Rapid Open Geospatial User-Driven Enterprise (ROGUE)

Joint Capability Technology Demonstration White Paper

January 30th, 2014

ROGUE in Brief

The Rapid Open Geospatial User-Driven Enterprise (ROGUE) Joint Capability Technology Demonstration (JCTD) is committed to providing an innovative approach to geospatial data development, management, and sharing that leverages the combined strengths of institutional interactions, provides provenance, and enables rich collaboration. These objectives are being accomplished by leveraging a suite of open source software, based on open standards, in order to allow the widest availability and interoperability. This approach helps to overcome current challenges that organizations experience when trying to collaboratively develop and share geospatial information.

The ROGUE JCTD is providing the capability for organizations to collaboratively develop geographic feature data with traditional and non-traditional partners by improving the ability of the OpenGeo Suite to ingest, update, and distribute non-proprietary feature data utilizing open source software and open standards. By integrating these capabilities with Pacific Disaster Center's DisasterAWARE platform and the Department of State Humanitarian Information Unit (HIU), the DoD and mission partners are able to plan, analyze, and collaborate using dynamic map data supporting humanitarian and disaster response.

The OpenGeo Suite is a robust set of software capabilities that provides a full-featured spatial data infrastructure that has been designed for the web and is based on open standards. GeoNode and GeoGit build on this foundation, providing a foundation to enable discovery of and collaboration on geospatial information.

ROGUE in Practice

The development of GeoGit is the cornerstone of the ROGUE project. With this capability we can move into a paradigm that allows for distributed collaboration on geographic data development and management. GeoGit will provide the ability to maintain a history of the changes to geospatial vector data, track who provided the changes, and store comments on the reasons for the changes. GeoGit provides the ability to track and maintain the provenance of the data in a distributed and sometimes connected environment. By combining GeoGit with user-focused applications and portals, we can do this in such a way that the provenance of the data is maintained throughout the process.

The Pacific Disaster Center (PDC) and the DoS HIU will become the first two sites to which the base ROGUE capabilities will transition. At PDC, the ROGUE-enhanced OpenGeo suite will be integrated to allow for collaborative data development and management in support of humanitarian assistance, disaster response, and disaster risk reduction activities around the globe. Further, PDC's DisasterAWARE decision support platform will be enhanced to allow visualization of collaboratively developed content in its Map Viewer. The HIU is integrating ROGUE capabilities into its CyberGIS project, an existing effort that utilizes the OpenGeo Suite to build web mapping and geographic data sharing applications focused on complex humanitarian emergencies. MapStory.org is integrating the same suite of capabilities to enable a public commons of geospatial information organized around telling stories with maps.

The standards based approach that we have embedded in ROGUE allows for the integration of a complete ecosystem of tools and capabilities, such as web client interfaces, focused mobile applications, and desktop clients.

1 Introduction

The Rapid Open Geospatial User-Driven Enterprise (ROGUE) effort is focused on providing a new approach to geographic data that leverages the strengths of institutional interactions, provides provenance of data source, and enables collaboration. All of this is being accomplished through a suite of open source software that is based on open standards in order to allow for the widest availability and interoperability.

ROGUE was initiated in August 2012 as a Joint Capability Technology Demonstration (JCTD). ROGUE team members include:

- US Army Corps of Engineers, Engineer Research & Development Center (ERDC) - Technical Manager
- US Southern Command (SOUTHCOM) - Operational Manager
- Pacific Disaster Center (PDC) - Transition Manager
- Department of State - Humanitarian Information Unit (HIU) - Deputy Transition Manager
- Deputy Assistant Secretary of Defense, Rapid Fielding Office – Oversight Executive

The ROGUE JCTD is providing the capability for organizations to collaboratively develop, manage, and share geographic feature data with traditional and non-traditional partners by improving the ability of the OpenGeo Suite to ingest, update and distribute non-proprietary feature data utilizing open source software and open standards. By integrating these capabilities with Pacific Disaster Center's DisasterAWARE[1] platform and the Department of State Humanitarian Information Unit, the DoD and mission partners are able to plan, analyze, and collaborate using dynamic map data supporting humanitarian assistance, disaster relief, and other geospatial collaboration scenarios. The ROGUE JCTD timeline is August 2012 through August 2014.

2 Problem Statement

In the past decade we have seen an increased use in locational information and the value of opening up data for contribution. The question is no longer whether we should be leveraging the capabilities of partner organizations, but rather what is the best way to do so. As a reflection of our increasingly connected world, most endeavors involve multiple organizations with a need to share their unique perspectives and expertise. In many cases these engagements are *ad hoc*, as in the case of a disaster response, without the benefit of established agreements for data sharing or workflows. These endeavors also seek to benefit from the wider awareness of internal and external viewpoints. It is widely acknowledged that local knowledge should inform operational decisions, but

[1] PDC's Emergency Operations Systems (EMOPS), a version of DisasterAWARE which is used widely with USG and by PDC international and NGO partners can be accessed at www.pdc.org/emops

organizations are still wrestling with how to treat volunteered information versus “authoritative” information. This current reality challenges the static nature of spatial data infrastructures as they’ve been implemented to date.

Currently the typical way to share geospatial data is to copy or email files in more commonly used formats such as shapefiles (SHP), Keyhole Markup Language (KML) or comma-separated values (CSV). As conditions change (i.e., as the feature and attribute data are updated), the files are modified, and then hopefully passed around again. This results in at least two major problems, particularly for rapidly evolving situations such as humanitarian assistance and disaster response (HA/DR). First, the data history is not - and often cannot be - properly tracked. Essentially no information on how the data evolved is preserved. Typically only the most basic of metadata is preserved. Generally, the recipient of the updated data does not have information on specific changes made to a feature: who made the changes, when they were made, upon what basis they were made, etc. This information is the provenance that is critical to understanding the lineage of the data. Second, it is often the case that more than one organization is actively updating feature data at the same time, albeit completely unknown to each other, resulting in disparate versions of the data. In a well-connected environment, the use of modern Open Geospatial Consortium (OGC) web services provides one solution to this problem: edits are instantly seen by all who are editing or consuming the data. However, not all geospatial development or consumption takes place in an internet-enabled environment. Network connectivity is often limited, especially in deployed environments, which characterize most HA/DR situations. Connectivity is sporadic or non-existent due to the severity of a disaster, shortage of resources, or the remoteness of the operational area. Recognizing this, any collaboration and data sharing capabilities must address disconnected and limited-connectivity scenarios as well as the well-connected situations.

3 The Building Blocks of ROGUE

The mission of ROGUE is to enable collaboration on development of geographic information by multiple organizations. The capabilities being delivered address some of the core challenges in the geospatial community right now – disconnected editing, provenance of data, and collaborative development of geographic information. Each component being developed and integrated as part of ROGUE – GeoGit and GeoNode - provides unique capabilities to further the cause of collaborative geospatial data development, management, and sharing.

The OpenGeo Suite[2] is a robust set of software capabilities that provides a full-featured spatial data infrastructure that has been designed for the web and is based on open standards. **GeoGit** and **GeoNode** build on this foundation, providing the potential to completely change the game away from the existing limitations that plague attempts to provide a dynamic spatial infrastructure. Recent efforts by Boundless[3] and the MapStory

[2] More detailed information about the OpenGeo Suite can be found at http://boundlessgeo.com/solutions/opengeo-suite/
[3] For more about Boundless (formerly OpenGeo), see http://bounlessgeo.com

Foundation[4] have created the opportunity to drive forward with open technology that enables greater collaboration around geographic data. The ROGUE project is taking this foundation and leveraging it to deliver a set of capabilities that will shift the paradigm of how geospatial data is shared by organizations. In 2013, the first full versions of two powerful pieces of technology will be released – GeoGit[5] and GeoNode.

GeoGit is a new approach to a distributed version control system for geographic information. It is inspired by the very successful Git distributed version control system that creates a unique methodology for software developers to collaborate and share source code. Geospatial users, groups, and organizations have long required a similar capability to share and collaborate on geospatial data. Currently there are not any solutions to the presented problem for geospatial data. ROGUE will incorporate the software developer success methodology of Git and apply a similar approach tailored to geospatial data known as GeoGit.

GeoGit is a set of utilities that enables the creation of - and interaction with - repositories of geospatial vector datasets. GeoGit enables the creation of these repositories with the following advantages:

- Enables workflows allowing collaboration around creating and editing of geospatial data
- Enables users to manage updates (including conflict resolution[6]) from known contributors as well as ad hoc contributors
- Contains full version history including who made the change and what changed
- Provides the ability to maintain different versions of data
- Provides the ability to operate in distributed and sometimes connected environments

The second piece of powerful technology leveraged by ROGUE is **GeoNode**[7]. GeoNode is an implementation of a spatial portal that allows for interactive discovery and visualization of geographic content. GeoNode is the result of a collaborative effort between OpenGeo and the World Bank and is the key technology that enabled the creation of MapStory.org[8]. The underpinning concept behind GeoNode is to encourage users to collaborate by lowering the barrier to entry and providing value back to the user at every step of the way. Traditional spatial data infrastructures set metadata and data as priority over the end-user. We believe that the end-user is the first priority, that the data

---

4 The MapStory Foundation established http://mapstory.org, which utilizes the OpenGeo Suite, GeoNode, and will eventually incorporate GeoGit to provide a commons for storytelling based around geographic information.

5 The details of the GeoGit project have been outlined here and will be continuously updated: http://geogit.org

6 Conflict resolution is the process of resolving the changes made by multiple users to the same data element. In the case of GeoGit these elements are geographic features and attributes. This is essential to allowing integration of multiple changes over time.

7 Additional information about GeoNode can be found at http://geonode.org/

8 http://mapstory.org

must be accessible, and that much of the critical metadata can be derived (and enhanced) from the data and users themselves. For example, the data provides the bounding box, the time it was generated, and the attributes. The user information is available from the user profile.

Instead of scattered files and isolated databases, the infrastructure should be built around nodes of collaboration. These nodes should exist wherever anyone consumes spatial information. A large organization would host a large node, but there would be nodes at individual departments or field offices as well. In fact, a node would exist wherever there is a need to control the state (and provenance) of the data. There would also be a wide variety of mobile, web, and desktop applications that can connect to any of these nodes. In this federated approach, the ability to work distributed and disconnected is an inherent characteristic.

4 A New Paradigm for Spatial Collaboration

4.1 Data Provenance

The development of GeoGit is the cornerstone of the ROGUE project. With this capability we can move into a paradigm that allows for distributed collaboration and versioning of geographic data[9]. GeoGit will provide the ability to maintain a history of the changes to geospatial vector data, track who provided the changes, and store comments on the reasons for the changes. It will provide the ability to manage what changes are made to the main repository and to review, accept, or reject those changes. It will provide the ability to create a network[10] of repositories where each data owner has control over his/her data, but every node in the network is able to collaborate on that data – even if he/she needs to disconnect from the network to perform operations and then reconnect. Therefore, GeoGit provides the ability to track and maintain the provenance of the data in a distributed and sometimes connected environment.

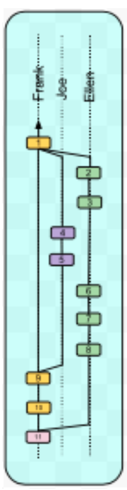

Figure 1 Branching with GeoGit

GeoGit is fundamentally different from previous geospatial versioning efforts in that it is designed to support distributed operations at its core. It does not rely on network connectivity to be fully functional. GeoGit is designed to handle projects that have a very large number of contributors. These contributors can clone the repository (either fully or a geographic subset) to their local repository, work on the data for as long as needed to accomplish some tasks, and then gracefully merge their changes with other contributors who may have been working on the same data as well.

[9] Boundless has published three white papers on Distributed Versioning for Geospatial Data: http://boundlessgeo.com/resources/white-papers/

[10] In this context we are referring to a network in the general sense in which it is defined as a series of connected nodes.

Throughout this process, the full history of all changes made to the files is maintained as well. This means the users have access to the historical data even if they are not connected to a network. Contributors can make some changes and then commit them to the repository. Alternatively, they can choose to undo recent changes or even undo any changes made by anyone else at any point in the past. They can also review the commit history of a specific feature to see what changes were made, as well as read the comments made by the previous developers/committers. Once changes are made, the contributor can connect to someone else's repository over the Internet, local area network, or a removable storage media, and push his/her changes. Changes can also be pulled from any other repository. Because each GeoGit repository has the full history of changes, it can accurately decipher exactly what changes each repository has and does not have. As a result, it can ensure that only changes that have not been applied are pulled.

Another critical feature of GeoGit is that it will provide the ability to perform spatial operations over the items it tracks. For example, a partial clone can be created from a geometry outlining a specific area of interest. Not only does GeoGit have to perform geometry-based operations, it has to perform them on potentially large datasets and do so very quickly. GeoGit also has to handle such operations across commits and branches. In order to ensure scalability, the development team is testing using the OpenStreetMap history – over 200GB of change-sets that represent the past decade of edits.

An emerging behavior that we will cultivate is the ability for users to keep working versions of their data private until it's ready for users to begin consuming and contributing. Instead of gathering full datasets on their own machines with manual management processes, they can maintain a 'clone' (or branch) of the core information model. Some users may have attributes that are private or information that isn't ready for release to the wider groups.

4.2 Towards a Collaborative Spatial Approach

Creating a better mechanism for tracking, managing, and curating data is a foundational component, but it's not the full solution for enabling collaboration. Doing so requires putting the user at the center of the infrastructure and encouraging contribution by making the barrier to contribute as low as possible. By combining GeoGit with user-focused applications and portals, we can do this in such a way that the provenance of the data is maintained throughout the process.

Under ROGUE, we are developing web services within GeoServer (a component of the OpenGeo Suite) that are integrated with GeoGit. These services are being exposed as REST[11] and OGC compliant Web Feature Services-Transaction (WFS-T). This will enable web, desktop, and mobile clients to make contributions and visualize the spatial and temporal aspects of the changes in GeoGit. Mobile application development under ROGUE is geared toward targeted data collection. The current mobile development

[11] "REST" stands for Representational State Transfer and is a predominant web service design model to support distributed systems.

allows for offline viewing and editing and syncs with the WFS-T service when it establishes connection. The web client offers additional levels of interaction. Desktop clients are typically the tools of dedicated analysts and would be the most full-featured implementations.

GeoNode serves as the link between traditional GIS and the web 2.0 world of interactive user contribution. It has been designed around the driving principle of encouraging the creation and sharing of geospatial information. GeoNode provides a venue for data creators and analysts to publish their products and make them easily accessible to everyone. It also provides the tool for users to discover geospatial information and to easily style maps based upon the information found.

GeoNode, when combined with GeoGit, provides a very powerful environment for data collaboration. This provides a low barrier of entry for users to contribute to improve the data. When multiple such nodes are set up, a community of collaboration is built where organizations are able to expand the breadth of users to include those in other organizations, including volunteers.

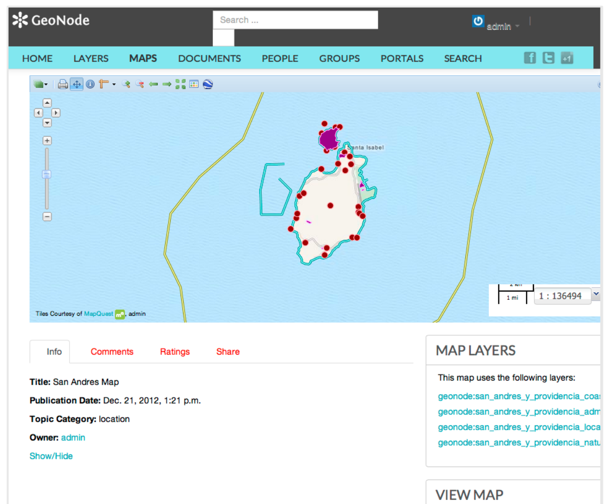

Figure 2 GeoNode

While GeoGit's emphasis is on editing data, GeoNode's emphasis is on the use of data. GeoNode makes it simple for users to create their own workflows to make the data work for them. For example, users can:

- Search for geographic information by theme, area, or contributor
- Create new maps from existing data
- Discover linkages between feature layers & published maps
- Style layers in a map to communicate a story
- Redline geospatial data to show changes in the field
- Rate or comment on data to give feedback to the data authors
- Search for maps and layers according to their usage and ratings
- Report errors and provide corrections
- Identify who is contributing in both a spatial and a temporal context

All activities in the GeoNode are tied back to user profiles, which can be configured to carry across nodes. This creates an emergence of important social aspects. It makes it possible for users to follow contributors that provide input, products, or analysis. It will also allow for organizations to take a new approach to how they publish data. The versioning from GeoGit, along with the social aspects of GeoNode, make it possible for organizations to publically post their "stamp of approval" on the current state of a dataset.

This certifies it as being legitimate for use by those who follow that organization – that the data is the best for a particular purpose. In this way, "authoritative" data does not have to be a particular database or static delivery. It can become a version stamp on a dataset that many people are continually collaborating on. Therefore, similar to the source code paradigm, there can be alpha, beta, and release versions of geospatial data managed inherently as part of the versioning system.

This new paradigm will allow authoritative geographic information and volunteered geographic information to be able to coexist. The authoritative organization can make certified copies (versions) available, and most users would clone or fork it according to their needs. They could work on a dataset outside the main repository or push the changes back to the authoritative organization. Either way, those individuals in charge of the central repository would be able to pull the new changes and incorporate them into their quality assurance process before publishing the changes as an updated version.

As more nodes come online and communities of contributors grow, it develops a set of social networks based around geographic information. Tools such as the JavaScript InfoVis Toolkit or D3.js can be used to visualize who has made contributions to particular datasets (and where), what individuals and groups are collaborating, and who are emerging as major contributors and experts on particular themes in different geographic regions.

4.3 Driving Data Collection

Once geographic information is available in a GeoNode, users can begin to leverage the network of contributors to fill gaps, make updates, and improve accuracy. The example we're using for this approach is the OpenStreetMap Tasking Manager[12], which allows for the community to post areas that need concentrated collaboration. The Tasking Manager is specific to OpenStreetMap, but ROGUE will achieve the same design goals within GeoNode for other geospatial information, which are to:

- Publish tasking requirements
- Allow contributors to identify the areas they are working on
- Provide the ability to communicate the areas that have been completed and reviewed.

The services inherent in GeoNode and GeoServer allow for publishing web feature services that can be consumed by mobile and web clients. These services will be set up by a simple wizard that guides the user through the steps to select geospatial data for collection and publishing. Through the integration of GeoNode and GeoGit, the user is able to see who is responding and contributing. These capabilities continue that social thread for collaboration on the data by providing a communication channel based around the user's needs and the unique qualities of the data.

[12] For an example of the OSM Tasking Manager see http://tasks.hotosm.org/

The first instantiation of this is an open source mobile application called Arbiter[13] that is specifically designed for data collection in the field (whether connected or disconnected). When Arbiter syncs data to GeoServer, it's done through WFS-T and all changes are tracked by GeoGit. In spring of 2013, Arbiter will be able to provide the capability to take a photo when making updates and associate that photo with a map feature. This helps provide additional visual context (such as the actual extent of damage) to the geographic information.

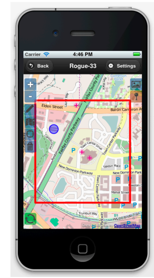

Figure 3 Arbiter

4.4 Integrated Spatial Analysis Workflows

Within a node, users are able to edit data, style layers, and create maps. Given these capabilities, they will also need basic spatial tools to manipulate the data (such as merge and subset), and to perform basic analysis (such as buffering by a distance or determining data intersects). For example, a user may want to take a spatial subset of schools in a disaster-affected area and visualize what areas fall within 5km of those schools that have not been flooded. More advanced analysis tools can be made available through Web Processing Services (WPS) – an OGC standard for geospatial analysis tools. These WPS services can be custom developed tools or a chain of tools. Once complete, the results of an analysis can be published for discovery and use by GeoNode users. The ROGUE development team has developed a prototype that provides distance and bearing between map features.

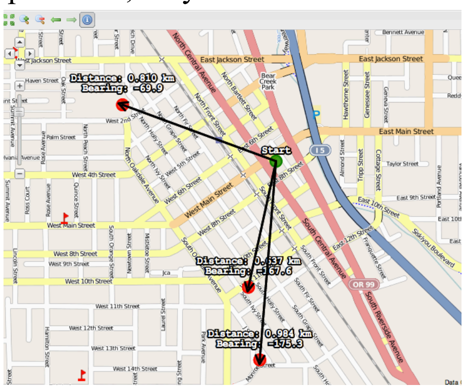

Figure 4 Distance & Bearing Analysis

4.5 Interoperability and Integration Opportunities

The ROGUE development effort will deliver a fully integrated solution consisting of the OpenGeo Suite, GeoGit, and GeoNode, complete with UI components and tools. The entire effort is open source and uses open standards (including OGC compliant services). This will ensure the greatest access to these capabilities, so that others can integrate geospatial information according to their own needs (and infrastructure) and become another node in whatever network is needed to accomplish that organization's mission.

The modular approach for ROGUE will allow for the community development approach envisioned in this paper. Different nodes in this collaborative paradigm can be implemented in different forms according to each organization's needs. Therefore, if an organization already has an interface that it has invested in or a particular operational focus, it doesn't have to replace it to realize these capabilities. It can integrate the services and back-end capabilities so that its user-base can work with the interface that is

[13] Arbiter has been developed by LMN Solutions for the ROGUE JCTD and is a public open source software project located at https://github.com/ROGUE-JCTD/arbiter. Arbiter is meant to provide an example application that others can use to develop their own data collection applications.

most familiar to them. This decreases training time and allows for the development of tailored workflows. The PDC's DisasterAWARE, along with the DoS HIU, will become the first two sites that will contain the capabilities from ROGUE.

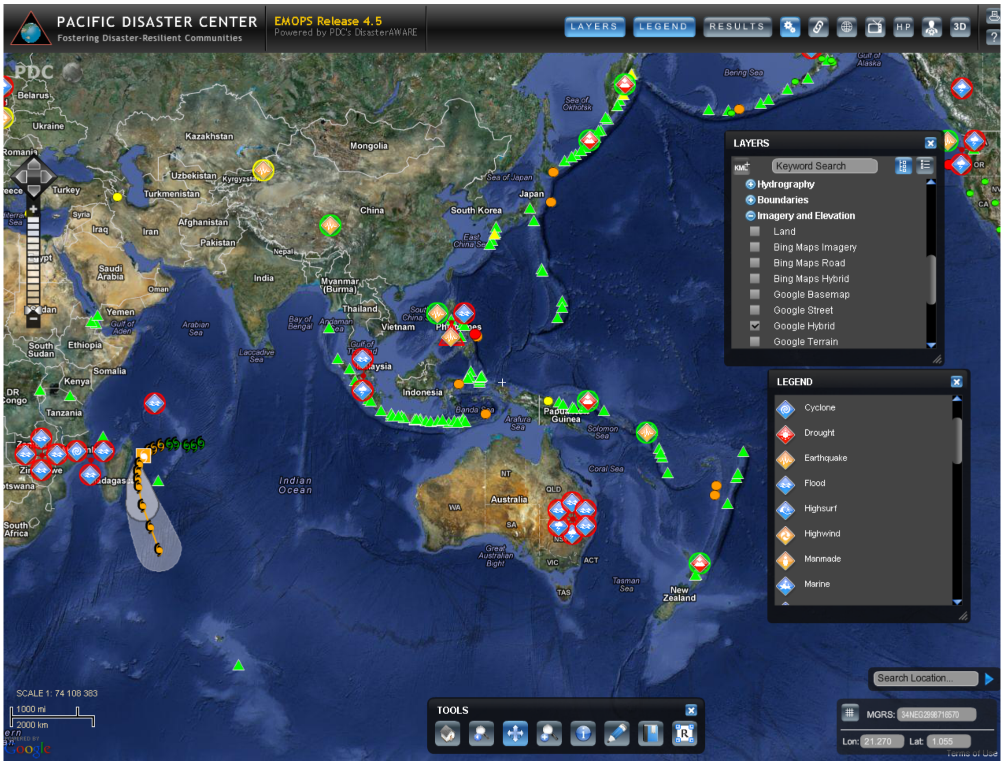

Figure 5 DisasterAWARE (EMOPS)

GeoNode will become the HIU's portal for data sharing with the broader humanitarian community, and GeoGit will reinforce the collaborative geographic data production processes developed in the HIU's "Imagery to the Crowd" projects. In addition, MapStory.org is integrating the same suite of capabilities to enable a public commons of geospatial information organized around telling stories with maps. The standards based approach that we have embedded in ROGUE allows for the integration of a complete ecosystem of tools and capabilities, such as web client interfaces, focused mobile applications, and desktop clients.

5 Implications

The provenance on every bit of data (features, layers, maps, analysis, etc.), coupled with the social aspects of a network of collaborative nodes, lets individuals and organizations share and thus build up implicit trust. That element of trust becomes part of the infrastructure itself. Authority becomes a concept based on the sum of each action and actor that has gone into the development of the geospatial information. There doesn't need to be a large trade-off between 'authoritative' and 'time-sensitive.' That consideration will become orthogonal to the whole infrastructure, with data quality tools and processes running in parallel to collection efforts. These quality assurance aspects can be both automated (edge matching, topological, completeness), and human (number of people that have signed off). The authoritative data will be stamped as such, available to all who have access to that authority's latest release. For those who truly need all updates – the working version – they will instantly know how many of the up-to-date changes they pull in are not authoritative.

Under the ROGUE JCTD, we are striving to enable the development of enterprise nodes for geospatial collaboration among mission partners. The open source software approach allows us to also transition a set of capabilities that are accessible by other groups that wish to set up the same capabilities for their own use. It also allows us to make our efforts available for community involvement and contribution[14]. By leveraging OGC standards, the potential for integration with existing capabilities and toolsets is increased even further. Therefore ROGUE is taking a community approach to allowing the development of an ecosystem for geospatial collaboration. More information about ROGUE is available at http://rogue.lmnsolutions.com

Points of Contact

Robin Rodgers
Technical Manager, USACE ERDC
robin.e.rodgers@usace.army.mil

Joshua Campbell
Deputy Transition Manager, HIU
campbelljs3@state.gov

Chris Chiesa
Transition Manager, PDC
cchiesa@pdc.org

Scott Clark
Program Manager, LMN Solutions
scott.clark@lmnsolutions.com

[14] GeoGit can be found at https://geogit.org