2021-04-02

GEOGloWS HydroViewer: Open Software-as-a-Service for Localizing Global Hydrologic Forecasts of the Group on Earth Observations Global Water Sustainability Initiative

Kyler Ralph Ashby
Brigham Young University

GEOGloWS HydroViewer: Open Software-as-a-Service for Localizing Global Hydrologic Forecasts of the Group on Earth Observations Global Water Sustainability Initiative

Kyler Ralph Ashby

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

E. James Nelson, Chair
Dan P. Ames
Gus Williams

Department of Civil and Environmental Engineering

Brigham Young University

ABSTRACT

GEOGloWS HydroViewer: Open Software-as-a-Service for Localizing Global Hydrologic Forecasts of the Group on Earth Observations Global Water Sustainability Initiative

Kyler Ralph Ashby
Department of Civil and Environmental Engineering, BYU
Master of Science

Earth observation data is increasingly ubiquitous, easily accessible, freely available, and generally usable due to improvements in software, data standards, network infrastructure, and national policies. As a result, greater opportunities arise for using these data in a wider field of application including decision support for local and regional environmental and water resources management efforts. In parts of the world where in situ data are less readily available, global Earth observation data used in such decision support tools can be a boon to underfunded government and private water management agencies. The United Nations Group on Earth Observations Global Water Sustainability initiative (GEOGloWS) works to coordinate such solutions, bringing global water management capabilities to local decision makers. The recent development and deployment of a global hydrologic modelling system based on historical simulations and daily ensemble forecasts from the European Centre for Medium-Range Weather Forecasts (ECMWF) using Earth observations and streamflow routing on every river of the world results in a highly informative and potentially transformative dataset for users at local scales. However, for this data to reach its full potential at the local level, it needs to be subsetted at a regional or local scale, presented in a local geographic context, and interpreted in terms of local water management challenges. Furthermore, this subsetting allows for customization to support the way information is used and the kinds of decisions that are made. This paper presents the design, development, and experimental testing of the GEOGloWS HydroViewer, which is an open source, web-based software that effectively localizes global ECMWF forecasts to meet the needs of water managers and decision makers through subsetting the mapping and modelling services and supporting other customization as needed. The unique Software-as-a-Service (SaaS) deployment method, developed and tested here, allows for individual water management agencies to automatically generate custom HydroViewer applications that can be managed and/or customized depending on need and capacity in-country without reliance on external software and capacity, removing typical interdependence relationships that often define technology transfer to developing countries.

Keywords: ECMWF, GEOGloWS, Streamflow Forecasts, Software-as-a-Service

ACKNOWLEDGEMENTS

I would like to thank Brigham Young University (BYU) for helping fund this research and providing the tools, help, and environment to let me work at my best on this research. I also thank them for providing me the opportunity to learn and work in ways I thought were beyond my reach. I thank all members of the BYU Hydroinformatics Lab for providing the technical support and feedback that helped bring this research into fruition and for fostering a professional, friendly environment. In particular, I want to recognize Riley Hales who more than generously gave his time and knowledge in aiding this research. I also want to acknowledge Chris Edwards, who provided key feedback and collaboration.

I would like to acknowledge Dr. Jim Nelson, who provided a tremendous amount of support and guidance during my graduate studies and this research. Likewise, I extend my gratitude to the other members of my BYU graduate committee, Dr. Dan Ames and Dr. Gus Williams, for their help and feedback.

Finally, I wish to thank my family who have supported me from the very beginning in all my endeavors. My mom and dad, Kaye and Kevin Ashby, never let me settle for anything less than my absolute best and inspired me every day to try harder and be better. Without a doubt, I would not have had these incredible opportunities without their constant inspiration and loving guidance. I also recognize my siblings, Kolby, Kameron, and Karson Ashby, for their encouragement. They helped give me the courage and resolve to push forward.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

EXECUTIVE SUMMARY

Water data is becoming increasingly available in today’s world. As technology advances, so too does our ability to collect and interpret data. This also means that the data collected can be used in a wider array of applications, which can help support or inform decisions regarding water resources. Several organizations are already taking advantage of this water data availability. GEOGloWS works to coordinate global data and the results derived from it. Part of their goal is to bring global water management capabilities to local decision makers. ECMWF, with help from various partners, developed and hosts a global hydrologic modelling system using observations and streamflow routing on every river of the world.

The purpose behind this research was to develop or improve the resources and tools necessary to deliver global water resource modelling data at a more localized scale. Subsetted water data can be used by local agencies to make decisions regarding the management of their water resources. While this is advantageous to users around the world, this is certainly beneficial to countries and governments who lack the infrastructure and manpower necessary to develop and maintain a reliable hydrologic model of their own.

My main contribution to this research was the development of a new global network of streams and catchments, which was referred to as a GeoFabric. This GeoFabric data can be better used in other applications because of the new organization system applied to it, which assigns a unique identifying ID to each feature (stream or catchment) contained within the GeoFabric. This GeoFabric also better represents regions of the world that had previously lacked appropriate spatial and modelling data.

Another important contribution was the development of a Tethys-based web application. The application has a global HydroViewer which displays the entire streamflow network included in the GeoFabric and can display streamflow data for selected stream segments. The application can also create subsetted data taken from the GeoFabric. The data can then be uploaded to either a custom GeoServer or HydroShare database. Doing so generates the data services necessary to display the spatial data on custom web maps. This data can then be used by the app to create custom HydroViewers which are exported as HTML files. These lightweight HTML HydroViewers can be further customized according to the needs and goals of the user who created it. The HTML HydroViewers can also be easily shared and hosted.

Though not part of my contributions to this research, a key part of it includes changing how the streamflow modelling results were accessed to be displayed in the web application and the custom HTML HydroViewers. Originally embedded as part of the web application that used it, the service, which is accessed through a REST API, was separated from the app. This allows the service and the data it contains to be more accessible to a wider array of applications and software.

All these resources and technologies were combined to deliver data and tools to those who can use it to better understand or predict the hydrologic processes occurring within their region of interest. This can help regions better manage their available water resources and be better prepared to meet the increased demand that will come.

This work is summarized in the following paper which has been prepared for submission to a scholarly journal.

1 INTRODUCTION

Expert, informed, and conscientious management and modelling of water resources is essential for providing a sustainable source of water for a growing global population. It is predicted that global water demand will only increase for the foreseeable future, which generates uncertainty as to whether natural supplies can keep up with this trend (Piesse, 2020; Vörösmarty et al., 2000). To compound this issue, many people facing water insecurity or instability live in countries with limited water monitoring infrastructure such as discharge and river stage gauging stations. Traditionally, donor organizations and programs have attempted to combat this issue by building streamflow monitoring stations and developing models targeted at countries where they are needed the most. While these programs provide significant benefit during the program and shortly after the program completion, they are not always sustainable in the long term. This is often the case because these countries do not always have the financial capacity or trained personnel to operate and maintain the models or collect and manage observational data when the intervention program's funding is gone, and yet the need for good hydrologic information persists.

Global hydrologic modelling is a promising potential solution that overcomes the issues arising from lack of financial and logistical support for developing local data and infrastructure resources (Fatichi et al., 2016; Wu et al., 2012). However, there are challenges and limitations to using global models because they either lack the detail or resolution needed to be of use outside of a global scale and/or are unable to deliver the necessary data to be used with confidence.

Hydrologic Modeling as a Service (HMaaS) is a novel approach suggested by Souffront Alcantara, et al. (2019) that addresses these barriers. HMaaS involves providing hydrologic model components, model results, and derived information to local end-users via web services using a Service Oriented Architecture (SOA) computing paradigm. Implementing this approach, a global hydrologic model can be built, optimized, monitored, and maintained by hydrologic experts, with immediate updates to model components, codes, and data sets immediately benefitting users of the system. This technique removes many barriers that keep essential water data out of the hands of those that need it most by pushing the funding and expertise responsibilities onto the agencies capable of delivering it. The various output data (i.e., historical simulation, forecasts, etc.) from the HMaaS model are made available through a standards-based web service to local end-users and are incorporated into their decision-making for flood and water resources management. To implement such a service for a global hydrologic model the following components are necessary:

1. A global spatial data network of streams and watersheds upon which the computational model is based and for which resulting streamflow data can be derived
2. An Application Programming Interface (API) service from which the modeled streamflow data can be retrieved
3. A web application or “web app” framework where the spatial data and modelling results can be viewed and explored

To help meet these goals, we previously created a global vector data network of streams and watersheds, or “GeoFabric”, designed to support a global scale web application

(HydroViewer) for visualizing streamflow forecasts (Souffront Alcantara et al., 2019). As additional customized HydroViewers for countries, regions, or watersheds were needed, it became clear that the foundational GeoFabric needed to be updated. Some of the weaknesses of the original GeoFabric included:

- The uniform area of watersheds as initially delineated was large, and in particular did not adapt to smaller regions which require modelling at a finer scale (e.g. Japan or the isthmus of Central America).
- There were not unique identifiers for all streams, nor a rational organization of the numbering for different areas globally.
- It was not designed for customization of application such as flood modelling, nor for extraction and services as a regional, country, or watershed version of a HydroViewer.

While an API was previously created to extract the hydrologic modelling results from our global streamflow forecasting system, this API was tied strictly to the web app technologies associated with the specific application developed. There was a need to separate the API from the web app and create a more standard way to implement the data services in the HydroViewer and other applications. Finally, better tools and methods needed to be provided for the creation and hosting of customized HydroViewers.

We have found the web app paradigm to be an ideal format to implement the HMaaS concept. Web apps allow for flexible access to anyone with a reliable internet connection. A web app can also be easily updated by its maintainers to the most current and accurate information and software upgrades. All users who access the web app benefit from the same updated data and software. These qualities complement the sustainability approach of HMaaS by removing the need for powerful computing resources and software licenses that would otherwise make intervention programs fail in the long term. Web apps are also highly customizable which allows the flexibility to adapt to a particular watershed, organization, or hydrologic issue on a user-by-user basis.

Souffront demonstrated the viability of the HMaaS method by creating an initial implementation of a global hydrologic model using cloud computing services and a series of web apps. However, these initial web app implementations identified additional improvements that could be made so that the full potential of local adaptation and customization capable with the service can be achieved.

The remainder of this paper presents work done to improve the underlying GeoFabric and data services and develop a web application that can provide both a global HydroViewer as well as tools for a good launch-point to create customized HydroViewers representing specific geographic extents and/or application of the hydrologic modelling data services. Not only can this help those at the local level better use the global data at their disposal but could also help them gain greater insight and confidence in the data they are using.

2 METHODS

The end goal of the HMaaS concept is to use a global model to provide actionable information to local stakeholders with decision-making needs for their river, watershed, or region. This is done through the development of a robust underlying GeoFabric from which an API can be constructed to make hydrologic modelling information accessible in combination with a web mapping service to visualize and uniquely identify each river as illustrated in Figure 1. Global raster datasets representing elevation are processed using watershed delineation tools in the Data Processing step. This results in the global GeoFabric which can be either subset for viewing (using the provided Web Map Services) or analysis (using the provided Streamflow Data Service). The GEOGloWS HydroViewer web app (global or regionally customized) uses both the visualization and data service capabilities as a core component of the HMaaS system.

The remainder of this section outlines the improvements to the GeoFabric that were required and the associated services necessary to deliver customizable HydroViewers.

2.1 Improving the GeoFabric

The previous attempt at creating and incorporating a GeoFabric for use with HMaaS revealed several shortcomings. It did not use unique identifiers for stream and catchment elements, but instead differentiated them for different global areas through a more programmatically complicated scheme of using region/watershed names with repeated stream IDs for each.

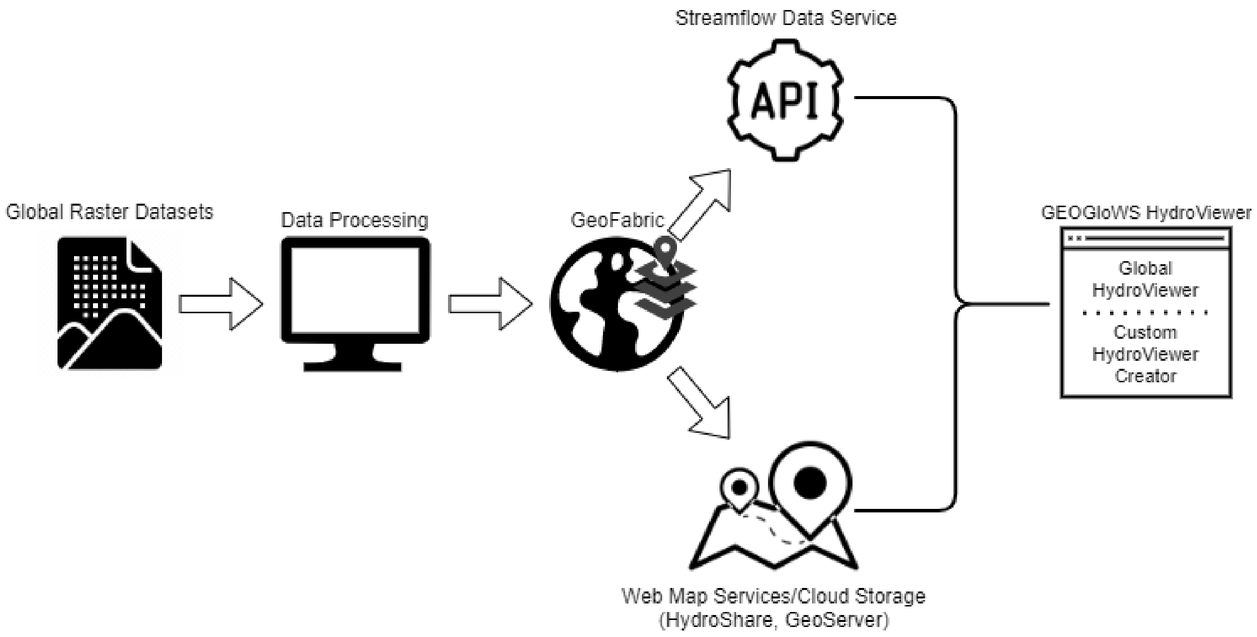


Figure 1: Core components of the GEOGloWS HydroViewer

Having a unique global stream identifier is essential not only in helping manage and update the data but also so that it can be used programmatically to be accessed by other software. Another shortcoming of the previous GeoFabric was that all areas of the globe were delineated with the same area threshold criterion. This resulted in smaller or narrower landforms with relatively small average watershed size such as Japan and Central America having important watershed areas unrepresented, which in turn led to reduced usability and confidence in the modelling results. Finally, the GeoFabric was created for the sole purpose of delineating watersheds that could be used for mapping of the gridded runoff and routing through a stream network. With the intention of moving beyond just streamflow analysis to floodplain mapping, it was desirable to have an underlying GeoFabric that might support this. The creation of a new GeoFabric was necessary to address these shortcomings.

The delineation of a new GeoFabric was accomplished using Esri ArcMap GIS software and ArcHydro tools. This combination provided the tools required to create and modify the necessary spatial data. The process of delineation requires flow direction and flow accumulation raster datasets derived from digital elevation maps (DEMs). Arc Hydro tools (Li, 2014) analyze these raster datasets to produce vectorized spatial data for streams and their accompanying watershed boundaries. The specific Arc Hydro tools used to create the updated GeoFabric include:

- Flow Accumulation – generates a flow accumulation raster from a flow direction raster
- Stream Definition – generates a raster that defines stream locations based on an upstream area threshold provided by the user
- Stream Segmentation – assigns a unique value to each branch of a stream network and creates a stream link raster which assigns unique values to network between intersections
- Catchment Grid Delineation – uses a flow direction raster and a stream link raster to create a raster of all the watershed areas
- Catchment Polygon Processing – converts the catchment raster into a shapefile with vectorized watershed features
- Drainage Line Processing – uses a stream link raster and a flow direction raster to create a shapefile with vectorized drainage lines
- Adjoint Catchment Processing – generates adjoint catchments

The Stream Definition tool, with its accompanying upstream area threshold parameter, directly affected the number of streams and catchments delineated and the size of the catchments. Hence, this was the main parameter adjusted when creating the GeoFabric.

Instead of producing one global delineation, we partitioned the earth into distinct regions and then delineated each region separately. Subdividing the GeoFabric not only aided in the organization of the data but also better used the computational resources available. We delineated each region with a different upstream area threshold parameter according to the geographic characteristics of the region. This was to ensure sufficient detail or resolution was captured to accurately model the hydrologic processes for the region, particularly for islands and regions such as Central America with relatively small areas as compared to larger continents. Upon completion, a delineated region was subjected to post-processing methods to be of use in the hydrologic modelling and data retrieval processes. For the hydrologic modelling process to work there needed to be a 1:1 ratio between the catchments and streams (Figure 2). Catchments that did not contain a stream were removed from the GeoFabric. Certain catchments that contained multiple streams were modified by splitting the catchment according to the underlying topography of the region or by removing the extraneous stream(s). Extra streams found in catchments were removed if they were deemed to be inconsequential to the integrity of the stream network or if their removal would have little impact on the flow generated in the modelling results.

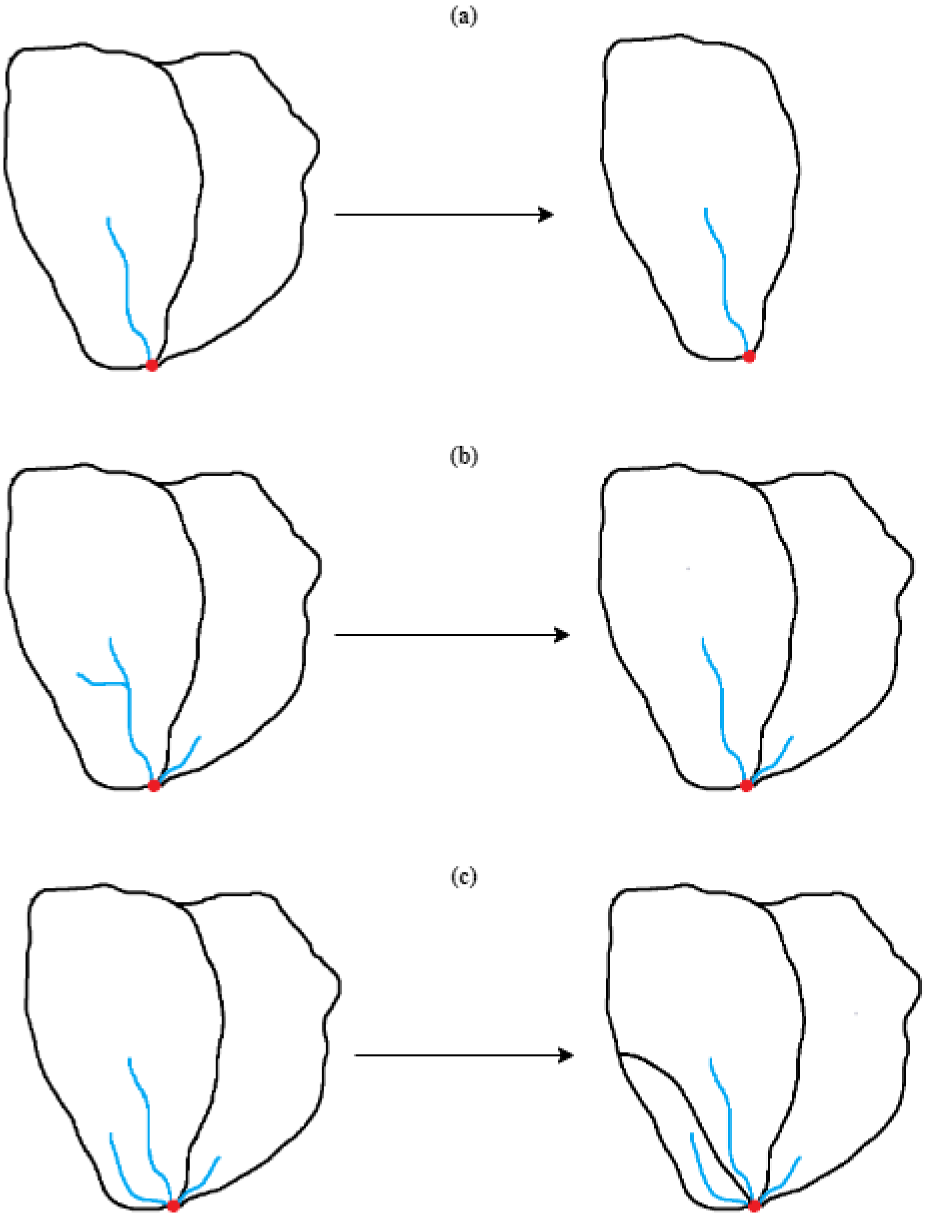


Figure 2: A 1:1 ratio was achieved by either (a) removing empty catchments, (b) removing extra streams, or (c) splitting catchments

The GeoFabric was derived from flow direction raster datasets that were provided by the Esri Living Atlas team who used the HydroSheds (Lehner et al., 2008) flow direction raster datasets and improved delineation of streams through “hydroconditioning”. Because of limitations in digital elevation data resolution and processing, the flow through known rivers is often obstructed, creating artificial or “digital dams.” Hydroconditioning adjusts the digital elevations in a way that forces delineated rivers to conform to known linear features that represent rivers. A disadvantage to using the HydroSheds raster datasets is that as part of the hydroconditioning streams are “burned” in, meaning the source elevation data were artificially lowered where streams are located. The major advantage of hydroconditioning the DEMs in this extreme way is that watersheds correspond to well-known and accepted river networks. However, extending the use of the system to other applications such as hydraulic models and floodplain mapping is limited when using underlying elevation have been so severely modified.

As a result, the use of the MERIT DEM (Yamazaki et al., 2017), which was created from a conditioning of DEM data that did not so drastically alter streams by burning them, with the intention of better supporting hydraulic modelling efforts, was explored to see if it could adequately infer streams and watershed boundaries. However, the tools used for automated delineations in many cases either placed streams erroneously or failed to correctly connect stream networks in the way they are known to exist. Adjustments could have been made manually as outlined above but were deemed too extensive for the available resources. For this reason, we determined not to use the MERIT DEM to create the foundational GeoFabric. Because the most important priority for developing a global hydrologic service was accurate delineation of stream networks and watershed boundaries, we ultimately chose to build the GeoFabric with the HydroSheds raster datasets. We note that a new version of the MERIT DEM, MERIT Hydro (Yamazaki et al., 2019), has been released since we created our GeoFabric. MERIT Hydro provides accurate automated delineation while preserving the MERIT’s known strength of better support for hydraulic modelling and floodplain mapping.

To aid with organization and facilitate data retrieval, each catchment and stream feature pair was given a unique numeric identifier. This unique ID contains a prefix that identifies which region the feature belongs to. The assigned prefixes are listed below in Table 1 and a map of the region boundaries can be found in Figure 3. The shapefile data were uploaded to HydroShare where it can be viewed and downloaded (*Derived Hydrography of World Regions*, n.d.).

Table 1: List of regions with identifying prefix and average catchment size

Region Name	**Number of Reaches**	**Avg. Catchment Area (km^2)**	**Prefix ID**
Africa	106,706	308	7
Australia	62,987	149	2
Central America	55,358	106	9
Central Asia	55,621	151	8
East Asia	79,931	154	4
Europe	80,769	149	12
Islands	31,894	100	1
Japan	5,302	100	3
Middle East	41,041	155	6
North America	40,802	250	13
South America	149,384	136	11
South Asia	84,095	100	5
West Asia	84,713	161	10

2.2 Streamflow Data Service

We used the streams resulting from the delineation process as the computational network for a hydrologic routing model. The hydrologic model consists of three components: the ECMWF forecasted surface runoff data product from the HTESSEL (Balsamo et al., 2009) land surface model, the mapping of runoff from the gridded model to the watersheds of the underlying GeoFabric, and Muskingum routing through the stream network of the GeoFabric as implemented through the Routing Application for Parallel computation of Discharge (RAPID) Software (David et al., 2008).

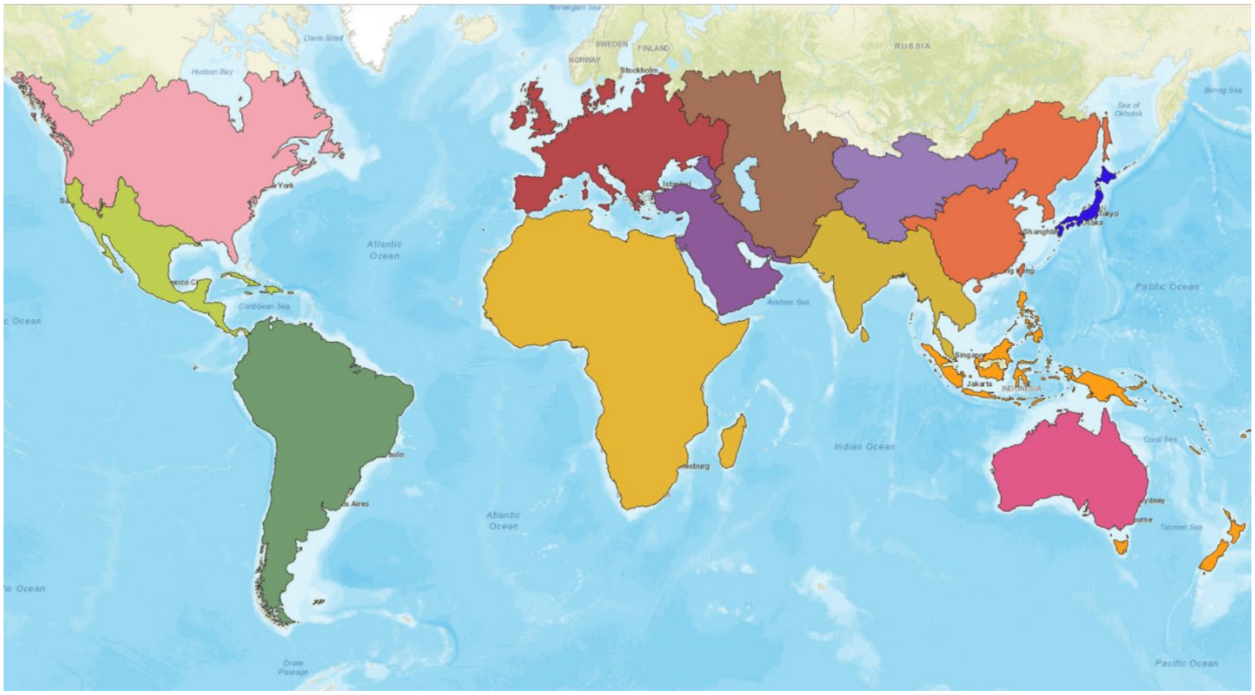


Figure 3: Boundaries of 13 global regions

The surface runoff dataset is a gridded dataset produced every twelve hours. The gridded forecasted values are mapped to the vector watershed boundaries using simple geospatial analysis methods. That volume of runoff water assigned to each basin is then routed downstream to the watershed outlet. The parameters for the routing are derived from the delineated geospatial datasets already described (Qiao, Nelson, et al., 2019).

The results of the model are available through a data service. The service provides forecasted and historically-simulated discharge at each of the delineated stream segments which are the computation points for the hydrologic model. The data service is a REpresentational State Transfer Application Programming Interface, or REST API. A REST API is a method for sharing data over the internet by encoding requests for data into a URL which are understood by the service provider and used to generate the data response. Our data service builds on a pattern for hydrologic model data sharing previously proposed and tested by Souffront Alcantara, et al. (2018). The original implementation also involved using a REST API that was a component of or embedded in the web-application which was developed to consume the service. That implementation presented challenges to accessing the data outside of the demonstration app. Expanding the HMaaS concept to many apps and use cases depends on a completely open and accessible data service. We improved the original implementation by separating the model execution and REST API data service from the web app which consumes the service. The model and data service are executed in a separate computing environment that is not tied to a specific use case. This allows the same data and workflow to be shared to many possible applications and enables the model results to more fully be a globally accessible and adaptable service.

2.3 Web Mapping Services and GIS Cloud Based Storage

The ability to upload, download, and graphically display the spatial data subsets are vital features of a geospatial web app. This can be done using various software or systems that provide such capabilities. These functions are supported by GeoServer, an open-source Java software that allows users to store, view and edit geospatial data, following standards set forth by the Open Geospatial Consortium (OGC). This includes the Web Map Service (WMS) protocol, which helps in the displaying of spatial data on custom built web maps (Michaelis & Ames, 2012; Michaelis, Christopher & Ames, Daniel P., 2008). GeoServer can also easily integrate with mapping APIs and traditional GIS architectures, further increasing its accessibility and versatility.

HydroShare is a cloud-based system that helps with the sharing and publishing of data and models (Tarboton et al., 2014; Yi et al., 2018). It includes extensive metadata for all data as well as the ability to permanently publish data resources with a Digital Object Identifier. Its primary targets are hydrologic scientists, researchers, and water resources, though it can be used for a variety of water data centric applications. HydroShare organizes data into user created resources. These resources can be customized to include an abstract describing the data it contains, limitations as to who can edit the resource, and how the resource can be discovered should it be made public. If GIS data is included in the resource, such as shapefiles, the resource displays a preview showing the extent of the included GIS data. Metadata to specific files can also be previewed. HydroShare.org includes an instance of a GeoServer. All spatial data that is uploaded to HydroShare is also automatically registered in the HydroShare GeoServer and made available via WMS, WFS, and WCS services. Given this readily available capability, we chose to store our GeoFabric files on HydroShare and retrieve them through the WMS into our HydroViewer apps as an efficient method for managing and displaying the spatial data because it takes advantage of third-party tools and reduces local storage space on the app server.

2.4 Web App Architecture

The GEOGloWS HydroViewer was developed using the Tethys Platform (Qiao, Li, et al., 2019; Swain et al., 2016). Built using the open-source Python web framework Django (Forcier et al., 2008), Tethys was developed with the goal of simplifying the web app development process. It can automatically construct a user database, custom settings, and several app page templates which can be tailored to display and perform the necessary functions.

The structure of a Tethys web app follows the Model-View-Controller (MVC) schema. The MVC schema is a popular way to design user interface software by separating the data and logic into three interconnected components. This software design pattern was chosen since both

Tethys and Django were built using this pattern and it also greatly simplifies the app development process. Figure 4 illustrates the implementation of MVC for the GEOGloWS HydroViewer and how the GeoFabric and its derived components work inside this schema as described in the following sections.

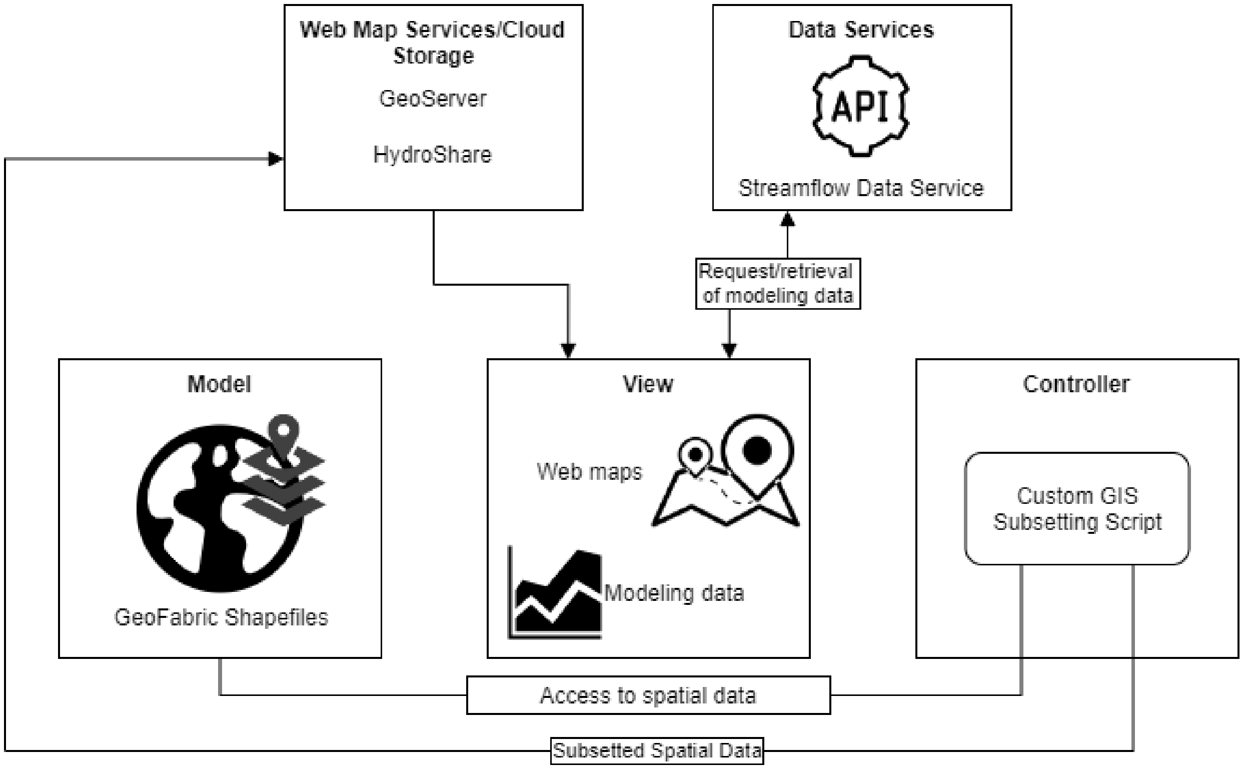


Figure 4: Overview of model-view-controller layout for the GEOGloWS HydroViewer app

2.4.1 HydroViewer Model – Interacting with the spatial data

The model portion of the MVC pattern includes all the data that are used when working with the application and how this data is stored and accessed. This data includes the GeoFabric shapefiles and the subsetted data that is stored onto HydroShare and/or GeoServer. The GeoFabric shapefiles are stored on the app server, where they are easily accessible to the rest of the app. The precise location of these files can be specified using a custom setting in the Tethys app portal.

2.4.2 HydroViewer View – Viewing the spatial and hydrologic data

The View portion of the MVC pattern deals with everything that is accessible to the user. While Tethys Platform provides the framework for the app, everything that the user interacts with is built using the three prevailing web development programming languages: Hypertext Markup Language (HTML), Cascading Style Sheets (CSS) and JavaScript (JS). All these languages work together to create a web page allowing navigation between various pages and perform certain functions in them. These programming languages can also have plugins added or access libraries that extend their functionality.

Leaflet is an open-source library that is used to build web maps. Leaflet was chosen to build the web maps contained in the GEOGloWS HydroViewer since it has a focus on performance, simplicity, and usability (Crickard III, 2014). Leaflet handles how the GeoServer WMS images are retrieved and displayed, as well as allowing the user to interact with the maps. This includes drawing custom boundaries and displaying specific map layers.

The hydrologic data that is made available via the data service is also displayed here. A trigger activated by the user makes a request to the Streamflow API. The API then identifies the location where the user is requesting data. It locates and compiles the data, then sends it back to the application. The data is unpackaged and formatted into charts, hydrographs, and other formats so the user can more easily interpret the requested data.

2.4.3 HydroViewer Controller – Transferring and requesting data

The Controller aspect is what accesses, formats, and prepares the data to control how it is viewed by the user. The bulk of this processing is accomplished using the general-purpose programming language Python. Since Tethys was built using the Python-based Django framework, Python is highly integrated into Tethys, allowing for any sort of Python functions and scripts to be performed inside the apps. These functions can also be extended or added on to using Python packages. Python packages are modules of Python code that can be called on to perform specific functions. There are numerous software repositories that contain thousands of Python packages which can help with all sorts of research and academic study.

The GeoPandas Python package was developed to help work with geospatial data in python (Jordahl, 2014). GeoPandas is an extension of the popular Python package Pandas, which is used to manipulate and run complex calculations on tabular data. GeoPandas works in much the same way, though it instead performs spatial operations on geometric data. GeoPandas can work with a large assortment of geospatial files, including ESRI shapefiles and GeoJSON. GeoPandas can access such files when they are contained inside a zip archive. This further reduces the amount of local storage required to host the app. GeoPandas can also be described as self-contained. It does not rely on external dependencies to function, such as a spatial database. The geoprocessing is accomplished within the code itself. This makes hosting or transferring the app easier, as well as reducing the effort and cost to maintain the application. The GeoPandas code can also be placed into a custom Python script and work in concert with the Python functions that Tethys possesses.

The hs_restclient package (http://hs-restclient.readthedocs.io/) and the geoserver-restconfig package (https://github.com/GeoNode/geoserver-restconfig) allow the app to access HydroShare's and a GeoServer instance's REST API. This allows the app to establish connections and transfer the subsetted data selected by GeoPandas to HydroShare or GeoServer. Using the HydroShare or GeoServer REST APIs, the subsetted data generated by the app can be uploaded and edited directly from the app itself. This allows the spatial data to have the necessary web map services generated to allow their display on web maps.

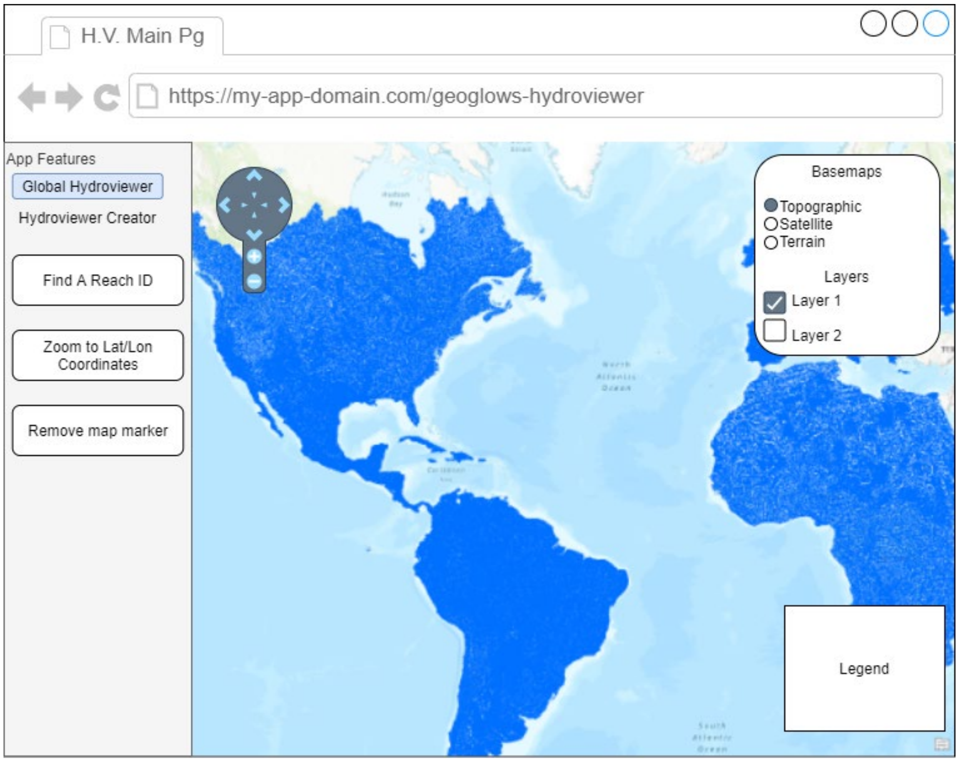
2.5 Core Functionalities

Following is a list of three functionalities of the app which are intended to demonstrate the implementation of the GEOGloWS ECMWF Streamflow Services for water management agencies and decision-makers:

1. A global HydroViewer that would:
    a. Display the global stream network
    b. Display flow data, including forecast, historical and bias correction data
    c. Search for specific streams/go to specific coordinates
    d. Change or toggle layers of the map (i.e. streams, catchment, basemap, etc.)
2. A HydroViewer project where:
    a. A preview of the custom HydroViewer is displayed
    b. Data can be uploaded to Hydroshare or GeoServer
    c. The data subsets can be downloaded in their original format for use in other applications
    d. The created HydroViewer can be exported in a format that would be easy to share and customize
3. A custom HydroViewer that:
    a. Can be further customized using the subsetted data
    b. Can be easily stored and viewed on various systems/platforms
    c. Displays flow data like in the global HydroViewer

2.5.1 Global HydroViewer layout

The global HydroViewer is where the global streamflow network can be viewed in its entirety (Figure 5). Streamflow data, including forecasts and historical data, can be viewed by interacting with the displayed spatial data. The option to search for a specific stream or location using stream IDs and coordinates is included. We also added a feature where only logged in users with the correct credentials would be able to access the customization portion of the web app. This separated the functionality of a standalone global HydroViewer from the tools needed to subset and create regional/local HydroViewers so that it could not be accidentally corrupted.



● Topographic
○ Satellite
○ Terrain

☑ Layer 1
☐ Layer 2

Figure 5: Layout of the main page of the web app that includes the global hydroviewer

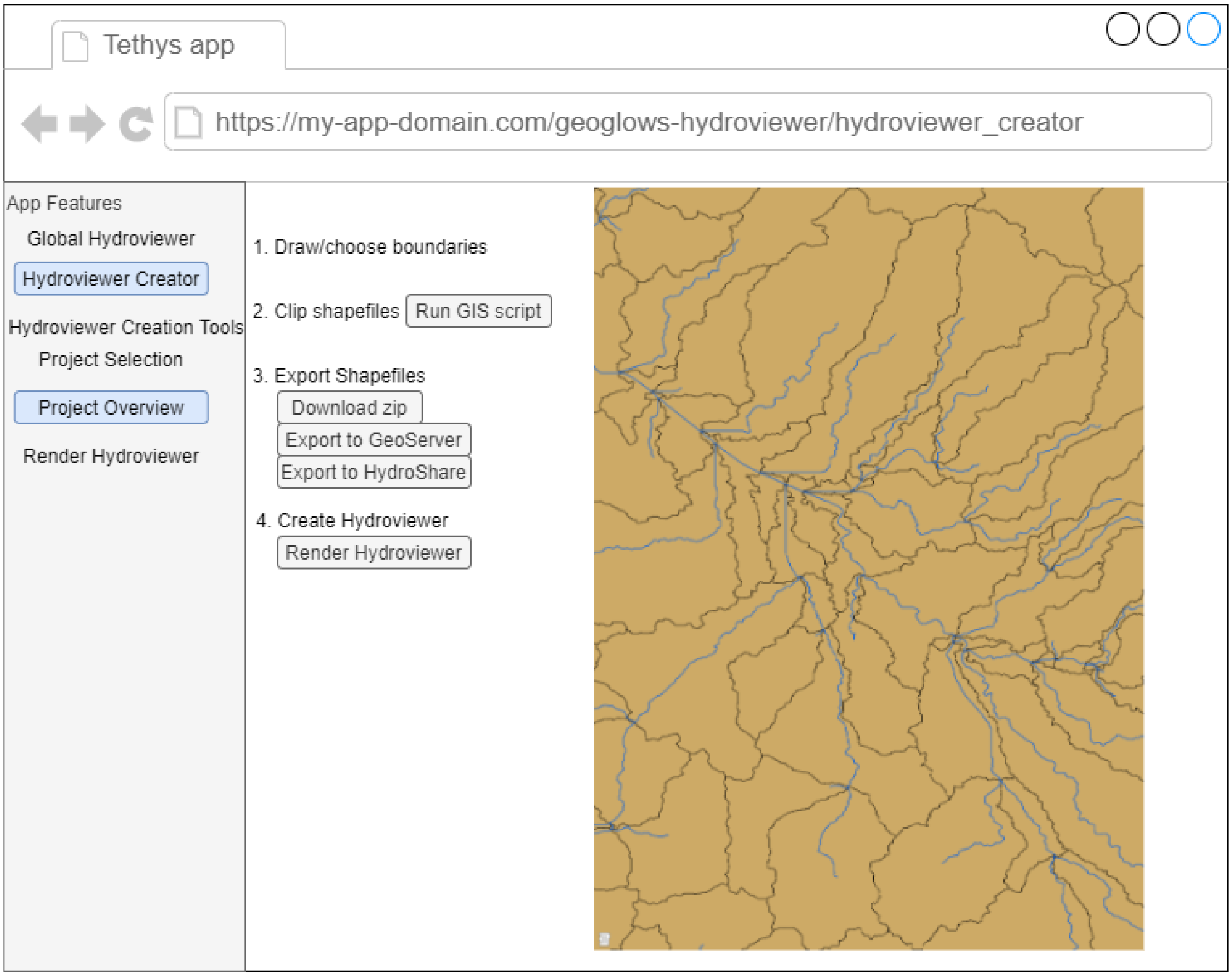
2.5.2 HydroViewer Project

The HydroViewer Creator is managed through three “views” or web pages. First is the Project Selection page. By filling out the “New project name” field and clicking the “Create new Hydroviewer” button, a new folder with the same project name is created inside the app’s directory. This folder will contain all the files necessary to create and run the Hydroviewer, such as configuration settings and shapefile data. Editing existing projects will be managed by selecting one of the listed projects in the “Existing project” drop down menu and clicking the “Edit Hydroviewer” button. This displays the Project Overview page of the HydroViewer Creator, where the details of the selected project are displayed. The drop-down menu only lists the projects that have a folder in the app directory. By selecting a project in the “Delete project” field and clicking the “Delete Hydroviewer” button, this removes the selected project from the app directory permanently.

The Project Overview page (Figure 6) is where the user is able to run the processes necessary to create the delineation data subsets for their own HydroViewer. These processes include:

1. Draw or select boundaries for the project’s area of interest
2. Run a custom script that will subset shapefile data
3. Export shapefiles to a local zip archive, GeoServer, or HydroShare resource
4. Render an HTML file for the HydroViewer

As some of the options listed above require preceding processes to be completed, some options will not be active until those necessary processes are completed. For example, in order to export the shapefiles, boundaries for the project need to be created and the subsetting script run. The Project Overview will also include a preview map that will display the created data when the WMS data service is generated.



1. Draw/choose boundaries

2. Clip shapefiles

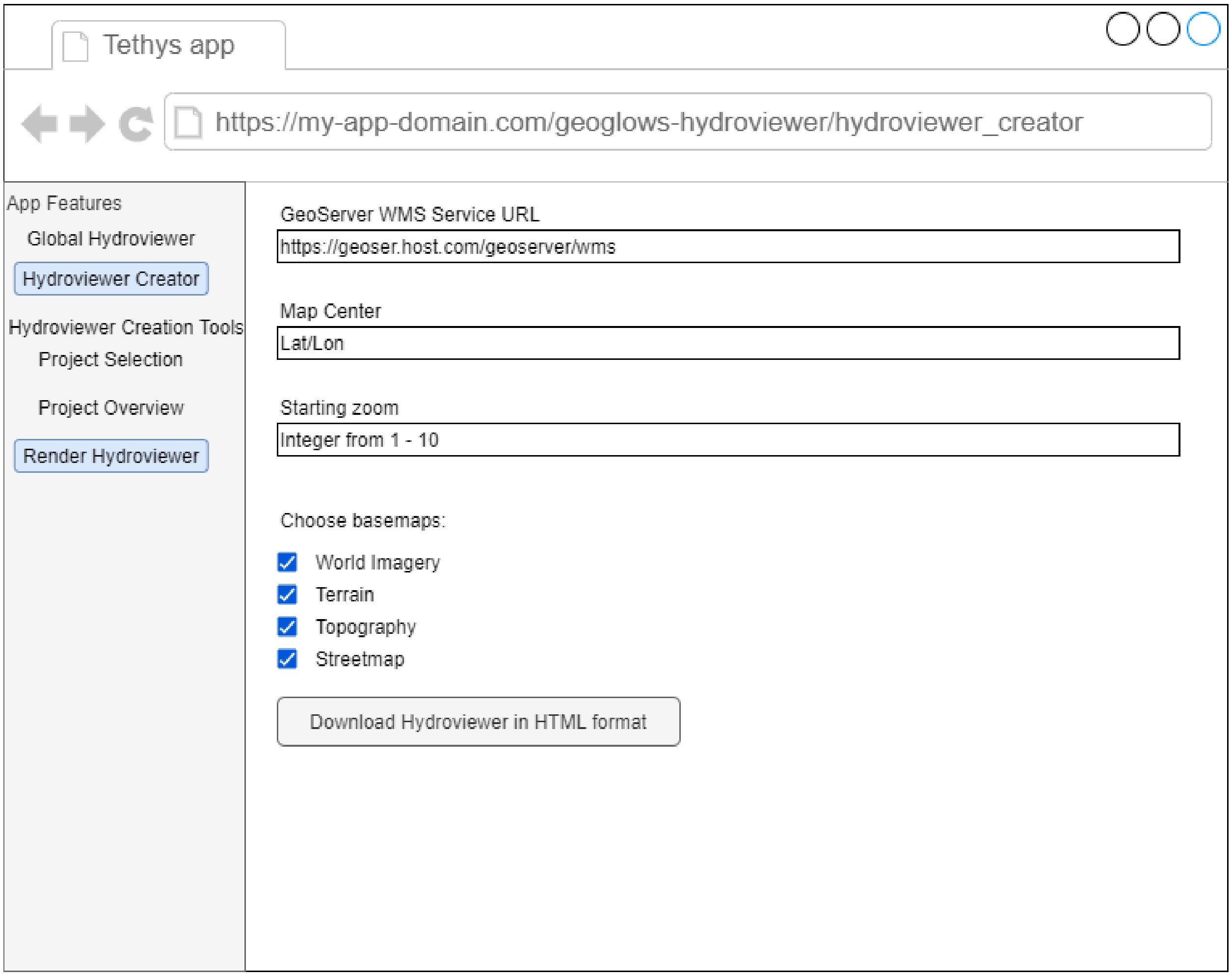
3. Export Shapefiles

4. Create Hydroviewer

Figure 6: Layout of the Project Overview page

2.5.3 Custom HydroViewer

The final page of the HydroViewer Creator is the Render HydroViewer page (Figure 7). Here the user will be able to create and download a HTML file that will act as the HydroViewer itself. The fields on this page will have default inputs filled in based off previous inputs provided by the user and which project the user is working in. These values can be changed to create new HTML HydroViewers according to the options the user specifies. These options are either added or reflected in a HTML template stored inside the project directory. When ready, the user can select the option to download the HTML file to the device they are using to access the app.


Tethys app
https://my-app-domain.com/geoglows-hydroviewer/hydroviewer_creator
App Features
Global Hydroviewer
Hydroviewer Creator
Hydroviewer Creation Tools
Project Selection
Project Overview
Render Hydroviewer
Download Hydroviewer in HTML format

GeoServer WMS Service URL: https://geoser.host.com/geoserver/wms

Map Center: Lat/Lon

Starting zoom: Integer from 1 - 10

Choose basemaps:

- ☑ World Imagery
- ☑ Terrain
- ☑ Topography
- ☑ Streetmap

Figure 7: Layout of the Render HydroViewer page

3 RESULTS

3.1 Software Implementation

The current version of the GEOGloWS HydroViewer can be viewed on BYU's Tethys portal (https://tethys.byu.edu/). However, administrative privileges are required to access the HydroViewer Creator of the app. You can also download the source code from GitHub (https://github.com/BYU-Hydroinformatics/geoglows_hydroviewer) and install it using Tethys (http://docs.tethysplatform.org/en/stable/). When installed through Tethys, the app can be viewed in its entirety.

3.2 Global, preview, and custom HydroViewers

The global HydroViewer included within the GEOGloWS HydroViewer displays the global stream network. There are two display options available. The first is provided by a service through Esri. This service changes the display of the streams depending on how much flow is running through them. A stream with a greater amount of flow is thicker than streams with less flow. The Esri display also adjusts the scale of river reaches displayed depending on the zoom level of the map. At the default zoom level, only streams with a significant amount of flow are displayed. As the zoom level increases (zooming into the map), the scale for river reaches changes, displaying the smaller reaches that make up the stream network. The service also animates the stream display, showing past, present, and future changes to stream flow. These changes are represented by changing the line thickness. Extreme flows are represented by changing the color of the stream. This dynamic display helps with the loading time for the map and effectively relays important information in an intuitive manner. However, there are cases where the Esri service might not be able to display the stream network as intended. In these instances, the display can be changed to use the raw shapefile data that is part of the GeoFabric. This displays the spatial as is without any modifications or extra visuals.

The global HydroViewer can display streamflow data from any selected river (Figure 8). By clicking on the map, the app identifies the stream selected by its unique identifier. Using this identifier, a request is made to the Streamflow API. The API locates the data associated with the identified stream and sends it back to the app to be displayed.

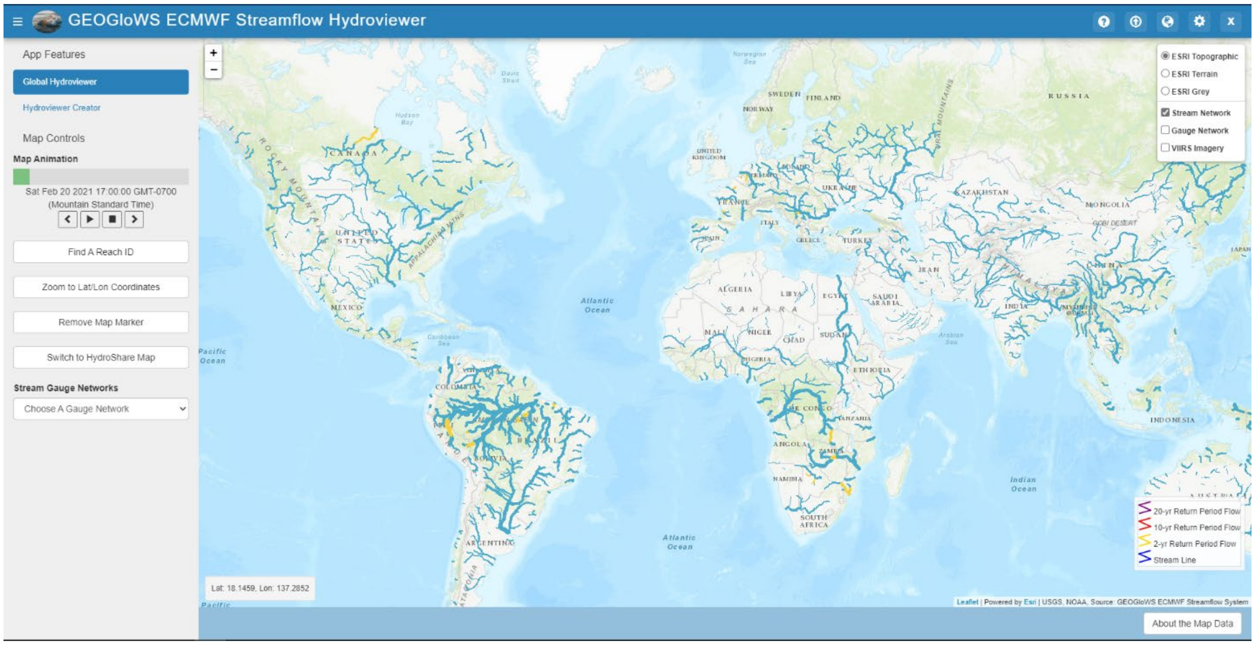
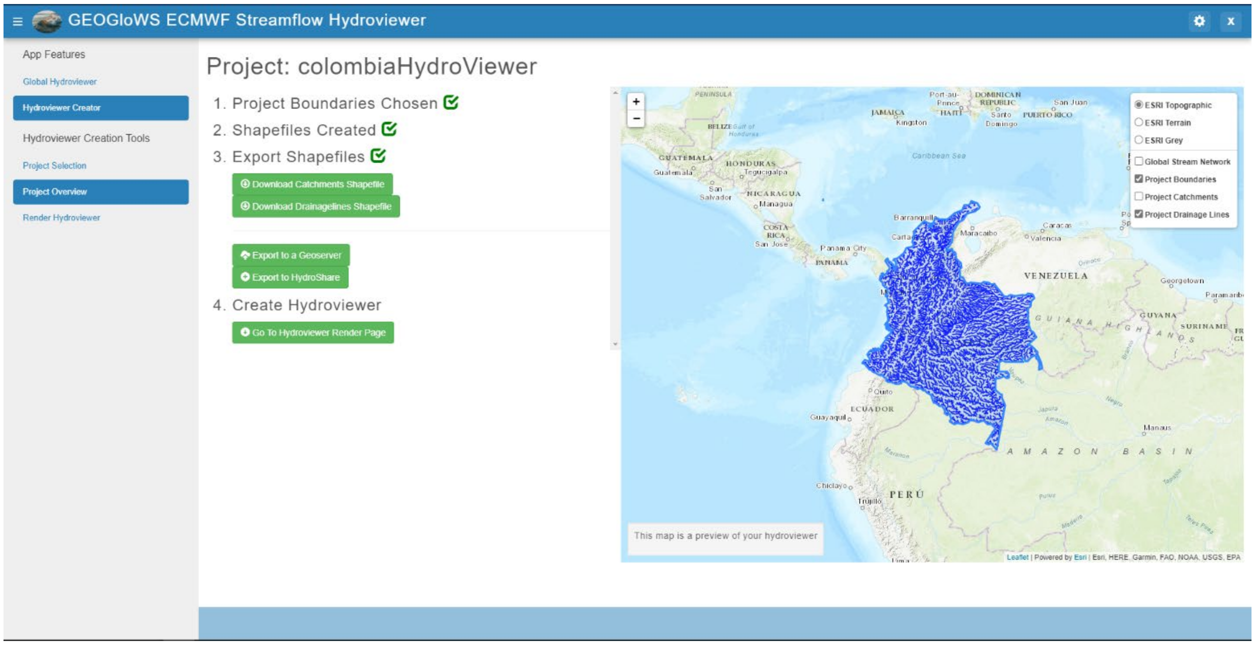


Figure 8: Global HydroViewer page of the web app

The HydroViewer Creator portion of the web app can subset the GeoFabric spatial data using either custom defined boundaries or a country's political boundaries as shown in Figure 9. The subsetted data can then be uploaded to HydroShare or GeoServer to generate the WMS service for the created data or directly downloaded to a local machine. The motivation for providing the data to be accessed or saved in these formats is so local organizations or agencies can use or analyze the data according to their needs. They can use the data in other applications or compare it to similar data. Being able to work with the data on this level helps to increase their familiarity with it, which could help to increase their confidence in using it for other applications and analyses.



GEOGloWS ECMWF Streamflow Hydroviewer

Project: colombiaHydroViewer

1. Project Boundaries Chosen ☑
2. Shapefiles Created ☑
3. Export Shapefiles ☑
4. Create Hydroviewer

● ESRI Topographic
○ ESRI Terrain
○ ESRI Grey
□ Global Stream Network
☑ Project Boundaries
□ Project Catchments
☑ Project Drainage Lines

This map is a preview of your hydroviewer

Figure 9: Recreated Somalia HydroViewer using GEOGloWS HydroViewer app

After uploading the selected spatial data to either GeoServer or HydroShare an HTML file can be exported from the app that shows the subsetted data. This HTML HydroViewer can display the same modelling data included in the global HydroViewer by linking it to the data service that accesses the streamflow data and the WMS service generated by HydroShare or a custom GeoServer for the spatial data. Since the HTML HydroViewer accesses this data via third-party databases, the files that make up these HydroViewers are lightweight in terms of storage and can be easily shared, uploaded, and hosted. This HTML file acts as a base for which further customizations and information can be applied. Other data services can be added to the HTML file, further extending its applicability. How the spatial data is displayed can also be edited, which could reflect the research or purpose of the HydroViewer. Figure 10 displays two instances of a HydroViewer for Colombia. The first is a basic HTML HydroViewer generated by the GEOGloWS HydroViewer. It displays the catchments and streams from the GeoFabric that are within Colombia and has access to the Streamflow API and the data it contains. The second is a customized HydroViewer that was created previously. It uses the same spatial data as contained in the GeoFabric while also displaying other relevant data. These figures illustrate that not only can the data extracted by the web application be used for other practical purposes in hydrologic studies but that the GEOGloWS HydroViewer app can greatly facilitate the creation of custom HydroViewers by automating much of the process.

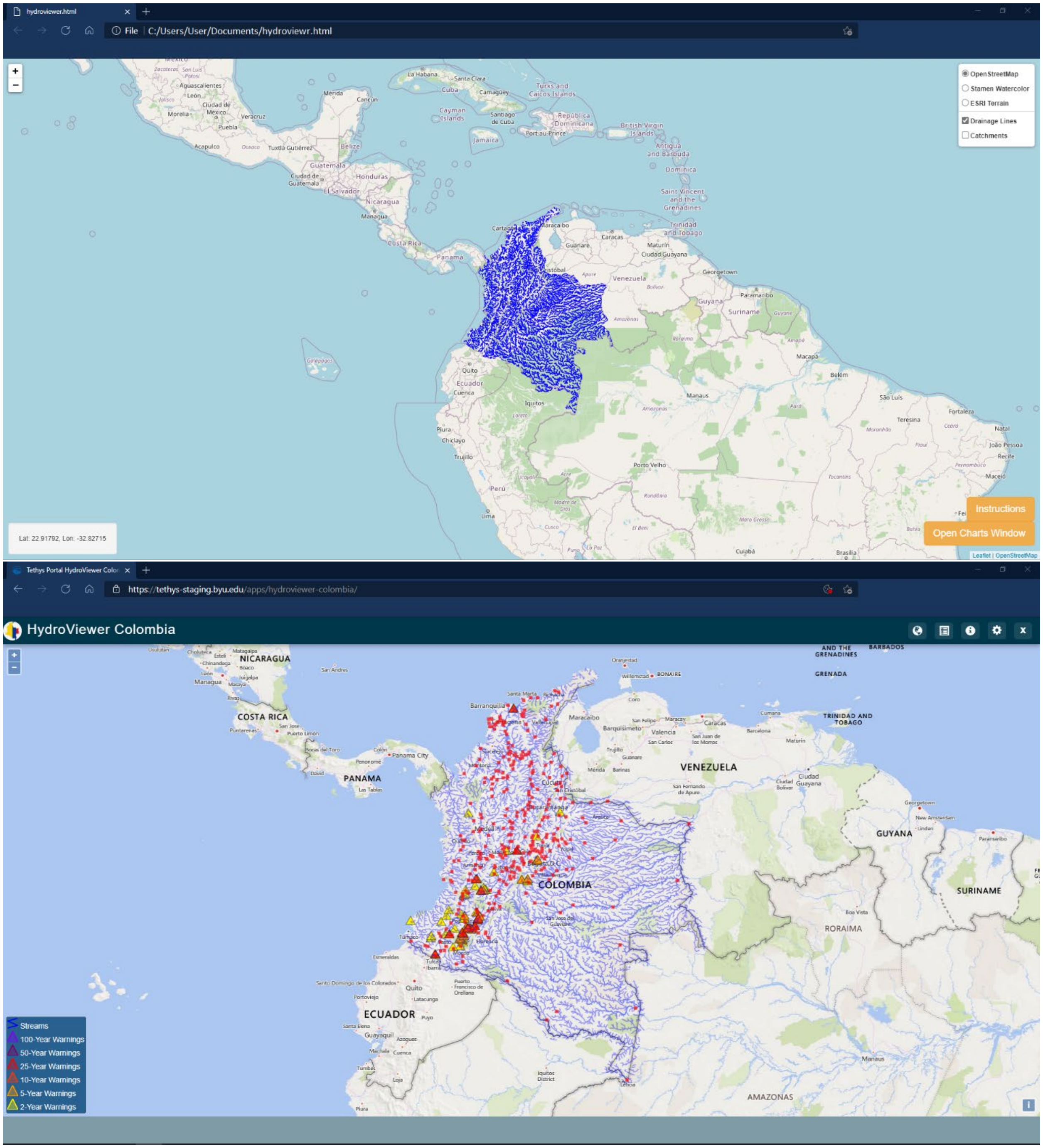


Figure 10: Simple HTML HydroViewer for Colombia (top) compared to custom HydroViewer for same area (bottom)

4 CONCLUSION

This research was to develop a system through which global hydrologic data can be shared in various formats or displayed in a customizable manner. A new global network of streams and catchments, referred to as a GeoFabric, was developed so that the features of said GeoFabric could be used in other software and better represent regions that previously had lacked appropriate spatial and modelling data. Various software and services were implemented to create a web application that was built using the Tethys web app framework. The app can display and subset the spatial data. It also displays streamflow data via a data service. The application is also capable of creating custom HydroViewers that display selected spatial data and is linked to the data service that provides the streamflow data. These HydroViewers can be customized further to include other information relevant to the research or needs of the agency that created it.

REFERENCES

Balsamo, G., Beljaars, A., Scipal, K., Viterbo, P., van den Hurk, B., Hirschi, M., & Betts, A. K. (2009). A Revised Hydrology for the ECMWF Model: Verification from Field Site to Terrestrial Water Storage and Impact in the Integrated Forecast System. *Journal of Hydrometeorology*, *10*(3), 623–643. https://doi.org/10.1175/2008JHM1068.1

Crickard III, P. (2014). *Leaflet. Js essentials*. Packt Publishing Ltd.

David, C. H., Maidment, D. R., & Yang, Z. (2008). Routing Application for Parallel computatIon of Discharge. *AGU Fall Meeting Abstracts*, *51*, H51H-0967.

Derived Hydrography of World Regions. (n.d.). [Data set]. https://doi.org/10.4211/hs.d5155cb57987489a95b83364d2c0f6c0

Fatichi, S., Vivoni, E. R., Ogden, F. L., Ivanov, V. Y., Mirus, B., Gochis, D., Downer, C. W., Camporese, M., Davison, J. H., Ebel, B., Jones, N., Kim, J., Mascaro, G., Niswonger, R., Restrepo, P., Rigon, R., Shen, C., Sulis, M., & Tarboton, D. (2016). An overview of current applications, challenges, and future trends in distributed process-based models in hydrology. *Journal of Hydrology*, *537*, 45–60. https://doi.org/10.1016/j.jhydrol.2016.03.026

Forcier, J., Bissex, P., & Chun, W. J. (2008). *Python web development with Django*. Addison-Wesley Professional.

Jordahl, K. (2014). *GeoPandas: Python tools for geographic data* (0.8.2) [Python]. https://github. com/geopandas/geopandas

Lehner, B., Verdin, K., & Jarvis, A. (2008). New Global Hydrography Derived From Spaceborne Elevation Data. *Eos, Transactions American Geophysical Union*, *89*(10), 93–94. https://doi.org/10.1029/2008EO100001

Li, Z. (2014). Watershed modeling using arc hydro based on DEMs: A case study in Jackpine watershed. *Environmental Systems Research*, *3*(1), 11. https://doi.org/10.1186/2193-2697-3-11

Michaelis, C. D., & Ames, D. P. (2012). *Considerations for Implementing OGC WMS and WFS Specifications in a Desktop GIS*. *2012*. https://doi.org/10.4236/jgis.2012.42021

Michaelis, Christopher & Ames, Daniel P. (2008). Web Mapping Service (WMS) Web Feature Service (WFS) Web Processing Service (WPS). In *Encyclopedia of GIS* (pp. 1259–1261). Springer.

Piesse, M. (2020). *Global water supply and demand trends point towards rising water insecurity* [Report]. Future Directions International. https://apo.org.au/node/276976

Qiao, X., Li, Z., Ames, D. P., Nelson, E. J., & Swain, N. R. (2019). Simplifying the deployment of OGC web processing services (WPS) for environmental modelling–Introducing Tethys WPS Server. In *Environmental modelling & software* (Vol. 115, pp. 38–50).

Qiao, X., Nelson, E. J., Ames, D. P., Li, Z., David, C. H., Williams, G. P., Roberts, W., Lozano, J. L. S., Edwards, C., & Souffront, M. (2019). A systems approach to routing global gridded runoff through local high-resolution stream networks for flood early warning systems. *Environmental Modelling & Software*, *120*, 104501.

Souffront Alcantara, M. A., Kesler, C., Stealey, M. J., Nelson, E. J., Ames, D. P., & Jones, N. L. (2018). Cyberinfrastructure and Web Apps for Managing and Disseminating the National Water Model. *JAWRA Journal of the American Water Resources Association*, *54*(4), 859–871. https://doi.org/10.1111/1752-1688.12608

Souffront Alcantara, M. A., Nelson, E. J., Shakya, K., Edwards, C., Roberts, W., Krewson, C., Ames, D. P., Jones, N. L., & Gutierrez, A. (2019). Hydrologic Modeling as a Service (HMaaS): A New Approach to Address Hydroinformatic Challenges in Developing Countries. *Frontiers in Environmental Science*, *7*. https://doi.org/10.3389/fenvs.2019.00158

Swain, N. R., Christensen, S. D., Snow, A. D., Dolder, H., Espinoza-Dávalos, G., Goharian, E., Jones, N. L., Nelson, E. J., Ames, D. P., & Burian, S. J. (2016). A new open source platform for lowering the barrier for environmental web app development. *Environmental Modelling & Software*, *85*, 11–26. https://doi.org/10.1016/j.envsoft.2016.08.003

Tarboton, D. G., Idaszak, R., Horsburgh, J. S., Heard, J., Ames, D., Goodall, J. L., Band, L., Merwade, V., Couch, A., Arrigo, J., & others. (2014, June 15). HydroShare: Advancing collaboration through hydrologic data and model sharing. *Proceedings of the 7th International Congress on Environmental Modelling and Software*.

Vörösmarty, C. J., Green, P., Salisbury, J., & Lammers, R. B. (2000). Global Water Resources: Vulnerability from Climate Change and Population Growth. *Science*, *289*(5477), 284–288. https://doi.org/10.1126/science.289.5477.284

Wu, H., Adler, R. F., Hong, Y., Tian, Y., & Policelli, F. (2012). Evaluation of Global Flood Detection Using Satellite-Based Rainfall and a Hydrologic Model. *Journal of Hydrometeorology*, *13*(4), 1268–1284. https://doi.org/10.1175/JHM-D-11-087.1

Yamazaki, D., Ikeshima, D., Sosa, J., Bates, P. D., Allen, G. H., & Pavelsky, T. M. (2019). MERIT Hydro: A High-Resolution Global Hydrography Map Based on Latest Topography Dataset. *Water Resources Research*, *55*(6), 5053–5073. https://doi.org/10.1029/2019WR024873

Yamazaki, D., Ikeshima, D., Tawatari, R., Yamaguchi, T., O'Loughlin, F., Neal, J. C., Sampson, C. C., Kanae, S., & Bates, P. D. (2017). A high-accuracy map of global terrain elevations. *Geophysical Research Letters*, *44*(11), 5844–5853. https://doi.org/10.1002/2017GL072874

Yi, H., Idaszak, R., Stealey, M., Calloway, C., Couch, A. L., & Tarboton, D. G. (2018). Advancing distributed data management for the HydroShare hydrologic information system. *Environmental Modelling & Software*, *102*, 233–240. https://doi.org/10.1016/j.envsoft.2017.12.008