GABBs - Reusable Geospatial Data Analysis Building Blocks for Science Gateways

Carol Song, Ph.D.
Purdue University

International Workshop on Science Gateways
June 19-21, 2017

This work is supported in part by the NSF grant #1261727
Co-Authors & GABBs Team

The HUBzero team
Outline

• Background
• GABBs components and integration with HUBzero
• GABBs-enabled science applications
• Dissemination
Motivation & Driving Use Cases

Geospatial data is now everywhere from everyone!
Collaboration around data

- Hydrology
- Meteorology
- Remote sensing
- Climate science
- Create project
- Manage team
- Share data
- Track metadata
- Publish
- Create and use tools
- View data
- Browse and search
- Crowd sourcing
- Agriculture economics
- Phenotyping
- Communications
- Training & education
- Crowd sourcing
Software stack for spatial data

It is definitely not trivial to deal with geospatial data (processing, displaying, exchange/sharing, etc).

Choices, choices, choices ...
GABBS: Geospatial Data Building Blocks

Lower the barrier
Make it easy to visualize geospatial data
Make it easy to share geospatial/georeferenced data
Open source, community driven

Build a geospatially-enabled, integrated, self-service collaboration platform

Broaden participation
Faster dissemination
Enhance learning
Guiding Principles

Making it **easy** to manage, visualize and share geospatial data and applications

- self service
- enable end-to-end workflow
- tools that domain scientists and high school students can use
- easy to share “live” dataset (interactive, visual)
- accessible and available
- science driven, co-development
- Reusable software modules/building blocks for other gateways
High-level View

End User

Computation
Visualization
Data Sharing

New Capabilities

Maps
Overlays
Data management

Control widgets
Geo-processing
Data sharing

Data presentation
Data formats
Remote servers

Tool builder
Standard protocols

Remote servers
Data-Tool connectors

HUBzero Platform for Scientific Collaboration
Computation tools and online databases, Content publishing, Collaboration (group, project), Learning (courses, self-help), Support (tickets, Q&A), Community (forum, review, calendar)
User Community

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

iData

iData service API
Specific Goals

• Integrated data management environment with built-in geospatial data support (*iData/Project*)

• Toolkits for rapid application development, no GIS programming expertise required (*Rappture, pyMapLib*)

• Data visualization builders and tools that require no programming (*GeoBuilder, MultiSpec, mobile app*)

• Production system open to research and education use, 24x7

• Packaging for easy installation
GABBs Data Considerations

- Support standard protocols (WMS, WFS, WMTS, ...)
- Geospatial viewing (quick and comprehensive)
- Automated metadata extraction
- Geospatial search
- Easy data publication
- Bridge the separate tool space and web server space to enable dynamic work flows
- Seamless workflow for users -> integrate with HUBzero’s Project implementation
End-to-end work flow

Create Project, add members, start collaborating!

Manage files, edit metadata as needed, view to verify
End-to-end work flow

Open with tool, and start interacting and exploring data

Start new publication, select data sets and publish with DOI!
Data Service API

• Access to all project file operations
  – Create, delete, rename
  – Metadata management

• Useful for third-party applications
  – Authorization via OAuth maintains access control
  – Applications for building data repositories (crowd sourcing, etc)

• Useful for augmenting metadata not automatically captured
  – Typically used in hub tools
Example: GrABBSs Mobile Application

- Enable users to connect to GABBSs iData using devices
  - iPhone and Android versions
  - Browse, upload, download, annotate, map display, rename, delete
  - Data sources: Camera, photos, videos, record audio
  - Automatically extract geospatial information
Geospatial Data Visualization

Rappture Toolkit

- Geospatial Output Widget
- Render server support
- 2D / 3D support
- Rappture applications and supported languages
- Tcl/Tk

PyMapLib

- Python Based
- Generic Framework
- 2D Data support
- Compatible with C++ & Java user interfaces
Rappture & Render Servers

HUBzero Infrastructure

Web Server

Middleware

Submit Proxy

Tool Containers

Visualization

External Services

Grid
### Visualization

**HUBzero Infrastructure**

- Web Server
- Middleware
- Submit Proxy

**Visualization Type**

- Molecules
- 3D Structures
- Geospatial Data

**Supported?**

- Molecules
  - PyMOL
- 3D Structures
  - Vtk
- Geospatial Data
  - Unspecified
Visualization

HUBzero Infrastructure

Web Server

Middleware

Submit Proxy

Tool Containers

Visualization

Visualization Type

Molecules

3D Structures

Geospatial Data

Supported Data Formats:

Raster: GeoTiff, netCDF, PNG, jpeg, TMS, WMS, XYZ

Vector: ESRI shapefile, TFS, WFS, PostGIS

Model: 3D model formats of OpenSceneGraph

Supported?

☑ PyMOL

☑ VTK

☑ OSGEarth
## Visualization

### Visualization Type | Supported ?
--- | ---
Molecules | ✔️ PyMOL
3D Structures | ✔️
Geospatial Data | ✔️ OSGEarth

### Supported Data Formats:
- **Raster:** GeoTiff, netCDF, PNG, jpeg, TMS, WMS, XYZ
- **Vector:** ESRI shapefile, TFS, WFS, PostGIS
- **Model:** 3D model formats of OpenSceneGraph
5 Step Widget Integration

1. Setup the MapViewer
2. Describe the Layers
3. Configure Attribute Placard
4. Configure the Camera
5. Render Results
PyMapLib

- Generic framework for geospatial data visualization
- Requires little expertise in GIS programming
- Handles common data types, including:
  - Raster, ESRI shapefile, Delimited text
- Uses PyQGIS as the basic geospatial data rendering engine
- Configurable map tools for the map viewer widget
  - Pan, zoom, select, map tips, layer management, map overview, map value inspection and plotting, map style
- Default map viewer widget
  - Import as a QT widget and extend as needed in Python code
  - Embed as-is in Java or C++ tools
- Not HUBzero specific, can be used in any standalone python program
- Available in github: https://github.com/waneric/PyMapLib
• Add an Open Street Map as the base layer

```python
self.map = gabbs.maps.Map("WorldMap")
self.map.setMapCenter(-86, 39)
self.map.setMapZoom(7)
self.map.setMapScale(3, 9)
self.mapContainer.addLayer(self.map)
```

• Add map tools to the map viewer as toolbar buttons

```python
self.mapContainer = gabbs.maps.MapContainer()
self.mapContainer.setLayerControl(True)
self.mapContainer.setPanControl(True)
self.mapContainer.setZoomControl(True, size = "CUSTOM", options = "ZOOMIN, ZOOMOUT")
self.mapContainer.setSelectControl(True, size = "CUSTOM", options = "SINGLE, RECTANGLE, POLYGON")
self.mapContainer.setPlugin("drawing")
```
GeoBuilder

Table of data

To

Map of data

- No programming
- Instant visualization

https://mygeohub.org/tools/geobuilder
What’s on the hub?

- GeoServer
- Render Server (GPU based)
- GDAL
- MultiSpec
- iData
- hubzero
- Apache Solr
- Storage
- File System
- PostgresQL
- MySQL
- Metadata
- iRODS
- Tools
- CMS
- Search

GeoVis

XSEDE

Community Clusters
<table>
<thead>
<tr>
<th>Feature</th>
<th>Virtual Machine</th>
<th>Machine Image (AMI)</th>
<th>Packages</th>
</tr>
</thead>
<tbody>
<tr>
<td>iData</td>
<td>✓</td>
<td>✓</td>
<td>rpm</td>
</tr>
<tr>
<td>Rappture</td>
<td>✓</td>
<td>✓</td>
<td>rpm / tar</td>
</tr>
<tr>
<td>MultiSpec</td>
<td>✓</td>
<td>✓</td>
<td>rpm</td>
</tr>
<tr>
<td>PyMapLib</td>
<td>✓</td>
<td>✓</td>
<td>rpm</td>
</tr>
<tr>
<td>GeoBuilder</td>
<td>✓</td>
<td>✓</td>
<td>rpm</td>
</tr>
<tr>
<td>Render Server</td>
<td>✓</td>
<td>✓</td>
<td>tar</td>
</tr>
</tbody>
</table>

~8000 users a year

120K visitors a year
# Release Features Matrix

<table>
<thead>
<tr>
<th>Feature</th>
<th>MyGeoHub.org</th>
<th>Virtual Machine</th>
<th>Amazon Machine Image (AMI)</th>
<th>Packages</th>
</tr>
</thead>
<tbody>
<tr>
<td>iData</td>
<td>[x]</td>
<td>[x]</td>
<td>Multi-Instance</td>
<td>rpm</td>
</tr>
<tr>
<td>Rappture</td>
<td>[x]</td>
<td>[x]</td>
<td>Single Instance</td>
<td>rpm / tar</td>
</tr>
<tr>
<td>MultiSpec</td>
<td>[x]</td>
<td>[x]</td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>PyMapLib</td>
<td>[x]</td>
<td>[x]</td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>GeoBuilder</td>
<td>[x]</td>
<td>[x]</td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>Render Server</td>
<td>[x]</td>
<td>[x]</td>
<td></td>
<td>tar</td>
</tr>
</tbody>
</table>
# Release Features Matrix

<table>
<thead>
<tr>
<th>Feature</th>
<th>MyGeoHub.org</th>
<th>Virtual Machine</th>
<th>Amazon Machine Image (AMI)</th>
<th>Packages</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Multi-Instance</td>
<td></td>
</tr>
<tr>
<td>iData</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>rpm</td>
</tr>
<tr>
<td>Rappture</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>rpm / tar</td>
</tr>
<tr>
<td>MultiSpec</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>PyMapLib</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>GeoBuilder</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>Render Server</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>tar</td>
</tr>
</tbody>
</table>
## Release Features Matrix

<table>
<thead>
<tr>
<th>Feature</th>
<th>MyGeoHub.org</th>
<th>Virtual Machine</th>
<th>Amazon Machine Image (AMI)</th>
<th>Packages</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Multi-Instance</td>
<td></td>
</tr>
<tr>
<td>iData</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>rpm</td>
</tr>
<tr>
<td>Rapapture</td>
<td>✔️</td>
<td></td>
<td></td>
<td>rpm / tar</td>
</tr>
<tr>
<td>MultiSpec</td>
<td>✔️</td>
<td>✔️</td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>PyMapLib</td>
<td>✔️</td>
<td></td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>GeoBuilder</td>
<td>✔️</td>
<td></td>
<td></td>
<td>rpm</td>
</tr>
<tr>
<td>Render Server</td>
<td>✔️</td>
<td></td>
<td></td>
<td>tar</td>
</tr>
</tbody>
</table>
• Video illustrating the use of an AWS CloudFormation template to deploy a GABBs-enabled HUBzero instance in Amazon Web Services (AWS)

https://mygeohub.org/resources/1291
In Summary

Open source, Self serve
Platform for collaboration around geospatial data

Integrated geospatial data support

Toolkits for building map-based applications

Data exploration and analysis tools for end users

Data service API for third party tools

Create project
Publish
Manage team
Share data
Track metadata
Browse and search

Create and use tools
View data

Dissemination

VMs
& Linux packages
Questions?

Contact: carolxsong@purdue.edu
Project: http://mygeohub.org/groups/gabbs
Production site: http://MyGeoHub.org

This work is supported in part by the NSF grant #1261727