Common Web Mapping and Mobile Device Framework for Display of NASA Real-time Data

Jason Burks
NASAs Short-term Prediction Research and Transition (SPoRT) Center/Earth Science Office, Marshall Space Flight Center (MSFC), Huntsville, Alabama

Motivation
SPoRT and other NASA scientists produce data that can be used by others, such as the National Weather Service (NWS) and Federal Emergency Management Agency (FEMA). Two groups at MSFC are engaged in supporting disaster response:

• Jason.e.burks@nasa.gov

Java Topology Suite
f
The main focus of the ingest server is to provide a system easy to configure and ingest data, and to provide output in open standards (Fig. 2).
• We use WMS, widely used across the internet to server spatial data.
• The ingest system is configured using Extensible Markup Language (XML) files.
• XMLs control the flow of data within the server system.
• Once configured, XMLs allow the data to be ingested, properly registered, and discoverable.

Web User
GIS Application
Custom or Future DSS
WMS and tile cache
Smartphones and Tablets
Ingest Server

Figure 2. Overview of the common web mapping and mobile device framework for display of NASA real-time data, including ingest server, distribution to WMS for dissemination and tiling, and end-user applications.

In disaster response, SPoRT, SERVIR, and other NASA scientists need to be able to:
• Deliver data to collaborators in their Decision Support System (DSS) using:
  • web
  • mobile phones
  • native GIS applications
• Focus on science, not the delivery mechanism

Benefits of a Mobile Device Framework
• Supports web, mobile and GIS within one system
• Enables easy ingest of new datasets
• Enables science for decision making
  • Superstorm Sandy of 2012
  • Moore, Oklahoma Tornado of 2013.
• Control application maintenance costs by using a common web framework
• Open source reduces building and licensing costs.
• Provides data in open standards for easy integration.
• Web Mapping Service (WMS)
• Supports static and animated imagery
• Supports location based services
• iOS and Android supported for mobile at a minimum.

Figure 1. Sample mobile phone (left) and web browser (right) based application based on the framework. Data displayed is Day/Night Band difference from the Superstorm Sandy before and after the storm.

Server Architecture
• The main focus of the ingest server is to provide a system easy to configure and ingest data, and to provide output in open standards (Fig. 2).

Client Architecture
Mobile applications supporting Android, iOS, and other platforms represent significant effort and cost. The client architecture was developed by wrapping a JavaScript client inside a simple shell application using a webview in each supported client. This method has several benefits:
• use of standard web development tools
• faster development cycle
• easier to maintain and develop
• styling using web standards such as Cascading Style Sheets (CSS)

Japan

Spring (Development Framework)
Apache Camel (Enterprise Service Bus)
PostgreSQL / PostGIS Database
Servlets

Figure 4. An example of post-event imagery displayed in a common web mapping framework. Here, an ASTER false color Image from 11/13/2013 at 0322Z, after Typhoon Haiyan with damage analysis (inset).

Case Study of Web Mapping Service Applications
• Typhoon Haiyan (Yolanda) impacted the Philippines in November 2013.
• As a proof of concept, the SPoRT Disaster Team was able to easily deliver imagery to web and mobile clients with use of the system described here.
• The system dramatically reduced the level of effort and time it takes to provide NASA data in response to the disaster event.

Conclusions and Future Work
• The architecture demonstrated here provides a seamless, extendible way to deliver imagery to end-users on various platforms.
• System costs are contained by using open source projects a common JavaScript code base for mobile and web applications. Innovative client development can be performed using common web page development.
• Development of an Amazon-based cloud framework to address scaling of the architecture to handle large volumes of data and large numbers of end-users, including international users of the NASA SERVIR program.

Acknowledgements
Support for this project was provided by SPoRT, SERVIR, MSFC Office of Strategic Analysis and Communications, and the MSFC Center Innovation Fund.