CIF21 DIBBs: Integrating Geospatial Capabilities into HUBzero

PI: Carol Song, NSF award #1261727

Consider a researcher who has taken smartphone pictures of building damage from earthquakes. Another researcher has collected seismic data from the region and would now like to put these two datasets together on a map of the region. Consider another researcher who has collected water temperature readings from ocean buoys. She would like to plot that data on a map with overlays of ocean currents extracted from satellite data. The GABBs (Geospatial Data Analysis Building Blocks) project seeks to provide ready-to-use geospatial data widgets and toolkits that will make it possible to build such web-accessible data views with zero to minimal programming. In addition, researchers will be able to self-manage geospatial data and construct complex data-driven workflows involving simulation and visualization tools.

Our primary challenge, is in integrating these geospatial capabilities with cyberinfrastructure supporting the entire research data lifecycle from data collection, management and processing to publication. Uniform access to a central data store is required throughout the lifecycle, but often through varied interfaces. Data collection may be accomplished on a smartphone; data management is web-based; processing tools are containerized to simplify dependency management, and data publications require metadata that will aid discovery via search. Geospatial data is also highly structured and may contain provenance information. Such metadata needs to be captured automatically, while allowing for subsequent augmentation. Geospatial previews are vital in verifying data relevance and error checking. Such processing is best implemented adjacent to the data store to avoid unnecessary data transfer; while also abstracting these operations across the various access points. In practice we achieve these goals by integrating iRODS with the HUBzero cyberinfrastructure framework. iRODS storage underlies the collaborative HUBzero "project"; project files are then made available in web-accessible, containerized HUBzero tools. GABBs data access APIs enable ingestion from devices such as smartphones. FUSE and bind mounts expose iRODS storage uniformly as local files in the hub web-server and tool containers, simplifying development. File processing is instrumented in iRODS micro-services that are triggered on-demand or automatically in response to various file events. Metadata is indexed into Apache Solr, enabling search by keywords and geospatial extents.

Our intent is not that GABBs be a monolithic GIS library; instead we seek to build synergistic capabilities with other CI projects, such as data and tool interactions between GABBs and other CIs, and additional services to augment GABBs functionality. Extensions to our metadata extraction capabilities with those of NCSA's BrownDog are being explored. iRODS storage supports data sharing via zone federation, allowing our hosted tools to be run on data managed by other CI projects (e.g. Hydroshare) in their own iRODS storage. However, such interoperation has its challenges in supporting uniform resource discovery across the various providers. They may each have their own metadata schema, requiring a basic minimum metadata standard for communication. Exploratory work in this direction was proposed in recent meetings involving the DIBBs projects and we intend to support such a standard. Geospatial research data, typically multi-dimensional and of large size, poses significant challenges on the overall GABBs system performance and scalability. Similar challenges also arise when we need to deal with large data across two CI systems such as between HydroShare and GABBs. Techniques such as shared file system, on-demand processing, resume-able file transfer, and iRODS zone federation have been used within GABBs and across systems to tackle this problem. Globus is also being investigated to further improve the user experience. Interoperability between major CI projects is also significantly affected by constantly evolving software and interface dependencies, both due to evolving technologies as well as shifting schedules, priorities and resource allocations. In our integration with the HydroShare project, for example, we opted for a "loosely coupled integration" through OAuth authentication, compatible metadata/data model, and data services REST APIs on both sides. This reduces complexity of the integration effort, allowing us to make phased progress in releasing new functionality.