Web Services Requirements Subgroup Report

to the Minnesota Geospatial Commons Committee
September 15, 2010

Subgroup members:

- Jessica Fendos (DEED)
- Susanne Maeder (MnGeo)
- Matt McGuire (MetroGIS)
- Hal Watson (DNR)

The Web Services Subgroup has been assigned the following task areas:

1. Web Services:
    a. Clarify what comprises a comprehensive documentation of a web service
    b. Determine key characteristics needed to achieve "trust" in a web service
2. Broker
    a. Define the roles of *broker* (machine and human) and *service provider*. Document how these roles relate to the concept of trust in a web service (1b above).
    b. Define options for, and recommended functions of broker and how it interfaces with service provider and the application client.

As of this writing, the group has been working on the tasks under item 1. "Web Services."

Comprehensive Web Service Documentation

Minimum standards recommendation:

We recommend to the group that "ISO 19139 / 19119 Web Services" be used as the standard for basic, minimum level documentation of web services on the test-bed implementation of the Minnesota GeoCommons Portal. As the test-bed is evaluated, this standard needs to be evaluated as well to make sure it supports the evolving project requirements.

Discussion:

There are any number of different ways to document web services. The issue is complicated by the fact that there are competing documentation standards, and different audiences for this documentation. We struggled for some time with the simple fact that implementing a particular web service documentation standard does not necessarily solve the problem of providing the right information to a particular type of "user" in the proper context.

Nonetheless, choosing a standard does accomplish at least two things. First, it provides a single framework within which all services in the GeoCommons must be described. Second, it provides us (the GeoCommons stewards) with a tool to hold service providers accountable to a minimum level of documentation if they wish to participate in the system. The result is that developers and GIS users that come to the portal as clients can be assured they will be presented with at least a standard core level of information that they can evaluate.

In doing our research we confined our options to the three currently supported standards available directly within the ArcGIS Portal Extension (9.3) software. We also looked at the Minnesota "GeoService Finder" application and the meeting notes describing how this registry of services was designed.

Within the ArcGIS Portal Extension, there are three documentation standards for services that are supported out of the box: North American Profile (Services), ISO 19139 / 19119 Web Services, and INSPIRE Metadata (Services). Each of them are similar in most respects, and any of them would probably suffice as a basis for our efforts. We chose to compare them by creating a matrix that illustrates the fields of information that each supports, identifying those that are common among them, and highlighting those that are not (electronic attachment: report_appendix_a.xlsx).

It is worth noting that the ISO service metadata template as implemented in the ArcGIS Portal Extension is a subset of the full ISO standard, and therefore simpler to understand. It was also (in our judgment) the simplest of the three options readily available .

Another consideration was the work that has already been done to create a service registry on the MnGeo website, the "GeoService Finder" application. After reviewing the meeting notes from the workgroup who put that project forward, we discovered that the GeoService Finder uses as subset of the ISO standard as a basis for the registry database. This was considered a positive reason for continuing with the ISO standard.

Attached is a sample of the metadata report that results from using the ISO standard input screens in ArcGIS Server Portal Extension. (electronic attachment: report_appendix_b.pdf)

Beyond the Basics: "Comprehensive" Web Service Documentation:

When thinking about how to deliver information about GIS web services, it helps to group the people interested in these services into three broad categories: 1) data consumers, 2) data managers, and 3) application developers. Once we had looked at the standards for GIS web service documentation

available within the ArcGIS Portal Extension, we asked ourselves if that basic level of information was enough to satisfy the information requirements of these three user groups. The answer is, "No."

In our view, in addition to the ISO service metadata record, each GIS web service should be described at 3 levels, corresponding with the three user communities. Some of the information at each level can be drawn from the "standard" ISO metadata record, however much of this documentation is ancillary to the standard record. Moreover, each members of each user community need to be able to find the information they are after without having to sift through mounds of text that is not targeted at their needs. Customization and extension of the portal website and/or templates will be required to accommodate this function.

Data consumers, data managers, and application developers are all searching for GIS services, and trying to evaluate their "fitness for use" in a specific operating environment. What we might call "Level 1" service metadata needs to give them a basic understanding of what is being provided, whether it is a data service, a geoprocessing service or a map service. This information is typically found in the name, description and history sections of the service metadata. Links to a graphic or the ability to add the service to a map window in the Portal web site would also be appropriate here. Links to download ArcGIS "layer" files with embedded connections to the service could also be appropriate at this level. We could refer to these and other similar pieces of data as components of the service's "introductory" page.

The next level of documentation should be oriented to support the needs of data managers and application developers. These customers are typically looking for, evaluating, and presenting GIS services to a group of users whom they support. They are more sophisticated in how they evaluate and put a service to use. What we could call "Level 2" service metadata should include information that addresses "trust" issues. More about the topic of “trust” will be discussed in the next section of this report, but briefly this would include tools and information that would allow the customer to assess the uptime of the service, any service level agreements between state agencies that this service is a part of, commitments by the provider to inform customers of major changes in the content of the service, and so forth.

The third level of service metadata should be targeted at application developers and advanced data managers. At this level the service provider should provide code snippets such as MapServer mapfile connection and display examples. This would also be the place to include links to the REST endpoint(s) for the service, and example code in one or more languages showing how to implement a simple test application using the service. Links to wiki-pages or how-to discussion lists concerning implementation of the service would also be appropriate at this level. Information about the infrastructure that supports the service and how it may or may not be able to scale as the number of users increases is helpful to these

types of customers. As an example, MetroGIS has created developer oriented metadata about their Geocoder Service. Follow this link: http://www.metrogis.org/data/apps/geocoder/ to view it.

It is not reasonable to expect that all GIS web services that are registered and exposed through the GeoCommons Portal will be documented to this level. However there may be some "core" services that are considered part of the backbone of the state enterprise GIS environment that would be useful to use as examples of what is possible to achieve through this type of "comprehensive" GIS web-service documentation.

Key Characteristics Needed to Achieve "Trust" in a Web Service

Along with many benefits, choosing to use a GIS web service provided by another organization carries with it an inherent level of risk. That risk can run the gamut of low to very high, depending on your level of investment in using the service and the commitments offered by the service provider.

This topic has already been addressed by members of the GeoCommons committee (Mark Kotz and Matt McGuire) in two presentations they gave early on in the life of our group. The focus of our subgroup was to review and then follow up on the content of these presentations in order to provide an overall strategy that we can implement as part of the portal project.

Mark and Matt's presentations included some key words (or "factors") that provide a good starting point to the discussion:

- Reliability
- Performance
- Support
- Integrity
- Commitment

In light of these "trust factors", we posed this statement: *"In order to trust a web service, I need…"* and attempted to finish it with a set of of requirements and example methods to achieve them.

In order to trust a web service, I need…

1. *An honest measure of uptime of the service.* One of the most immediate and easy to understand measures of **reliability** is a statistic or set of measures summarizing the amount of time that the service is available and functioning properly. This is typically done using an automated program that sends a request to the service at regular intervals and checks to see if the proper result is returned. If so, then the service is considered "up," if not then the service is considered "down" or not functioning. The results are logged to a database and, depending on the level of sophistication of such a tool, the system can be queried to show not only the percentage of time the service is "up", but also to show the history of the service's availability over a set time period. This type of information can be crucial to a developer who is building an application that requires 24/7 availability, or which is being depended on in an emergency situation. It can also be used to validate claims made by a service provider or to evaluate whether the terms of a service level agreement (SLA) have been met.
2. *A meaningful measure of the performance of the service.* Typically when we talk about **performance** we are thinking about the speed at which the service responds to a request. While a provider can make promises about performance, some kind of independent measure is necessary. Like the uptime measure it should be fairly straightforward to use an automated software program to perform this measurement and log it to a database. In the case of performance however, having a good explanation of the "test" is important. Such tests need to be clearly defined, so that potential clients can understand the context of the result. Different types of clients have different expectations and needs when it comes to the performance of a service. What is an acceptable level of performance for one application may be wholly inadequate for another. Likewise, different applications will place different levels of stress on a service. What all clients need is some way to gauge whether a given service can meet their performance expectations. As results are logged to a database they can be reported on a web page and examined. A potential client should even be able "test" the service and possibly see the impact of that test on the service's performance measures during the relevant time period.
3. *To know who to contact if I discover a problem with the service.* It is important to remember that behind every service request is a real live person trying to accomplish a goal. Likewise, behind every published service is a person as well, the "service provider." While service providers might hope that they can just publish their services and remain relatively anonymous, the truth is that people using a service need some level of **support**, even if that is just knowing who to contact in the event of a problem and what what type of response they can expect in return. Basic service provider contact information is found in the services metadata record. This is the bare minimum requirement. In order to be clear about what a client can expect in terms of response, a service provider should also have a way to clearly state the type of response clients will provide when contacted. Such a response can range from informal (a simple acknowledgement) to formal (an incident ticket) depending on the relationship between the client and provider.

4. *To know when planned down-time is scheduled.* It is common for GIS web services (and the servers they run on) to require scheduled down-time for data updates and system maintenance. These blackout periods are often scheduled during what the system administrators consider "off-peak" hours. Given the global reach of the internet however, one person's "off-peak" time slot can occur during another person's prime work hours. Clients need to be able to plan their tasks around the published **availability** of a service. The documentation for a service should include a link to that service's "calendar of availability." One idea for achieving this is to create a "Google Calendar" for the service that clients can subscribe to. Clients can view the service's schedule and even use the calendar options to be notified when down-time is imminent.
5. *A commitment to notify users if there is a change in how the service is to be delivered.* Like everything else, GIS web services will change and evolve over time. What distinguishes a service with **integrity** from one that cannot be trusted is that the service provider has made a commitment to notify users about the nature, extent, and timing of these changes before they occur. This communication can take a variety of forms, from "service bulletin" emails to RSS feeds and twitter accounts which clients subscribe to. Such notification allows users and developers time to adjust their software. As a best practice, a service provider planning significant changes should publish them in a secondary "parallel" service for a set time period, giving clients the ability to test and migrate their systems over. At the end of this period, the original service can be retired.
6. *A commitment to notify users if the content of the service (above the feature level) will change.* Some GIS web services are very simple, they portray one type of content (e.g. "bus stop locations") that is clear and unambiguous. Other services may bundle two or more data resources together and publish them as a group (e.g. "base map"). Either way, service clients need to be able to trust that the content of a service is not going to change on them without warning. For example, the publisher of a base map service could remove a component layer of the map without affecting the overall delivery of the service. Like item 5 (above), a commitment to pro-actively communicate with interested clients and the service provider's track record in fulfilling that commitment stands as a measure of the the **integrity** of the service. The means of communicating such changes are similar as well – service bulletin emails, RSS feeds, twitter accounts, etc. can accomplish such a task without too much effort on the part of the service provider.
7. *Reliable information about the accuracy and currency of the data (for data services).* Generally speaking, this is a direct reference to keeping the basic metadata about a service up-to-date. The service metadata should provide some reference to metadata for the data that is being served, and that information needs to be kept up as well. Again, this goes to the **integrity** of a service. A data service does not have to be highly accurate or up-to-date to fulfill this obligation,

however the metadata for the service must reliably describe these factors so that a client can judge its fitness for use in their application.

8. *A clear indication of how long this service will continue to be published into the future.* Users of a service need to be able to gauge the **commitment** of the service provider so that they don't get caught short by a service that suddenly disappears. No one would commit to providing a service forever, but it is reasonable for a user or application developer to think that a service will continue to be published for one, two, or three years into the future. The clearest way to achieve this is for the client to enter into a "service level agreement" (SLA) with the provider. Such an agreement would (among other things) specify the length of time that the service will be provided. For users that are in a more casual relationship with the provider, it may be enough simply to know that there is a pre-existing SLA between two trusted organizations. These users can "draft behind" a pre-existing SLA, trusting that if the service provider has committed to such terms, no further agreements are necessary. For this reason, if a GIS web service is subject to any service level agreements between public agencies, these SLAs should be published on the web and linked to on that service's portal web page or metadata. In the absence of such agreements existing, service providers need to provide a clear statement indicating their intent to continue publishing the service over a specific time frame (start date, end date).

Any GIS web-service that achieved all 8 of these criteria could be considered "completely trustworthy." With the exception of number 2 (performance measures), none of these criteria are complex or particularly difficult to achieve. Building an infrastructure of services that supports an active developer community and a growing number of satisfied clients depends on the establishment of trust in the services. This is a result of clear and regular communication between clients and service providers.