

2020 Oct 4

Exploring MapStore as an Opensource Web Mapping Framework

For the past weeks I have been exploring MapStore as a web mapping framework. MapStore is developed by [GeoSolutions](#) and is opensource so you can use it freely and modify as you wish. “MapStore is based on OpenLayers, Leaflet and ReactJS, and is licensed under the Simplified BSD license.”

I think MapStore has the potential to be a great alternative to ESRI for those looking for an opensource solution. Mapstore has 3 components (Web Maps, Dashboards, and Story maps) in a single framework. I'm thinking a setup with GeoServer/PostgreSQL/PostGIS combo (GIS server serving as data catalog), Apache/Tomcat (Webserver) plus Mapstore (front end web app) along with QIGS (desktop GIS) would make a decent opensource web GIS stack.

As far as I understand, the components of MapStore are modular (via plugins) and can be fully customized to users needs. If you are a developer then you can fully make this your own. If not, you can still use it by just making some configuration changes (with perhaps some help from your IT team to install with your own GeoServer setup). Or you can hire a developer or why not hire GeoSolutions, the developer of Mapstore - they provide commercial support for MapStore and GeoServer and will help you set up everything (I am not associated with GeoSolutions in anyway. I just like the work they're doing in the opensource realm).

Note, I am only exploring MapStore mainly as a user who want some easy configuration. This posting is my take on things, from set up on my local machine and changing some configuration settings.

Installation of MapStore

The installation process was easy enough. Instructions can be found on the [Quick Start](#) manual, so I won't repeat it. On the [MapStore documentation](#) page are separate well documented [User Guide](#) and a [Developer Guide](#). MapStore can be installed on a Windows or Linux system via the binary package or war package route. The easiest is to download and install via the binary package - this provides a contained sandbox with examples ready to be explored. I went the war installation route with Tomcat on my Windows system.

Configuring the MapStore User Interface

This is the default user interface of MapStore2. Obviously, I want to customize the look of this.

The screenshot shows the MapStore web application interface. At the top, there are tabs for 'Featured' and 'Contents'. A green banner on the left says 'Fork me on GitHub'. The main header area contains the 'MapStore' logo, the tagline 'Modern webmapping with OpenLayers, Leaflet and ReactJS.', and a link to the 'documentation page'. A search bar is located below the header. The 'Contents' section shows 'Maps (1)', 'Dashboards (0)', and 'StoryMaps (0)'. Under 'Maps', there is a 'Sample Map' thumbnail. Below the map, there is a pagination control showing '1' of 1 item. The 'Keep in touch and stay up-to-date with the mailing lists' section includes Google Groups links for 'Subscribe Users Mailing List' and 'Subscribe Developers Mailing List', each with an email input field and a 'Subscribe' button. There is also a LinkedIn group link for 'Mapstore LinkedIn Group' with a 'Follow @mapstore2' button and '1,219 followers'. At the bottom, there is a 'GeoSolutions' logo and contact information.

Annotations on the screenshot include:

- Change Logo**: Points to the MapStore logo.
- Change the Banner image and text**: Points to the main header area.
- Remove "Fork me on GitHub"**: Points to the green banner on the left.
- Remove this section**: Points to the mailing lists section.
- Remove this section**: Points to the footer area.

So to change the looks of the default interface, I need to make some minor changes to the configurations files - see the Developer Guide [Configuration](#) section. I'm only making some changes to the front end configuration files, mainly the following files:

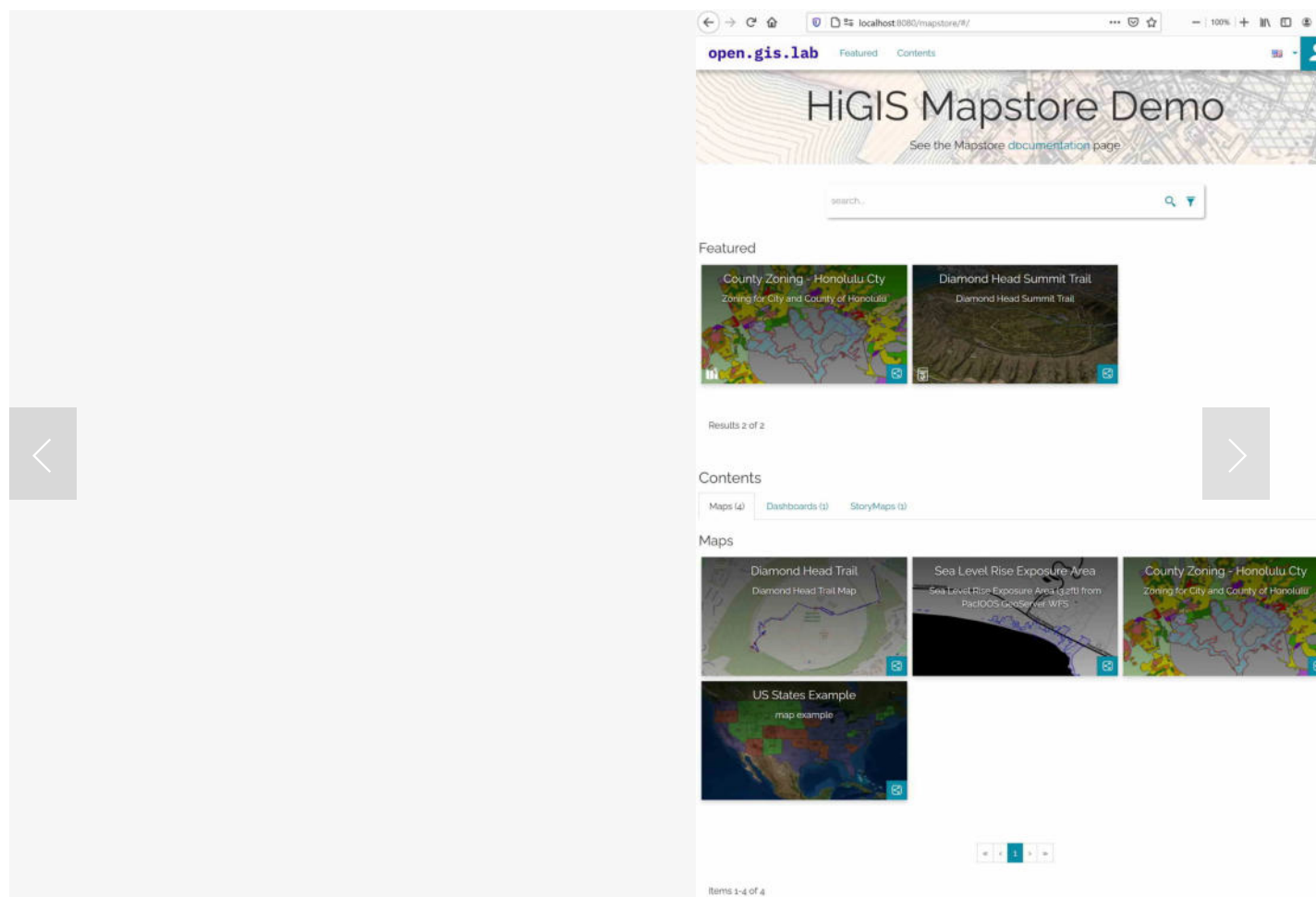
localConfig.json: this is the configuration file where you can over ride the logo, customize the banner title, and remove the mailing list and footer. This is also where you can specify your data catalog(s) for the application. Because I don't have current have a GeoServer to serve as a data catalog I included two public ones: Hawaii Statewide GIS Program's ESRI Mapserver (WMS) and PacIOOS's Geoserver (OWS).

new.json: this is the file where you can make changes to the default setting when creating a new map. For my configuration, I have set the map extent to show the Hawaiian Islands, change the CRS to EPSG 4326, zoom level, and added two new basemaps (Stamen terrain and ESRI Imagery).

data-en-US.json: this is the file where you can remove the “Fork me on Github” green banner and customize the subtitle in your banner (e.g. the HomeDescription text). This is the file for the English version of the application, but you can find other translation files in the Translations folder.

Configured MapStore Application

Below are some slides to show what my configured application looks like. If you'd like to explore MapStore without doing any installation, then check out the MapStore [demo page](#).



My first impressions of and thoughts on MapStore is that it's a very good tool that is very flexible. Used in conjunction with GeoServer, PostgreSQL/PostGIS, and QGIS - you could create a nice opensource GIS stack solution. I think it's worth a look. There is a bit of a learning curve - at least for me since I'm not a developer - but still it was doable. The only thing I had a little trouble with

was the dashboard component - I could not quite get the data layers (via a public GeoServer WFS) set up properly to make the dashboard widgets to function the way I want it to. But then again this could be how the public GeoServer WFS may have been set up in the first place since it's not something I have control over. I might have more luck with setting up my own GeoServer. The MapStore framework has other functionalities that I haven't explored, but overall I am impress with what MapStore has to offer.

I hope this post was useful for those looking for an opensource web GIS options. Thanks for reading. Until next time.

Comments (2)

Newest First

Daniel 6 years ago · 0 Likes

Hi there! The problem you had with getting the data to build the widgets, can be related to the CORS. Take a look at this web, which helped me to solve the data gathering from Mapstore.

Regards,

Daniel

Daniel 6 years ago · 0 Likes

<https://stackoverflow.com/questions/22363192/cors-tomcat-geoserver>

< Wind Vector Mapping and Animation in QGIS

Mapping and Viewing Geotagged Photos in QGIS >

open.gis.lab



© 2026. open.gis.lab