GABBs - Reusable Geospatial Data Analysis Building Blocks for Science Gateways

Lan Zhao, Carol X. Song, Rajesh Kalyanam, Larry Biehl, Robert Campbell, Leif Delgass, Derrick Kearney, Wei Wan*, Jaewoo Shin, I Luk Kim, Carolyn Ellis
Rosen Center for Advanced Computing (*: visiting scholar)
Purdue University, West Lafayette, IN 47906, U.S.A.
{lanzhao, cxsong}@purdue.edu

***Abstract*—Science gateways have gained wide adoption in recent years as an effective platform for a lower barrier entry to computational resources, research collaboration, dissemination of scientific data, applications and publications, online education, and community engagement. Although multiple portal frameworks and middleware toolkits exist to facilitate the development of a science gateway, the task of bringing data and tools online into a science gateway environment is still daunting for domain science users. In this paper, we describe GABBs, a National Science Foundation funded project that aims to reduce this obstacle by delivering reusable software building blocks for geospatial data management and analysis based on the HUBzero portal platform. The main components of GABBs include a geospatial data management system named *iData*, libraries for easy creation of geospatial data analysis tools hosted in the gateway, *GeoBuilder* for creating GIS-enabled data exploration tools without programming, and general purpose tools for geospatial data processing and visualization. GABBs also provides the software for linking these components/functions into dynamic workflow pipelines. The open source GABBs software has been deployed on MyGeoHub and utilized in several domain applications.**

***Keywords*—GABBs; DIBBS; science gateway; HUBzero; building blocks; geospatial data**

I. Introduction

The term *science gateway* often refers to a web-based system that provides integrated access to data, applications and tools targeted for a specific science community. In the past decade, science gateways have gained wide adoption as an effective platform for easy access to computational resources, research collaboration, dissemination of scientific data, applications and publications, online education and training, and community engagement. Many gateways have been developed across various science and engineering disciplines, such as CyVerse (www.cyverse.org) for life science research, HydroShare [1] for hydrologic data and model resource management, CyberGIS [2] for geospatial data analysis and modeling using HPC resources, and nanoHUB (nanohub.org) for nanotechnology education and research through user contributed online simulation tools and hosted education materials, to name a few.

The development of a science gateway requires a broad spectrum of knowledge and expertise, including web development for the front-end interface, system operations for deployment and execution of applications on HPC resources, middleware and cyber security technology, as well as an understanding of the science in order to work with the domain data and applications specific to the gateway. To facilitate science gateway development, a number of toolkits and portal frameworks have been created, providing out-of-box middleware and portal infrastructure for common gateway functions. Some popular examples are Drupal (www.drupal.org), Django (www.djangoproject.com), HUBzero (hubzero.org), Galaxy (galaxyproject.org), Spring (spring.io), LifeRay (liferay.com), Globus (globus.org), Agave (agaveapi.co), and Apache Airavata (airavata.apache.org).

Among them, HUBzero provides an open source platform for creating dynamic web portals (hubs) to support research, education, and outreach activities for scientific communities. It includes a number of ready-to-use functions for scientific collaboration, including project groups, wiki, forum, tagging, reviews, citations, Q&A, wish list, and a ticketing system for user support. Communities can develop, contribute, and share scientific tools online which can be launched on local or national HPC resources such as XSEDE through HUBzero's *submit* mechanism. Hub users can execute desktop tools securely in a remote virtual container and interact with the tool's graphical user interface in their web browsers, via virtual network computing (VNC). HUBzero also provides the RAPPTURE Toolkit to aid rapid tool development. RAPPTURE essentially web-enables desktop applications without web programming, hence, allowing scientists (mostly not expert web developers), to put graphical user interfaces in their scientific applications and make them accessible on the web, accelerating the deployment of new tools. HUBzero has been used to power more than 60 gateways for scientific domains ranging from hydrology, earth science, pharmacy, cancer care engineering, advanced manufacturing, study of human-animal bonding, to research data publication and engineering education, among others.

While the HUBZero framework satisfies the basic functionality of collaboration and networking among researchers, some domains have identified a need for fundamental and yet highly interactive tools for handling geospatial datasets, mapping, and modeling using high performance computing resources. Adding such capabilities in HUBzero would require significant expertise in GIS, visualization and system administration. Furthermore, although HUBzero provides simple data sharing functions via Hub Project, it lacked support for large scale scientific datasets, especially geospatial data which often have heterogeneous formats, are multi-dimensional, and come with rich metadata. The GABBs (Geospatial Data Analysis Building Blocks) project was conceived to address these needs. Funded by the NSF DIBBs program, GABBs is aimed at giving users the ability to easily manage data and create/share online geospatial data analysis tools by themselves.

Built on top of HUBzero, GABBs consists of reusable software modules enabling easy-to-use geospatial data management, exploration, visualization, and tool development capabilities, for users with different levels of expertise. In the following sections, we will first describe GABBs' design and implementation and discuss its main components and capability for creating data-driven workflows. We will then describe several use cases. In Section IV we will describe the packaging and dissemination of GABBs. We will discuss the broader impact and conclude the paper in section V.

II. GABBs Design and Implementation

The overall goal of GABBs is to enable non-expert science users to self-manage their geospatial data, bring their data analysis tools online, and construct workflows connecting the data space and tool space on the HUBzero platform. GABBs extends the HUBzero core capabilities to provide out-of-box support for (1) scientific data management with value added services for geospatial data such as preview, automatic metadata extraction, and map based search, (2) creating map-enabled geospatial data driven tools using RAPPTURE and other common programming languages, and (3) launching tools directly from the data browsing interface and to programmatically save tool output back to the data management system.

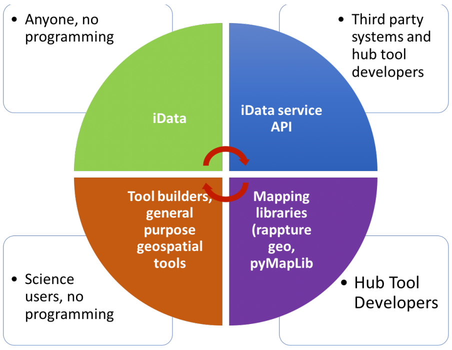

Anyone, no programming
Third party systems and hub tool developers
iData
iData service API
Tool builders, general purpose geospatial tools
Mapping libraries (rappture geo, pyMapLib
Science users, no programming
Hub Tool Developers

Fig. 1. GABBs components and multiple entry points for different user groups

Instead of providing one hard-wired comprehensive solution, GABBs was implemented with multiple entry points to serve different needs of the community. As shown in Fig. 1, GABBs provides *iData* for end-to-end data management with an easy-to-use web interface, *GeoBuilder* for programing-free tool creation, and general purpose geospatial data analysis tools such as *MultiSpec*, all of which are ready-to-use by the end users. On the other side, science users with some programming background will find it handy to use the drop-in map widgets, map libraries, and toolkits provided by GABBs in tool development. Finally, skilled application developers can invoke the GABBs data service APIs to connect their applications with iData. The design and implementation of these components is shown in Fig. 2 and will be discussed in the following sections.

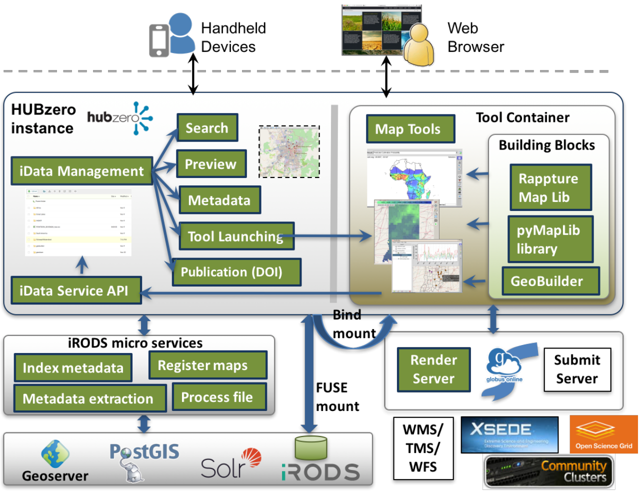


Fig. 2. GABBs system design (The green blocks represent software developed in GABBs.)

A. iData for Geospatial Data Management

Most common sources of geospatial data encode useful metadata in the data files. Geospatial data is also distinct in that useful information can often be gleaned from visualizing geospatial files or overlaying multiple such files. For instance, a data file containing a matrix of crop yields over a certain region is more intuitive to decipher and interpret when overlaid on a map of that region or other files containing land use and population maps. As shown in Fig. 2, iData is a data management system for hub projects that natively provides capabilities such as automatic metadata capture and preview that better serve structured, visualizable geospatial data. Scientific data is often copious in quantity and large in size. Both metadata extraction and geospatial preview require non-trivial resource intensive processing. Additionally, larger files are not ideal for web-based uploads and are typically transferred via mechanisms such as Globus transfer or SFTP. In view of these considerations and to make file processing agnostic to the ingestion method, such processing needs to be attached to the storage resource rather than on the hub side. These factors and the ease of expandability led to our choice of iRODS as the data management framework underlying iData. While iRODS has several client APIs, the iRODS FUSE client was used to mount iRODS files to the hub webserver's local filesystem, reducing file transfer overhead. iRODS supports pluggable functions termed "microservices" that can automatically run on various file events such as creation, rename and delete. This capability is exploited to attach a metadata extraction microservice to run automatically when a new file is uploaded. The GDAL geospatial library is used to process the uploaded file and extract as much metadata as possible. A separate metadata indexing microservice is employed to index this extracted metadata into the Apache Solr service used by HUBzero to support search for various hub resources. In addition to automatic execution, microservices can also be run on-demand. On-demand execution of a geospatial preview microservice is used to pre-process geospatial files for registration in Geoserver when previews are requested from the iData web interface.

The hub web interface isn't the only possible data ingestion or access method. In order to support access to all the iData capabilities from third party applications, an iData REST API is provided. Due to widespread use of HUBzero in scientific research, there is built-in support for publishing hub project files with an associated DOI (Digital Object Identifier). This functionality is extended to include iData files and all captured metadata is serialized and added to the publication.

B. *Software for Geospatial Tool Development*

Despite the availability of popular mapping libraries for web applications such as OpenLayers and Leaflet, the existing mapping libraries for hub tools (which are desktop based running inside an OpenVZ container) are either very complex and require expert knowledge (such as GRASS, QGIS), or difficult to set up in a Linux operating system (such as ArcPy). To fill that gap, GABBs provides two mapping libraries to help researchers create map-enabled geospatial tools in the hub workspace using both hardware and software based rendering techniques.

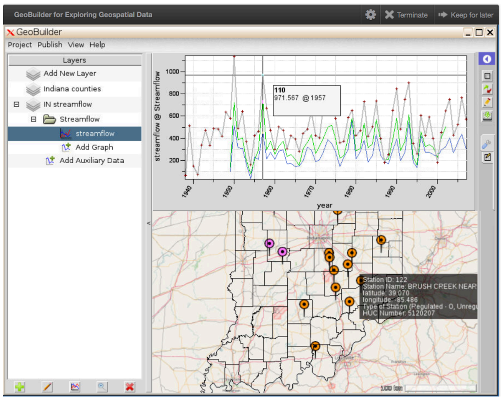
The hardware based rendering solution consists of an extension of the RAPPTURE Toolkit library. It includes new object types (e.g., maps, shapefiles, raster data) as well as controls (e.g., pan, zoom), views (e.g., extent) and visualization modes (e.g., line graph, bar chart), and a GeoVis render server which runs on a render server node and serves rendered images to a RAPPTURE-based map viewer client. The GeoVis render server performs GPU-accelerated OpenGL rendering using OpenSceneGraph (openscenegraph.org) and OSGEarth (osgearth.org). The RAPPTURE map viewer client communicates with the render server using a custom Tcl language based protocol.

For tools developed using Python, Java, and C, a general purpose, open source Python map library called PyMapLib [4] was developed for rapid integration of geospatial data and interactive visualization with research applications. It enables users to import various types of spatial data onto a base map, edit the data, perform spatial data analysis, visualize results, and share the tool with others. Built on top of open source GIS and visualization libraries including pyQGIS, GDAL, Proj4, pyQT, and matplotlib, PyMapLib consists of a set of simple Python APIs which abstract and wrap pyQGIS functions, making it easy for users to create basic map objects programmatically. PyMapLib also provides a set of map tools that support common interactive map operations. These map tools and APIs were used to create a highly configurable, generic map widget that can be imported to a Python program or embedded on-the-fly in tools written in Java and C++ with minimal programming. Users can change the layout and toolbar options in the map widget by setting properties of the map container object programmatically. PyMapLib is available on GitHub (github.com/waneric/PyMapLib).

C. *Builder and General Purpose Geospatial Tools*

In addition to mapping libraries and APIs which require a certain level of programming expertise to use, GABBs also provides general purpose tool builders and geospatial exploration tools which require no programming.

As data sharing and dissemination becomes more and more important for research and collaboration, there is an increasing need to help individual researchers to bring their data online in an interactive format instead of as a simple file download (e.g., a tar file). In the case of geospatial data, a GIS-enabled map interface is highly desired but often hard to develop by scientists without web programming skills. The GeoBuilder tool fills this gap by enabling users to explore and share their geospatial data through an interactive, map-based interface without having to develop code themselves. It makes direct use of the RAPPTURE map object, map viewer widget, and mapping API. Users follow a step-by-step guided interface to load and configure their geospatial data. As shown in Fig. 3, multiple map layers can be overlaid to facilitate information correlation. For geo-referenced spreadsheets, a user can filter the data by queries and have the data automatically selected on the map. For data that comes with associated time series measurements, a user can configure the tool to plot the variables of interest when a set of markers are selected. A user can either explore the data interactively during this process, or save the configured data view and share it with collaborators or the public via a URL. The latter function makes it very easy for data owners to share their data online with a GIS-enabled interface in a matter of minutes, which used to take weeks or months if a stand-alone web application is to be developed.



Station ID: 122
Station Name: BRUSH CREEK NEAR
latitude: 39.070
longitude: -85.486
Type of Station (Regulated - O, Unregu
HUC Number: 5120207

Fig. 3. GeoBuilder showing an overlay of two geospatial datasets and interactive data exploration

As an example of general purpose geospatial tool, MultiSpec [5] is a freeware data analysis software system developed for interactively analyzing Earth observational multispectral and hyperspectral image data from airborne and spaceborne systems, as well as a number of other types of multispectral image data. MultiSpec was integrated with the iData management system to provide users with a more seamless experience (Fig. 4). Some of the features available in MultiSpec include the ability to import many formats of image data (e.g., GeoTIFF, HDF4, HDF5, netcdf, GRIBS, jpeg2000), perform unsupervised and supervised classifications, overlay shapefiles, create transformations of images such as vegetation index or principal component images, and display histograms and line graphs of the data values.

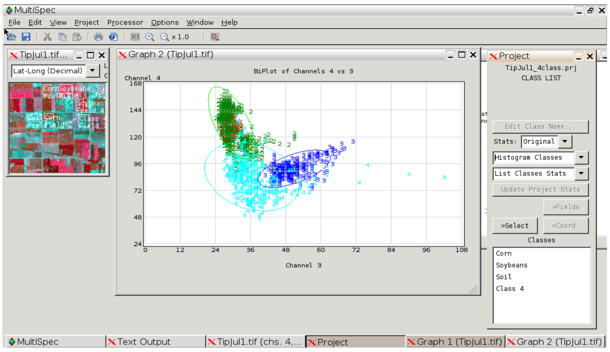


Fig. 4. MultiSpec Online tool with a false color image of a portion of a Landsat scene and a biplot of the training data for four classes

D. Data Driven Workflows

Besides enabling users to self-manage geospatial datasets and create online data analysis tools, GABBs goes one step further in providing the underlying infrastructure that allows users to link their data and analysis dynamically into workflows, via integration between iData and hub tools. The goal of this integration is to allow users to seamlessly manage their research data in iData, utilize them in hub tools, generate and save tool outputs back to iData and exploit the value-added services such as metadata capture and annotation and geospatial previews. Recall that the iRODS managed hub project files presented by the iData interface are physically accessed via a FUSE mount on the hub's webserver. This enables these files to be mounted into tool containers running on the webserver via a bind mount. A bind mount is necessary to preserve access control, only allowing the tool user to access hub project files that they can get to on the web interface. Subsequently, hub project files appear as local files in tool sessions, allowing tools to read and write to them just like they would any other local files. As a result, iData can function as the tool's input source and output destination. By locating the metadata extraction processing of iData files at the iRODS server, such processing is agnostic to whether these files were uploaded via the iData web interface or created in tool sessions. To further simplify tool discovery, iData allows tools that require a single file input to be automatically launched from its file-browsing web interface by providing a drop-down list of such launch-able tools (if any) for each file. Such tool-file associations can be registered by tool developers and created automatically once approved by a hub administrator.

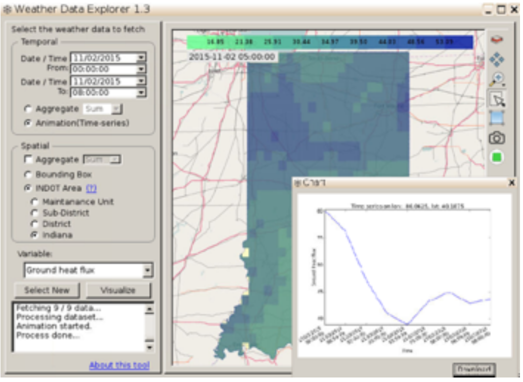
In effect, complex workflows can be constructed that start from data entry in iData, processing in hub tools or using HPC resources, results saved back to iData and subsequent use in other hub tools without the user ever having to worry about the data transports between tools or to and from iData. Moreover, files created at any step in this process and saved to iData have the same access to all value-added services such as metadata extraction and preview.

III. Use Cases

The complete set of GABBs software has been deployed on an existing hub called MyGeoHub (mygeohub.org). In addition, some of the GABBs components have been deployed in other hubs upon request, for example, PyMapLib on smarteragriculture.org, and the iData data management infrastructure on MATerials Innovation Network (https://matin.gatech.edu/). MyGeoHub hosts several projects with common geospatial data analysis needs and serves as a platform for early users to try GABBs software and provide feedback. Several tools and applications have been developed using GABBs software and deployed on MyGeoHub. In addition, users have used iData as a central data repository that connects multiple applications in a workflow. Some examples illustrating the broad functionalities of the GABBs building blocks are described next.

A. Weather Data Exploration

Funded by the Indiana Department of Transportation, a group of atomospheric scientists were interested in converting massive amounts of real-time weather modeling data into useful information that helps stakeholders to make timely decisions on how to efficiently distribute resources during winter severe weather conditions. PyMapLib was used to develop a multidimensional data visualization tool in Python called **Weather Data Explorer** that allows users to dynamically explore large amounts of weather modeling data using temporal and geospatial queries and an interactive map display. The raw data are stored in a multidimensional binary format, which includes many variables related to winter weather conditions. The data is automatically ingested into a MySQL Fabric database in a normalized format upon completion of a daily model simulation. The tool queries and fetches data from the database via a REST API interface. The data is further processed and rendered on a map widget provided by PyMapLib



Date / Time 11/02/2015
From: 00:00:00
Date / Time 11/02/2015
To: 08:00:00
○ Aggregate Sum
● Animation(Time-series)
□ Aggregate Sum
○ Bounding Box
● INDOT Area (?)
○ Maintanance Unit
○ Sub-District
○ District
● Indiana
Variable: Ground heat flux

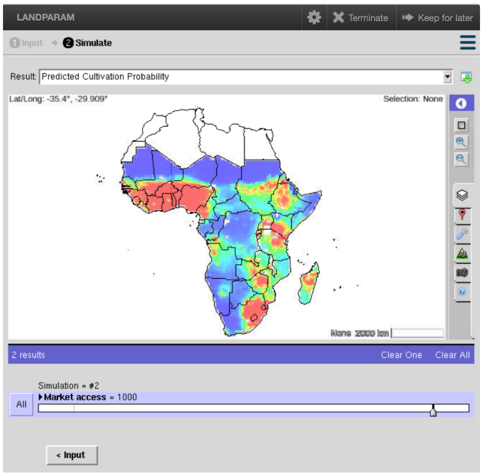
Fetching 9 / 9 data...
Processing dataset...
Animation started.
Process done...

Fig. 5. Winter Weather Explorer built using PyMapLib

(Fig. 5). With an out-of-box map viewer widget and configurable map controls and plugins, the PyMapLib API makes developing such GIS-enabled applications much easier for application developers who are not familiar or do not wish to deal with details of map data management and rendering.

B. *Study of Climate Change Impacts and Land Sustainability*

Land supply elasticity is a key parameter in assessing the land use response to changing environment, market conditions and policies. It reflects the fact that cultivation decisions depend on land profitability as well as on land suitability. In spite of its importance, high-resolution and aggregatable data on this elasticity is sparse. Using the map-enabled RAPPTURE toolkit and GeoVis render server, a group of Agricultural Economists developed the LandParam tool, aiming to provide land supply and transformation elasticities at any user-defined resolution. As shown in Fig. 6, a RAPPTURE map viewer widget is directly embedded in the user interface allowing users to explore the tool's geospatial output on an interactive map. Users may run different land use scenarios and compare the model outputs from different runs.



Result: Predicted Cultivation Probability

Lat/Long: -35.4°, -29.909°

Selection: None

2 results

Simulation = #2

Market access = 1000

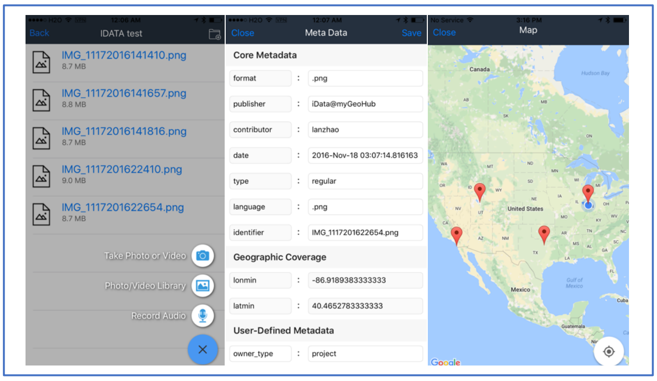
Fig. 6. The LandParam tool with mapping functions

C. *Enabling Data Collections Using Handheld Devices*

To help field workers upload and share their data easily using smart devices while working in the fields, an app called GrABBs was developed for the iOS and Android platforms. GrABBs serves as a proof-of-concept in enabling third party apps to connect to iData using the iData REST API. It provides features including secure authentication, iData files browsing, upload of different data types from a device (audio, video, image, pdf, etc), automatic geospatial information extraction, metadata annotation, and common file management (download, delete, rename, and edit metadata). A user can also visualize data that has GPS coordinate information on a map interface (Fig. 7).

D. *Managing Hydrologic Modeling Data Workflow*

The iData service API and the ability to invoke hub tools programmatically has been used by a group of hydrologists to manage their data flow in studying flooding in the Mississippi River Basin. In this study, the researchers developed the Soil and Water Assessment Tool (SWAT) model [6] for the Upper Mississippi River Basin and ran the models using a web application called SWATShare (mygeohub.org/groups/water-hub/swatshare) hosted on MyGeoHub. The output of the model simulation can be saved directly to the iData repository with automatically generated metadata using the iData REST API. Using the same API, the researchers can then load the result into another MyGeoHub application called SWATFlow (mygeohub.org/groups/water-hub/swatflow) to visualize the hydrograph for the streams of interest. Further, researchers can click on a point on the hydrograph and launch a hub tool called Water Extent Viewer which displays the flood extent of that stream at the selected time using the output from a LISFLOOD-FP hydrodynamic model [7]. The flood extent viewer is a hub tool that uses PyMapLib for geospatial data visualization. Using this workflow, researchers can create end-to-end experiments and explore results dynamically without the need to manually move data around.



IMG_11172016141410.png
8.7 MB

IMG_11172016141657.png
8.8 MB

IMG_11172016141816.png
8.7 MB

IMG_1117201622410.png
9.0 MB

IMG_1117201622654.png
8.7 MB

Core Metadata

format : .png

publisher : iData@myGeoHub

contributor : lanzhao

date : 2016-Nov-18 03:07:14.816163

type : regular

language : .png

identifier : IMG_1117201622654.png

Geographic Coverage

lonmin : -86.9189383333333

latmin : 40.4652783333333

User-Defined Metadata

owner_type : project

Fig. 7. GrABBs showing the interfaces for selecting a data source, displaying the metadata, and the map view.

E. *Total Camp for Education and Training*

A couple of the GABBs tools were used in a summer session for middle school students in June 2016 to introduce geospatial technologies to them. TOTAL (Turned Onto Technology & Leadership) Camp included 36, 8th and 9th graders from diverse backgrounds and from around the United States. Participants received a presentation covering geospatial data and research and an introduction to the GeoBuilder and MultiSpec tools. The following day, each student had 75 minutes of hands-on time with exercises to find locations within Indiana with highest reported rain events using GeoBuilder and to map a flood event in southern Illinois and Indiana using MultiSpec. The GABBs team worked closely with these participants in multiple sessions, helping them with technical questions and, at the same time, collecting valuable feedback.

IV. Packaging and Dissemination

Since GABBs is designed to provide reusable building blocks that can be used with any HUBzero instance, it becomes necessary to simplify the installation of these components on any pre-existing hub or when deploying a new GABBs-enabled hub. In addition to the central HUBzero instance, GABBs also relies on additional external resources such as the GeoVis rendering server and the iRODS server implementing geospatial data management. Deploying these various servers, installing necessary software, and configuring the connections between these resources can be a challenge for our science users who do not have server administration experience. To alleviate this challenge, our goal was to make GABBs installation as simple as possible while also catering to varying user needs and expertise. Interested, casual users can always come to MyGeoHub to test out the GABBs features and follow new developments and tools. For users seeking to set up their own GABBs-enabled hub installation, various options are offered differing in their ease of setup and supported features.

The simplest approach to packaging a server with some installed software packages is a virtual machine (VM). VMs have the advantage that they can be installed and launched on personal computers making it very easy for users to get started. In fact, HUBzero publishes a VM for use with popular VM software such as VirtualBox and VMWare Workstation. This approach fails though when there are several interconnected servers that need to be set up, with some servers having non-trivial hardware requirements. In the GABBs case, the need for a GeoVis render server with a dedicated graphics-processing unit (GPU) prevents it from use on all except very well provisioned desktop or laptop machines. However, a simpler single server setup that only contains a subset of the GABBs components is made available as a VM. This VM contains an iRODS server as well as a hub installation, iData, and the MultiSpec tool. It provides interested users with a sandbox environment to explore the data management capabilities of iData and its integration with the hub tool environment.

An Amazon Web Services (AWS) CloudFormation template is currently under development for users interested in a complete and fully customizable GABBs installation. Cloud computing services such as AWS are ideal for such deployment tasks involving multiple, highly interconnected resources. More importantly, they provide deployment management services (CloudFormation in the AWS case) that simplify the task of deploying the actual compute resources, scripting the installation and configuration of necessary software and setting up of interconnections between various resources. Moreover, when combined with auto-scaling and load balancing capabilities provided by AWS, this installation can be scaled up to support a large number of users making it ideal for production setups in the cloud. Another useful byproduct of this cloud-enabling process is that the software packages (rpms and debs) that are required in scripting automatic software installs can be used to add these GABBs components to pre-existing hubs. These packages will be released as open-source, allowing expert users to install them on their own pre-existing or new hub installations without having to necessarily overhaul their setup to use AWS.

V. Broader Impact and Conclusion

In this paper, we described the design, implementation, and application of GABBs building blocks to facilitate scientists from different domains in bringing their data and tools online to share with the community. GABBs expands the open source HUBzero platform with new capabilities for the broad community that uses geospatial data. Different from web-based mapping libraries such as OpenLayers and Leaflet which require web programing skills or commercial software such as Google Maps API or ArcPy, and desktop tools like ArcGIS (also commercial) or QGIS which are comprehensive GIS software, GABBs focus on enabling users with different levels of programming expertise to create their customized tools online, self-manage and share their datasets, as well as to create data processing pipelines via component linking. This approach allows HUBzero to support the application and data needs of many more science, engineering and educational domains. One such application is the development of geospatial training and educational materials for undergraduate and even secondary education such as the TOTAL Camp. The easy to use, web-based applications created with these new tools in the HUBzero collaborative environment will provide K-12 students with engaging geospatial-based, on-line activities that improve comprehension of geography, GIS, and remote sensing.

As of now, the major development work has been completed. Our current focus is to simplify the installation of the software and engage more user communities. Other areas that we are looking into include interoperability with other cyberinfrastructure systems such as HydroShare and Brown Dog (http://browndog.ncsa.illinois.edu/) to further broaden the use of GABBs software.

Acknowledgment

This work has been supported in part by the NSF grant #1261727.

References

[1] Tarboton, D. G., et al, (2014), "HydroShare: Advancing Collaboration through Hydrologic Data and Model Sharing," in D. P. Ames, N. W. T. Quinn and A. E. Rizzoli (eds), Proceedings of the 7th International Congress on Environmental Modelling and Software, San Diego, California, USA, International Environmental Modelling and Software Society (iEMSs), ISBN: 978-88-9035-744-2.

[2] Liu, Y.Y., Padmanabhan, A., and Wang, S. 2014. "CyberGIS Gateway for Enabling Data-Rich Geospatial Research and Education." *Concurrency and Computation: Practice and Experience*, http://dx.doi.org/10.1002/cpe.3256.

[3] M. McLennan, R. Kennell, "HUBzero: A Platform for Dissemination and Collaboration in Computational Science and Engineering," Computing in Science and Engineering, 12(2), pp. 48-52, March/April, 2010

[4] Wan, W., Zhao, L., Shin, J., Zhe, S. and Song, C. X. PyMapLib for Rapid Development of Geospatial Data Analysis Tools. *The Third International Conference on CyberGIS and Geospatial Data Science* (*CyberGIS16*), Urbana, IL, July 26–28, 2016.

[5] Biehl, Larry and David Landgrebe, 2002, "MultiSpec – a tool for multispectral-hyperspectral image data analysis", Computers & Geosciences, Vol. 28, no. 10, Dec. 2002, pp 1153-1159. http://dx.doi.org/10.1016/S0098-3004(02)00033-X.

[6] Arnold, J., Moriasi, D., Gassman, P., Abbaspour, K., White, M., Srinivasan, R., Santhi, C., Harmel, R.D., Griensven, A. Van, 2012. SWAT: model use, calibration, and validation. Trans. ASABE 55, 1491–1508.

[7] LISFLOOD MP: http://www.bristol.ac.uk/geography/research/hydrology/models/lisflood.