NSDI Cooperative Agreements Program
Category 2: Best Practices in Geospatial Service Oriented Architecture (SOA)

Final Report

Date:	September 25, 2009
Agreement Number:	08HQAG0035
Project title:	Documenting Best Practices in Geospatial SOA through the Development of a Wetlands Permitting Solution
Organization:	Image Matters LLC 201 Loudoun Street, SW Leesburg, VA 20175 www.imagemattersllc.com
Project Leader:	John Davidson V. 703.669.5510 F. 703.669.5515 johnd@imagemattersllc.com
Collaborating Organizations:	Environmental Protection Agency Tod Dabolt, Director, Project Management Office, Office of Water Voice: 202 564-1450 FAX: 202 564-0500 Dabolt.Thomas@epamail.epa.gov http://www.epa.gov/ow/ U.S. Fish & Wildlife Service Bill Wilen, Senior Biologist, National Wetlands Inventory Voice: (703) 358-2161 FAX: (703) 358-2232 Bill_Wilen@fws.gov http://www.fws.gov/nwi/

Executive Summary

The project was designed to fulfill two primary goals: (1) refine SOA best practices through implementation experience, by developing and integrating Web services to enhance the capabilities of DARTER, a SOA-based tool that facilitates EPA and ACE's Clean Water Act (CWA) Section 404 missions, in a manner that supports the USFWS mission of National Wetland Inventory (NWI) data provision, and (2) produce guidance materials that document the best practices and lessons learned about the SOA process for business process modeling, use case and requirements elucidation, service design, development, implementation, integration, testing and operational deployment. Generic Web Processing Services (WPS) performing "intersection" and "proximity" geo-analytical processes through an OGC-compliant interface were built, tested, and packaged for public distribution using the documented SOA process. A client application – the "JD Analyzer", was also developed to demonstrate the use of the "JDA WPS" in a stand-alone application which performed key geospatial analyses to support the jurisdictional determination (JD) of wetlands in the Section 404 permitting process. The JD Analyzer utilized the WPS to request and consume USGS National Hydrographic Dataset (NHD) and NWI features via OGC WFS. The documented SOA process was presented, along with live demonstrations of the JD Analyzer, at several federal conferences and workgroup meetings.

Project Narrative

Collaboration Environment and Best Practices Documentation

After the project kick-off, Image Matters established a wiki-type collaborative environment (http://cap.imagemattersllc.com/confluence/) for FGDC and the other two grant recipients, Indiana University and CubeWerx. This "Confluence" site was used to develop and archive project artifacts including "common terms" used in SOA development, guidelines for the development approach and SOA modeling environment, and examples of documentation from each of the three projects. The content and structure of the "Confluence" site was made available to federal agencies and others who seek guidance on how to build a SOA-based web service.

In addition to the Dashboard and project-specific pages, these primary site pages are populated:

- Common Terms – a list of commonly-used terms, their definitions, and source information
- Use Cases – the set of use cases for each of the three projects
- Templates – documents that serve as templates for development efforts in other projects
- Best Practices – guidance materials for federal agencies looking to develop SOA-based web services

Service and Application Overview

Image Matters developed and documented a SOA development process (discussed below) that it then followed in building and testing a generic Open Geospatial Consortium (OGC)-compliant Web Processing Service. The WPS performs "intersection" and "proximity" geo-analytical processes on two user-defined feature datasets accessed via OGC Web Feature Services (WFS). This WPS was originally intended to be utilized by the Data on Aquatic Resources Tracking for Effective Regulation (DARTER) system, which is targeted to be the next generation software framework for EPA's wetland analysts and permitting managers. The targeted data sources for the WPS were an instance of the US Geological Survey (USGS) National Hydrographic Dataset (NHD) accessed through the USGS Framework Web Feature Services, and an instance of the US Fish & Wildlife Service (FWS) National Wetland Inventory (NWI) database accessed through a new WFS hosted by FWS.

Because the DARTER development schedule precluded integration, we developed client software that accessed the WPS spatial operations to support Clean Water Act (CWA) Section 404 jurisdictional determination (JD) of wetlands, based on intersection of and proximity to NHD streams – known "waters of the US". Because USFWS was unable to provide a new WFS, the "JD Analyzer" application accessed NWI features from various temporary WFS services for use in the "JDA WPS"

Outcomes

The project fulfilled its primary two goals, generation of best practices documentation for the SOA process, and delivery of an extensible WPS that supported jurisdictional determination as part of the CWA Section 404 permitting process. The specific Lessons Learned, and documentation of Image Matters' development process – shared experience for agencies looking to develop SOA-based web services, have been all archived as attachment in their "Project/Sponsor Spaces" page. We understand that these documents, the primary products of our work, will also be made available on the FGDC website. The titles of this set of (PDF) documents, the content of which are described in the Best Practices section of this report (below), are as follows:

- ***Image Matters - EPA JDA - DARTER Geoanalytical Requirements and WPS Overview***
- ***Image Matters - EPA JDA - Use Cases***
- ***Image Matters - EPA JDA - Design***
- ***Image Matters - EPA JDA - Installation and Configuration***
- ***Image Matters - EPA JDA - Best Practices in SOA-based Geospatial Services***

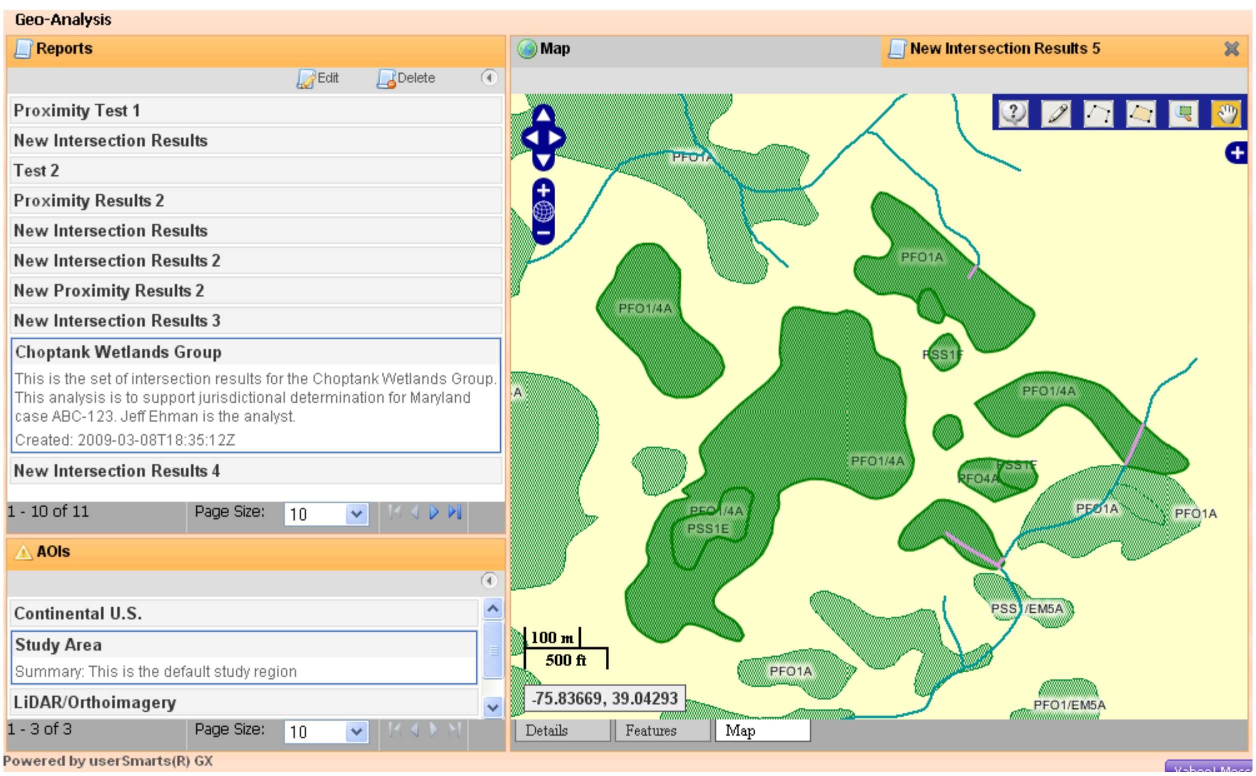
In addition, the slides from our final presentation on the project were delivered:

- ***Image Matters - EPA JDA - Geospatial SOA Workshop 9June09***

While we regret not being able to integrate with the DARTER application, we believe that the JD Analyzer did provide a demonstration of the potential for use of the JDA WPS in DARTER. Moreover, it was only through the flexibility of the SOA development process were we able to adjust mid-stream and deliver a functional application that demonstrated the original intended use for 404 permitting.

Furthermore, our application facilitated more rigorous testing of the JDA WPS, and allowed future DARTER users an opportunity to refine their functional requirements for the eventual application. These requirements included not only access to feature-level data hosted close to the authoritative source or data maintainer, but also Web-based tools capable of performing spatial operations historically limited to desktop GIS applications, and the integration of vital contextual information such as LiDAR. Finally, we were able to demonstrate the application and explaining its component services and architecture, as well as the best practices for the SOA process and our lessons learned, to as wide an audience as possible. Being able to do so openly, not behind the EPA firewall, was enabled by the in-house provision of the JD Analyzer. In retrospect, aside from the comprehensive set of documents for those ready to follow this architectural path, our largest contribution was to raise awareness that highly-functional SOA solutions, built though open-source components, can enable cost-effective geospatial SaaS.

A screen shot of the application is presented below:



<u>Outreach</u>

Our development and SOA process approach and that of our two 2008 CAP Category 2 co-awardees, and our individual projects used as case studies, were presented at the GIScience 2008 conference in Park City, UT on September 23rd, 2008, and also at the Geospatial Service-Oriented Architecture: Best Practices Workshop in Washington, DC on June 9, 2009. In addition, Image Matters presented these materials during a WebEx presentation to a USFWS and USGS audience, in person at our Final Project

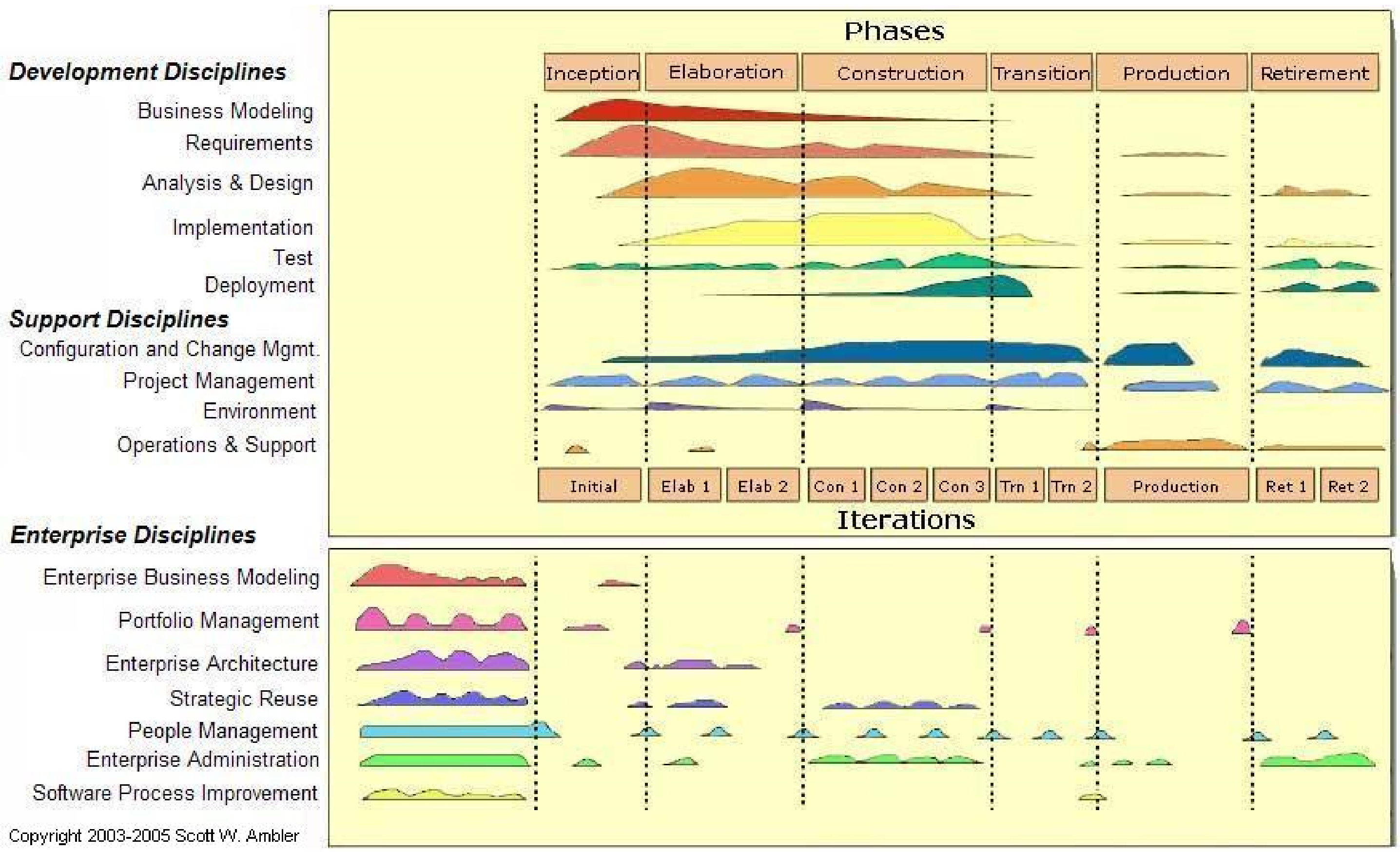
Meeting in DC at EPA HQ, as well as to a broader EPA audience at their GIS Workgroup, Spring 2009 Meeting in Crystal City, VA. After its development, Image Matters gave live demonstrations of the JD Analyst application in each presentation.

SOA Definitions

Image Matters generated an initial list of 33 commonly-used terms (presented below) and their definitions and posted these to the "Confluence" project collaboration site for consideration by FGDC and the two other CAP Category 2 project teams. Ultimately, 39 terms were agreed-upon. As appropriate, the terms and definitions were taken from authoritative sources, with the references to those sources are included in the listings. These terms, and their definitions and sources, are presented in Appendix A of this document.

Requirements and Process Definition

Image Matters is following the general software development life cycle pattern agreed upon by the three awardees: model process and elucidate requirements, design and develop, implement and test, deploy and monitor. We proposed the following diagram as a general model to follow during the projects:

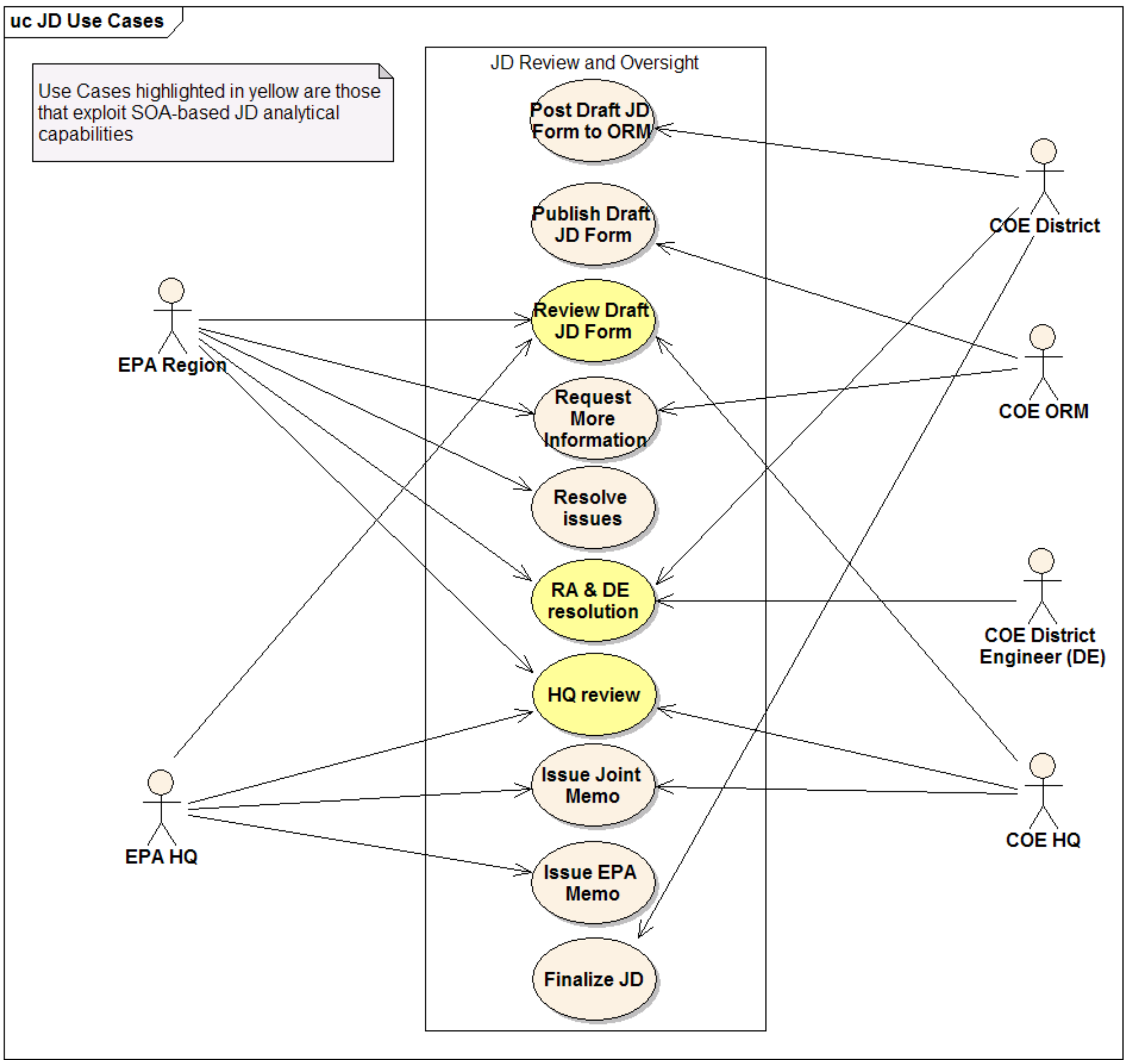


The first step has been broken down further into the following components:

1) Document Business Process
2) Create Concept of Operations
3) Develop Detailed Use Cases
4) Generate Technical Requirements

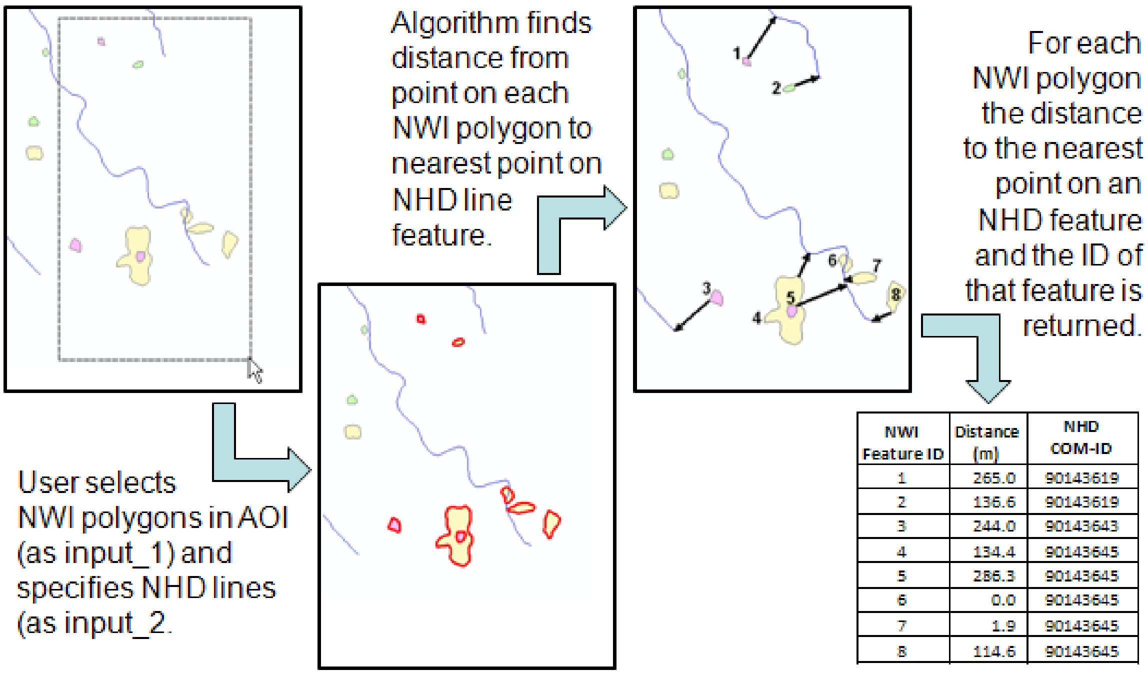
Our requirements gathering phase started, albeit informally, during the proposal formulation stage. At that time we were made aware of a specific geo-analytical processing need that would best be addressed with a SOA-based web service. After researching the definition of a "jurisdictional wetland" and confirming the correct inputs and outputs, we proposed our solution which was met favorably by the EPA and FWS.

Upon receiving funding, we requested, received, and examined the multi-agency business process needed to make Jurisdictional Determinations (JD), as prepared for the DARTER development. EPA's "Screening" of US Army Corp of Engineers (COE) preliminary decision regarding the permit was step 13 of a 24-step process, and is represented in the below as the "Review Draft JD Form" step in the process:

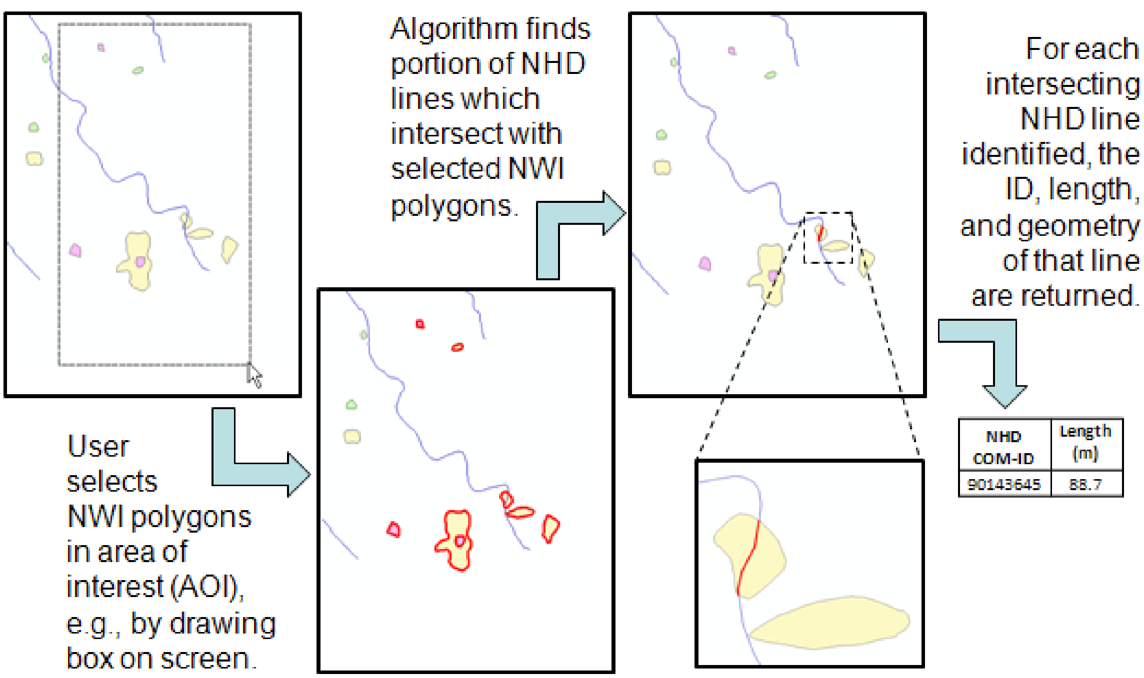


So although the business process placed the Screening Step, in which all EPA geospatial analysis took place, in the context of the larger business process, the process models were at too high of a level to provide much information about the specifics of geoprocessing required by EPA's Analysts for Jurisdictional Determination (JD). It was clear the specific step-by-step process involving geospatial data and geo-analytical processing required by the EPA analysts needed to be captured. We determined that this level of detail was best documented in a use case format.

As part of our first meeting with all involved parties, including program managers, data specialists, and wetland analysts from EPA, FWS, the US Army Corp of Engineers (COE) and consultants leading the DARTER development effort, we spent time gathering specific user requirements. We validated our proposed geoprocesses, and received input on necessary user controls, formats for presenting results, and desired contextual layers. We used the following storyboards to communicate our understanding of the necessary geoprocessing functionality back to the ultimate client – the DARTER users. The first of these represents a proximity operation, the second an intersection, both with NWI wetlands and NHD streams as the inputs.



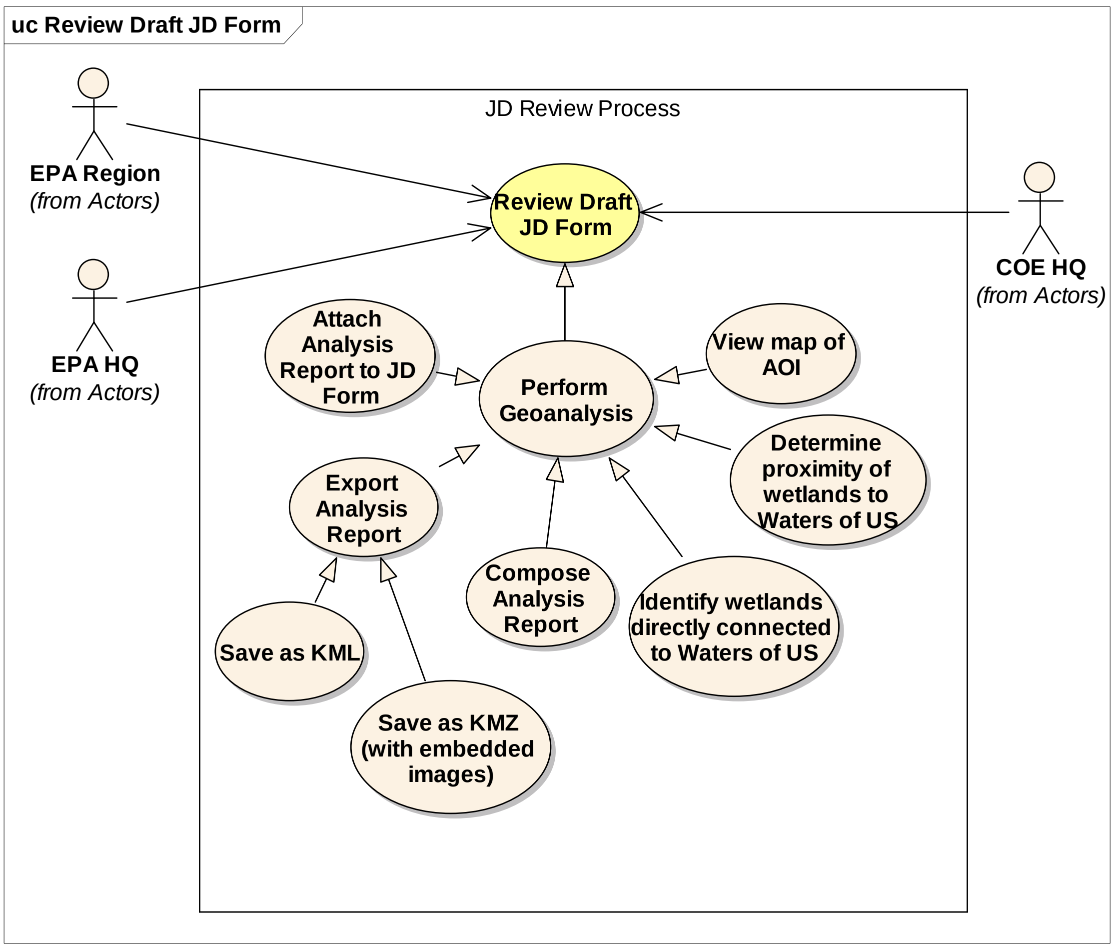
NWI Feature ID	Distance (m)	NHD COM-ID
1	265.0	90143619
2	136.6	90143619
3	244.0	90143643
4	134.4	90143645
5	286.3	90143645
6	0.0	90143645
7	1.9	90143645
8	114.6	90143645



NHD COM-ID	Length (m)
90143645	88.7

The information gathered at that initial meeting provided us sufficient information to develop detailed use cases for the specific step-by-step process involving geospatial data and geo-analytical processing required by the EPA analysts. These use cases, ultimately like the storyboards, were used for three purposes: 1) to verify our understanding of our client's (DARTER users) functional needs, 2) to help communicate functional requirements (fully documented in our Design document) to our developers and to serve as the basis for designing software interactions, and 3) for designing formal unit and final software testing.

Based on the input received at this initial meetings, and subsequent conference calls and document review, we agreed on a number of use cases, which are fully documented in our Use Cases document. All of the individual, sub-use cases are represented in the following diagram:



Currently, the constituent user base consists solely of EPA Wetland Analysts. However, at our requirements meeting, the COE expressed an interest in enabling their wetland analysts to access the WPS-based geoprocessing services, which are a good fit with the SOA-based design of the ORM-II (OMBIL Regulatory Module). [Note that the EPA's DARTER system is a part of the larger ORM-II system, and the two software frameworks exchange information as a part of the multi-agency Jurisdictional Determination workflow.] In addition, it is quite possible that States, Tribes, an even municipalities would find the geoprocessing services useful for landscape-level or site-specific assessment of wetland connectivity to water features. Given that the WPS-based services were designed to be generic, custom applications could be developed that utilized inputs other than NHD and NWI, e.g., high-resolution wetland delineations generated from field GPS work and stood up via WFS. Moreover, assuming adequate access and WFS-based data inputs, users faced with any intersection and/or proximity geoprocessing requirement involving web services could utilize these services.

SOA Deployment and Acceptance

Image Matters faced several unanticipated obstacles and changes in scope in the first half of the project. It turned out that neither the target DARTER application, nor the targeted NWI WFS were a part of the final solution, and that a considerable amount of additional data was required by the end users for effective use of the analytical tools.

The first roadblock was finding a willing host for the FWS's NWI WFS. The planned deployment at an EPA server in Madison (WI) fell through due to internal EPA resource allocations. The alternative, for FWS to host the FWS from their own server, was untenable because of the perceived burden that it would place on staff already stretched thin. Image Matters responded by standing up NWI data behind a WFS on one of their own servers, a temporary solution that allowed development to progress. Soon thereafter the EPA contractor for the DARTER application, the Indus Corporation, provided a suitable WFS to serve the EPA's NWI database instance. Ultimately, we configured the application to the OpenGEO / IU "Cloud" WFS.

The second impediment was that DARTER was behind schedule. About halfway through our project duration, it became obvious that the DARTER development schedule was not far enough along to permit integration with our services within the timeframe of our project. At this point we decided to quickly build our own client software in order to demonstrate, receive feedback on, and further refine the JDA WPS, as well as provide input to the final toolset requirements that will eventually be a part of the DARTER system.

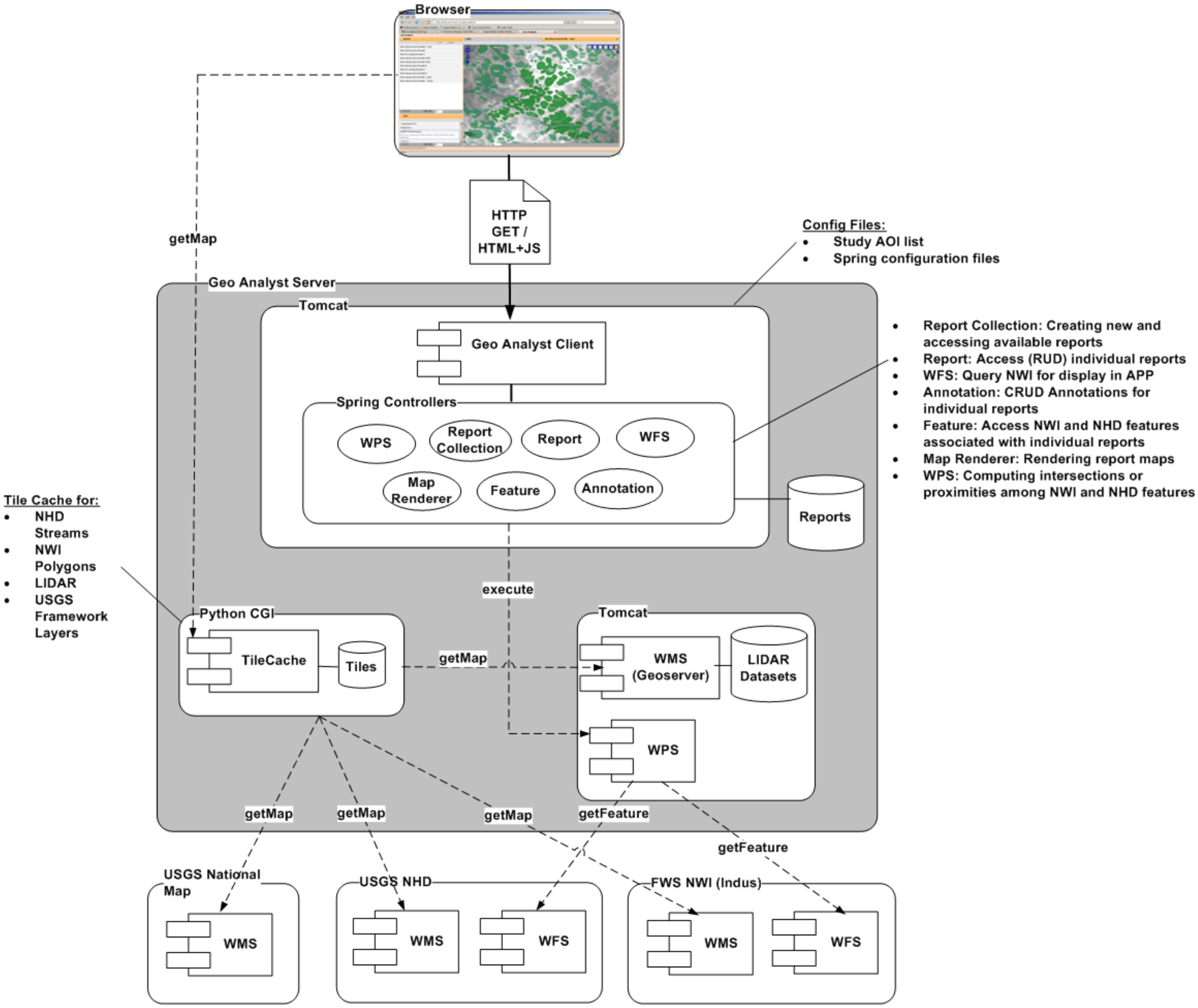
Image Matters responded by developing the "JD Analyzer", a stand-alone application which demonstrated the use of the WPS by performed key geospatial analyses to support the jurisdictional determination (JD) of wetlands in the Section 404 permitting process. The JD Analyzer utilized the WPS to request and consume USGS NHD features from the "Framework WFS", and NWI features from a temporary NWI WFS stood up in-house. The query results of the overlay and proximity analyses are presented to the user in both map and tabular form, and can be saved for future use and/or sharing between application users.

To be able to develop the JD Analyzer within the project budget and with a reasonable no-cost time extension, we relied heavily on our existing userSmarts®GX middleware code base, in-house development and production environments, available hardware resources, and experience with developing web applications in a standards-based SOA framework from open-source software components, using our Rapid Solution Engineering approach

Recognizing the need for contextual data to provide meaning to the NHD and NWI data, and moreover to provide a realistic and thus useful application, several OGC Web Map Services (WMS) were established to made this "background" data accessible. Thus, in addition to the WMS accessed for display purposes of the main features (USFWS NWI WMS and USGS NHD WMS), WMS were stood up in-house by Image Matters for LiDAR-derived elevation data and local-scale orthoimagery available for our "test site" – the Choptank watershed in northeastern Maryland.

Although we were provided with no quantitative quality-of-service requirements from our EPA and USFWS project partners, we knew that the application had to be fast enough so that there were no noticeable delays in executing any step in the use case, especially during the spatial analysis operations (WPS-based intersection and proximity) or in the rendering of maps. However, during testing of early versions of the JD Analyzer, it became apparent that there were two main performance issues to be addressed. The first, the speed of the USGS NHD WMS, was outside of our control. So, we focused on the second, the speed of the in-house elevation and orthoimagery WMS. An open-source tiling service was the obvious solution and was quickly implemented on Image Matters' servers.

The final architecture of the system is presented below:



Best Practices

Specific Documentation

Image Matters and the other two CAP Category 2 project teams, in conjunction with FGDC staff, developed a set of best practices and general patterns to follow during geospatial service and dependent application development within a SOA context. This documentation was aggregated in the wiki stood up for the project teams. Image Matters followed the general process for development, and documented each of its main steps in the artifacts mentioned in the Narrative section (above).

The ***Image Matters - EPA JDA - Best Practices in SOA-based Geospatial Services*** document provides the following components:

1. A summary of our recommended general patterns and procedures for developing geospatial services in a SOA
2. An example of the implementation of these patterns and practices, and pitfalls (to avoid) using the development of the JDA WPS and the JD Analyzer application as a case study.
3. Our “lessons learned” and best practices that emerged through the development process.

Standards and specifications applied

The table below lists the standards implemented as part of our WPS and JD Analyzer application.

Standards Body	Standard
OGC	Web Map Server, ISO 19128
OGC	Web Map Context
OGC	Web Feature Service, ISO 19142
OGC	GML 3.1.1, ISO 19136
OGC	Filter Encoding, ISO 19143 (Filter)
OGC	Web Processing Service 1.0.0
ANSI INCITS	Framework Data Standards (Draft)
OGC	OWS Common Implementation Specification 0.2.0
Sun Microsystems	Java™ 2 Platform Standard Edition API Specification 1.4.2
W3C	HTML/HTTP

Of special note here is Section 6 of our “***Best Practices in SOA-based Geospatial***” document, which contains two “lessons learned” sections regarding WFS, and their selection and use as WPS inputs:

6.1. Lessons learned about Geospatial Data accessed via Open Web Services

6.2. Managing Differences among Web Feature Services

Monitoring and testing performed on the service, and optimization applied

Image Matters performed unit tests on the WPS and on various components of the JD Analyzer application, followed by System Integration Testing (SIT), and finally User Acceptance Testing (UAT). As mentioned above, although we did not establish explicit, quantitative quality-of-service requirements with our EPA and USFWS project partners, we did discover major performance issues during the SIT phase of testing. These performance bottlenecks, all WMS-related, were addressed by establishing an open-source tiling service. A second iteration of SIT procedures confirmed the improved performance.

Service patterns applicable to other services

The OGC WPS defines a very basic yet powerful and extensible set of three operations: get capabilities, describe, and execute. This pattern maximizes interoperability while allowing a range of implementation – from very rudimentary and generic to specialized and sophisticated. The JDA implementation of WPS falls in-between this range. As described below, the JDA WPS implements two very essential spatial operators: spatial intersection and proximity filtering. The JDA WPS, however, also implements very specialized reporting capability based on the results of the intersection or proximity analysis of the NHD (hydrography) and NWI (wetlands) geometry and tabulation of feature properties defined by the schema of these datasets. The request to execute an intersection or proximity analysis is simple, requiring no specialized knowledge of the NHD or NWI source data. The results returned, however, are specialized by the choice of operation (intersect or proximity) and the properties of the NHD and NWI sources. It is difficult to implement a generic WPS that can handle spatial intersection and proximity analysis on any possible pairing of vector datasets and return meaningful results. The WPS specification provides for a "compromise" between simplicity and expressiveness – it allows a simple and standardized way to invoke spatial analysis while allowing the flexibility to return meaningful and easily processable results for very specialized business applications.

Common data structures, and Selection and performance of service binding patterns

As presented in our ***Image Matters - EPA JDA - Design*** document, The Atom Syndication Format (ATOM) was used as the representation of data being sent between the client and server-side components of the Wetlands JD Analyzer application. The data is serialized as JSON (JavaScript Object Notation) to minimize costs associated with parsing XML in the web browser. The Atom Publishing Protocol uses standard HTTP methods to facilitate access to or modification of resources provided by an APP-supporting server.

- GET is used to retrieve a specific representation of a resource
- POST is used to create a new resource through submission of representations of the to-be-created resource to the collection to which the new resource will be added
- PUT is used to update a resource using a modified representation of the resource
- DELETE is used to delete a resource

This technique of using HTTP methods to access and modify resources is known as being "RESTful". The server-side components of Wetlands JD Analyzer application provide several Java classes which act as RESTful servlets using Atom to process requests for the various types of objects within the application:

- WfsResource – provides access to the configured WFS – the NWI wetlands WFS in the default configuration
    - HTTP-GET – querying the WFS using the supplied querystring parameters
- ReportResource – provides access for retrieving, updating, and deleting analysis reports
    - HTTP-GET – retrieving a single report
    - HTTP-PUT – updating a single report
    - HTTP-DELETE – deleting a single report
- ReportsCollectionResource – provides access for retrieving lists of analysis reports and creating new analysis reports
    - HTTP-GET – retrieving a list of reports
    - HTTP-POST – creating a new single report
- AnnotationResource – provides access for retrieving, updating, and deleting report annotations
    - HTTP-GET – retrieving a single annotation
    - HTTP-PUT – updating a single annotation
    - HTTP-DELETE – deleting a single annotation
- AnnotationsCollectionResource – provides access for retrieving lists of report annotations and creating new annotations
    - HTTP-GET – retrieving a list of annotations
    - HTTP-POST – creating a new single annotation
- FeatureResource – provides access for retrieving lists of features (wetlands and streams) associated with a report
    - HTTP-GET – retrieving a list of features

Supported data structures

The WPS processes GML representations of the NWI and NHD datasets (as returned by their respective WFS implementations). The data structures (and schema) for NWI and NHD data are defined externally and independently of the JDA WPS implementation. GML provides the common language in which to represent geospatial features and their properties (geometry and other attributes). The WPS implementation need only know how to parse and process GML sufficiently to identify feature objects, extract values for specific properties, and perform coordinate geometry computations for intersection and proximity analysis. Beyond knowledge of the general model for feature representations defined by GML and the specific structure of properties defined by the NWI and NHD datasets, no specialized data structures or data transformations are required.

Methods taken to improve the scalability and extensibility of the WPS

There are at least two aspects to the JDA WPS's extensibility:

1. the capacity to easily swap WFS as NHD and NWI inputs, and
2. implement the WPS in entirely new application domains which require completely different feature types as inputs

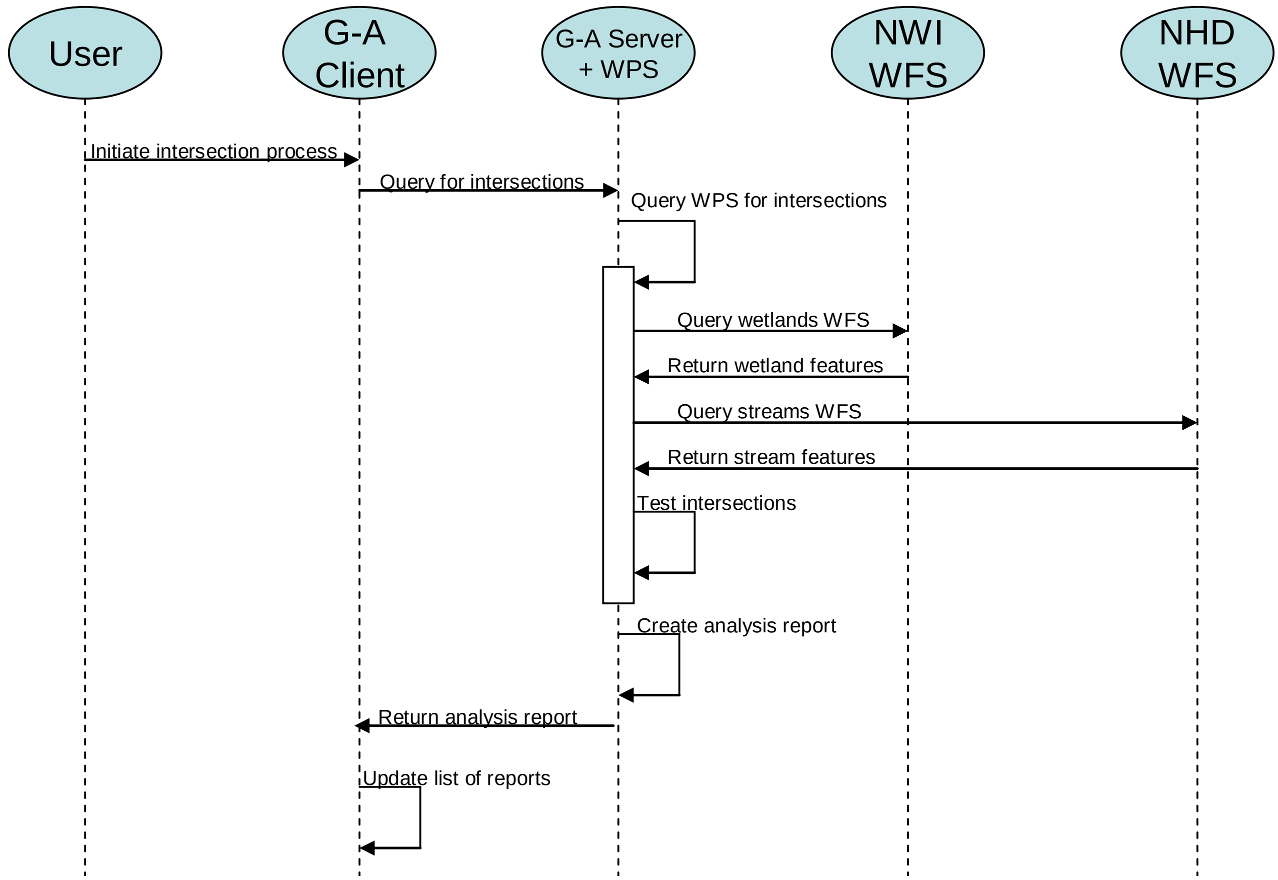
The former extension of the JDA WPS was rigorously tested, out of necessity, as we switched NWI inputs on four separate occasions to distinctly different WFS's. Working with different WFS services hosting "the same data" presents a challenge in that each implementation almost invariably models the data differently. As built, the JD Analyzer required that the data must be normalized against a common schema prior to WPS processing.

Although we would expect rapid and successful configuration to other pair of polygon and line dataset served through WFS, to date, our WPS has not been tested for use with other inputs other than the NHD and NWI data.

As we were required to shift resources to the development of the JD Analyzer application for testing and demonstrating the WPS, we performed only limited scalability testing. We would have liked to have been able to deploy on larger servers, and possibly in a cloud environment, to test queries involving large numbers (1000 to 10,000) of wetland polygon features. However, our cursory tests suggested that the rate limiting step in the system would not have been with the processing, but rather in the generation of voluminous GML payloads by the WFS resulting from GET requests.

Service composition, chaining, and orchestration in the JD Analyzer application

As evident in the final architectural diagram (above, page 9), and sequence diagrams associated with testing for intersections (below) , essential for clarifying communications among system entities), there were considerable map, feature, and processing service orchestration necessary for successful operation of the JD Analyzer application.



Employing standard interface specifications in the design phase was essential in ensuring that the system components would interoperate. As noted, OGC standards were heavily relied upon and were essential in this endeavor. Web mapping (WMS), feature (WFS), and processing (WPS) service interfaces proved to be easily plugged into the system as it emerged through the development process.

Emergent technical requirements for future deployments

The JDA WPS achieves normalization through implementations of WFS clients that translate from the actual schema to the normalized schema during feature requests. Future versions could include generic clients that make use of configured mappings, allowing normalization without requiring new Java code – a rules-based or configurable way to support generic computation of intersection and proximity analysis on any pairing of vector datasets. Additional requirements include support for multiple selectable or configurable customized reports generated by the JDA WPS. In this way the JDA WPS can become more generally useful for interoperable computation of intersection and proximity on a wider range of datasets but also a wider range of analytical problems required by applications.

Other considerations

Although the service was not re-deployed to host sites apart from Image Matters server environment, we did successfully re-deploy internally on different servers. Hosting requirements are relatively minor in terms of software, and once Java Runtime Environment (JRE, version 1.5 or newer) and Apache Tomcat application server (version 5.5 or newer) are on the server, the installation is a simple 3 step process. Therefore we would expect that re-deployment to other host sites (government, commercial, ISP) would be relatively easy. One obvious advantage of deployment at the government agency would be the capability to access WFS services that are behind the government firewall.

No service level agreements (SLA) emerged from the deployment of this SOA solution. Without the reliability provided by a SLA, it is doubtful that federal agencies will have the confidence to entrust other agencies or organizations with services that are critical to their business. This project helped to establish the need for a stable WFS near the authoritative source of the NWI database (at the Madison USFWS office and USGS support center). The development of the demonstration OpenGEO / IU "Cloud" WFS was due in part as a response to this need.

As it became clear that the JDA WPS was only going to be part of a demonstration, rather than an integrated component in an operational system, we did not devote resources to identifying or assessing measures to assure service availability and reliability (redundancy, failover approach, hosting requirements, synchronization approach, service recovery time).

Project management

At this point in time, no future activities are planned with our project partners (USFWS, EPA, or ACE) regarding follow-on work associated with this project. However, given our widespread dissemination of project materials and demonstration of the JD Analyzer application, we remain hopeful that someone will need these services. If this were the case, we would recommend the following tasks:

1. Identify an operational use of the WPS
    a. Better integrate with the DARTER development schedule so as to ensure the integration of the JDA WPS, or the entire JD Analyzer. As access to the USGS NHD feature-level data appears to the rate-limiting step in the process, we would recommend looking for alternative sources of the same information and/or ways to speed up access to the existing server.
    b. Find an alternative business need for our WPS, likely different than jurisdictional determination of wetlands and thus quite possibly using feature datasets different than NHD and NWI.
2. Encourage USFWS to host a NWI WFS, either at Madison or in the cloud, with an SLA in place with EPA and/or other agencies who would benefit from this service.
3. Find alternative sources for more ubiquitous contextual information via WMS. At present we are accessing local-scale elevation data and orthophotography for a limited area via WMS from an Image Matters Server.
4. If the entire JD Analyzer application were to be used by EPA, we would need to undergo more formal user testing and usability analysis.
5. In preparing for re-deployment and acceptance we would need to identify the measures to assure service availability and reliability (as listed above).
6. In preparing for re-deployment, training materials would need to be prepared, including the content of the Installation and Configuration guide.
7. A mechanism for making available and tracking the use of technical support to the administrators of the deployed service and/or application.

Feedback on Cooperative Agreements Program

We strongly support FGDC's CAP for promoting and implementing the NSDI. From our perspective, that of a small business with considerable experience in developing and implementing geospatial interoperability standards, the CAP provides a vehicle for engaging government customers that might otherwise be reluctant to try new approaches in developing geospatial solutions. The grants allow us to build on our experience, and our technologies, an investment which in turn feeds back into the NSDI in many different ways. For the government partner, the projects provide exposure to the NDSI resources available to them, greater experience in employing interoperability standards within their own enterprise, and hopefully a realization of improved efficiencies and an eventual return on their matching fund investment.

More specifically, for this particular project, the following comments come to mind. First, the amount of monetary assistance that we received through the cooperative agreement was sufficient in meeting our project goals. In retrospect, a greater awareness of the uncertainties associated with the DARTER development timeline would have been helpful in setting our internal project schedule and resource allocation. Delaying the project start date for a number of months might have allowed eventual integration in an operational setting – always a major goal for a small business. While we believe that we were able to adapt to this change, and benefitted in other ways from the necessary development of client software in which our WPS could be tested and demonstrated, we were still disappointed about that shortcoming. The approval for this change in scope, as well as the project no-cost extension that we granted was helpful in recovering from this unforeseen condition.

Regarding the integration of the three CAP Category 2 projects, we felt the coordination early in the project periods was greater, waning later towards project completion. In particular, we believe that the SOA best practices documentation could have been better coordinated between the 3 projects, thus offering potential federal service developers a more consolidated view of a recommended approach.

Appendix A. Common Terms for the SOA Process

Atom Publishing Protocol
A REST-style protocol and service specification for using ATOM as an information service about collections of Atom and other feeds. See http://tools.ietf.org/html/rfc5023.

Atom Syndication Format
An XML-based message syndication format, commonly used to convey time-dependent news items to news readers. Atom is formally defined using XML Infoset and is an IETF proposed standard. See http://tools.ietf.org/html/rfc4287.

blog
A web log or diary. Blogs are syndicated using formats such as RSS or Atom in addition to their standard HTML formats. Many blog software and service providers have programming interfaces for blog writing that allow posting from a variety of client devices.

bounding box
1. portion of a coordinate space that lies between a lower bound and an upper bound in each dimension of a coordinate reference system
2. a set of 2, 4, 6 or 8 numbers indicating the upper and lower bounds of an interval (1D), rectangle (2D), parallelpiped (3D), or hypercube along each axis of a given CRS (http://www.opengeospatial.org/ogc/glossary)

capabilities XML
Service-level metadata [encoded in XML]describing the operations and content available at a service service metadata. (http://www.opengeospatial.org/ogc/glossary)

capability
A real-world effect that a service provider is able to provide to a service consumer. (http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf)

client
A software component that can invoke an operation performed by a server. (http://www.opengeospatial.org/ogc/glossary)

geographic information
information concerning phenomena implicitly or explicitly associated with a location relative to the Earth [ISO 19128 draft]

GEO Microformat
An XHTML extension that encodes latitude and longitude.

GeoRSS
An extension RSS and Atom that allows geospatial information (such as the geo-location) to be embedded in a syndication feed. GeoRSS is an Open Geospatial Consortium standard and is supported by tools such as Google Maps.

interaction
The activity involved in making using of a capability offered, usually across an ownership boundary, in order to achieve a particular desired real-world effect. (http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf)

interface
1. A named set of operations that characterize the behaviour of an entity.
2. An implementation of operations including the syntax of the interaction for a given distributed computing technology.
3. A shared boundary between two functional entities.
4. An established ordering of parameters (with specific names and data types) and instructions (with specific names and functions) that enables one software component to exchange data and instructions with another software component. (http://www.opengeospatial.org/ogc/glossary)

interoperability
capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units [ISO 2382-1]

literal
any process input or output whose value can be represented in a character string, supplemented by metadata as needed

literal (XML encoding)
any process input or output whose value can be represented in a xsd:string supplemented by XML attributes as needed NOTE A literal process input or output can be a character string, integer, general number, URI, measure, etc.

map
A two-dimensional visual portrayal of geospatial data. A map is not the data itself. (http://www.opengeospatial.org/ogc/glossary)

microformats
Conventional extensions of XHTML using <div> and <span> tags to semantically describe small piece of metadata. Microformat rendering as HTML can be governed with CSS, and microformats can be associated with JavaScript tools.

operation
1. A single step performed by a computer in the execution of a program, or, in the context of object-oriented programming. (http://www.opengeospatial.org/ogc/glossary)
2. Specification of an interaction that can be requested from an object to effect behavior. [ISO 19119]

output
result returned by a process

parameter
variable whose name and value are included in an operation request or response

process
model or calculation that is made available at a service instance

request
Invocation of an operation by a client. (http://www.opengeospatial.org/ogc/glossary)

response

Result of an operation returned from a server to a client. (http://www.opengeospatial.org/ogc/glossary)

REST
Representational State Transfer, an alternative formulation of Web services. In place of WSDL, REST systems use the HTTP verbs such as GET, POST, PUT, DELETE for all operations. Operations are performed on URLs, which typically respond with XML messages. XML message formats can be SOAP, RSS, Atom, etc. Other message formats such as JSON can be used. See http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm.

RSS
A popular XML-based message syndication format, commonly used to convey time-dependent news items to news readers. RSS has several versions.

service instance
a particular instance of a service [ISO 19119 edited]

service

1. The means by which the needs of a consumer are brought together with the capabilities of a provider. (http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf)
2. distinct part of the functionality that is provided by an entity through interfaces [ISO 19119]
3. A computation performed by a software entity on one side of an interface in response to a request made by a software entity on the other side of the interface. [ISO 19119]
4. A collection of operations, accessible through an interface, that allows a user to evoke a behavior of value to the user. [ISO 19119]
5. capability which a service provider entity makes available to a service user entity at the interface between those entities [ISO 19104 terms repository]

service chain
A sequence of services where, for each adjacent pair of services, occurrence of the first action is necessary for the occurrence of the second action. [ISO 19119]

service consumer
An entity which seeks to satisfy a particular need through the use capabilities offered by means of a service. (http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf)

service description
The information needed in order to use, or consider using, a service. (http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf)

service interface
The means by which the underlying capabilities of a service are accessed. (http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf)

service metadata
metadata describing the operations and geographic information available at a server [ISO 19128 draft]

Service Oriented Architecture (SOA)

1. A computer systems architectural style for creating and using business processes, packaged as services, throughout their lifecycle. SOA also defines and provisions the IT infrastructure to allow different applications to exchange data and participate in business processes. These functions are loosely coupled with the operating systems and programming languages underlying the applications. SOA separates functions into distinct units (services), which can be distributed over a network and can be combined and reused to create business applications.These services communicate with each other by passing data from one service to another, or by coordinating an activity between two or more services. (http://en.wikipedia.org/wiki/Service-oriented_architecture)
2. A paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. It provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects consistent with measurable preconditions and expectations. (OASIS, http://www.oasis-open.org)
3. A system for linking resources on demand. In an SOA, resources are made available to other participants in the network as independent services (http://www.ianywhere.com/developer/rfid_anywhere/glossary.html)
4. Defines how two or more entities interact in such a way as to enable one entity to perform a unit of work on behalf of another entity. The unit of work is referred to as a service, and the service interactions are defined using a well-defined description language. (http://www.dunelm.com/resources/glossary.html)
5. A software design that integrates business functions. Users are able to decide the information which is to be shared between the functions. SOA is therefore more flexible and more loosely coupled than ERP and generally more suitable for service rather than manufacturing companies. (http://www.bpic.co.uk/jargon.htm)
6. A paradigm for design, development, deployment and management of a loosely coupled business application infrastructure. (http://keyintegrity.com/en/)
7. An architecture, the aim of which is to achieve a loose connection between integrated systems. From a common public Danish perspective, the integration of IT systems across public and private organisations is part of the vision of digital administration. (http://www.capevo.com/wm139851)
8. A service-oriented architecture is essentially a collection of services. These services communicate with one another. The communication can involve either simple data passing or it can involve two or more services coordinating some activity. (http://www.reactivity.com/soa/glossary.html)
9. expresses a business-driven approach to software architecture that supports integrating the business as a set of linked, repeatable business tasks, or "services". SOA is usually based on a set of Web services standards. (http://www.efforts-project.org)

service provider
An entity (person or organization) that offers the use of capabilities by means of a service. (http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf)

SOAP (Simple Object Access Protocol)
An XML network message format commonly used to convey instructions and responses between services in an SOA. SOAP messages are commonly transported using HTTP. See http://www.w3.org/TR/soap/.

version
version of an Implementation Specification (document) and XML Schemas to which the requested operation conforms. NOTE An OWS Implementation Specification version may specify XML Schemas against which an XML encoded operation request or response must conform and should be validated.

Web Service Architecture (WSA)
A form of SOA for network-based machine-to-machine interaction that is based on World Wide Web Consortium standards such as HTTP, XML, SOAP, WSDL, etc. See http://www.w3.org/TR/ws-arch/.

workflow
automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules

WSDL (Web Services Description Language)
An XML language for describing network service programming interfaces. WSDL is typically used by a Web service to describe how to construct SOAP messages that it can consume and the format of its responses. See http://www.w3.org/TR/wsdl.