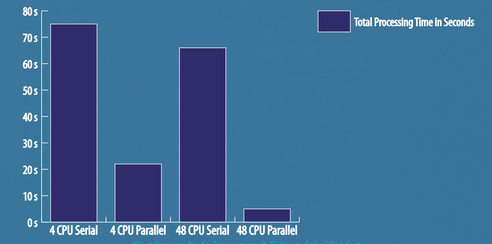
Ribbon Bar • Similar to Microsoft Office® suite, a ribbon of easily recognizable icons highlights major features. Larger buttons provide a larger clickable space. Stacking ribbons within tabs creates classes of ribbons which minimize searching for tools. The ribbon bar addresses user complaints on the difficulty of finding tools in the previous version.

Windows • Windows in the Dashboard are resizable containers that are fully movable within the web browser. The goal of windows is to maximize workspace without adding complexity. Windows can be docked into a resizable, tabbed container to allow the user to organize their work space.

Animation • Animation controls have been moved to their own ribbon. This allows for larger, easier to use buttons, and groups related controls into a single interface. The internal animation logic was also enhanced to use a "sliding window" method of rendering layers. This allows for much smoother animation of layers across different browsers and also reduces the amount of memory utilization during an animation sequence.

Server and Speed

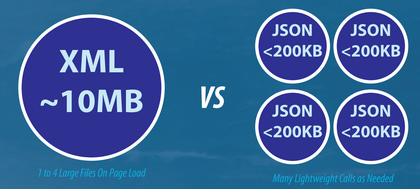
Multi-core vs Serial Processing



Client Request: 2m Air Temperature, 0-72 Hours, Color Filled Contour

Layers dynamically generated from the client interface were processed serially, which result in a significant wait time for the user while the layers were generated on the server. For example, a 72 hour forecast for 2 meter air temperature would require 73 calls to the server to generate the layers sequentially. Using built in Java support for concurrency, the Dashboard server was enhanced with a new parallel processing module along with an internal ticketing system which allows multiple clients to submit parallel jobs and monitor their status. The new parallel processing system allows the Dashboard server to utilize all the CPUs available to quickly process layers for the Dashboard clients. The same example now takes 1 call for the request and only a few for the status. This significantly reduced the amount of wait time for the user and provided a much better user experience when using the Dashboard.

JSON vs. XML Communication



1 to 4 Large Files On Page Load

Many Lightweight Calls as Needed

In previous versions of the Dashboard, potentially large XML files were retrieved from the server and processed by the client to display available layers. This design did not scale well and would slow down the client initialization with a large number of layers included in the XML files. A new mechanism was designed using dynamic queries and JavaScript Object Notation (JSON) output which significantly improves the performance of the client initialization as well as the resource consumption (memory, CPU, etc.) on the client machine. The new mechanism utilizes an automated 'fetching' system which only retrieves from the server necessary data as the user browses through the available product layers.

Background

In 2011, the Naval Research Laboratory released the first version of the Coupled Ocean/Atmosphere Mesoscale Prediction System (COAMPS®) – On-Demand System (COAMPS-OS®) Dashboard into operations at the US Navy's Fleet Numerical Meteorology and Oceanography Center (FNMOC). The Dashboard is a web Geographic Information System (GIS) interface for creating, viewing, and exporting global and mesoscale model fields, mesoscale data assimilation results, radar and satellite data overlays, model verification information, and other standard GIS products in a properly geospatially-registered display for military users. The Dashboard also has the ability, utilizing GIS standards, to display other GIS and geospatial intelligence information.

Previous Works and Links

2012 AMS Manuscript

This Poster's URL

Contact

Chad Hutchins
Meteorologist
Naval Research Laboratory - Marine Meteorology Division
Monterey, CA
chad.hutchins@nrlmry.navy.mil
www.nrlmry.navy.mil

Chad Hutchins' Contact

Legal

Sponsored by the Office of Naval Research