Open GIS Consortium

Technology Office

4899 North Old State Road 37
Bloomington, IN 47408

Telephone: +1-812-334-0601
Facsimile: +1-812-334-0625

Request For Quotation

And

Call For Participation

In the

OGC Web Services 2 Initiative

Initial Operating Capability and Demonstration

Annex B: OWS-2 Architecture

RFQ Issuance Date: November 24, 2003

Proposal Due Date: January 9, 2004

TABLE OF CONTENTS

Annex B: OGC Web Services 2 (OWS-2) - Architecture

1 Overview

The architectures presented in this Annex are based upon a collaborative effort between OGC Web Services 2 (OWS-2) **Sponsors** and OGC's IP Team. The architecture team used results from previous and ongoing OGC Interoperability Program initiatives, existing OGC discussion papers and specifications, OGC Technical Committee activities, and publicly available documentation from related standards initiatives (W3C and ISO) and elsewhere.

Section 2 provides an overview of the OWS-2 development threads for OWS-2 that resulted from the analysis of the RFT responses.

Section 3 discussed the architectural approaches and issues for each of the OWS-2 development threads.

Appendix A contains key references applicable to this RFQ.

Appendix B is a description of a fictitious scenario that is to be used in support of the Information Interoperability thread of OWS-2.

2 OWS-2 Initiative Threads

The OWS-2 Initiative threads were developed as a result of the analysis of the responses to an OWS-2 Request for Technology (RFT). Originally there were 8 technology themes in the RFT to which industry responded. After analyzing the responses the OGC Interoperability Team recommended to the sponsors of the RFT that the content of the OWS-2 initiative be constructed of the 5 following thread:

1) Common Architecture

2) Technical Baseline Maturation

3) Information Interoperability

4) Image Handing /Decision Support

5) Open Location Services

An Introduction to these 5 threads is described below which is then followed by a detailed discussion of the architectural implications of each of the initiative threads.

2.1 Common Architecture

The Common Architecture (CA) thread addresses issues and requirements driving the further elaboration of the of the core set of OGC specifications to bring them inline with the industry mainstream Web Services implementations. The objective of CA is to define a consistent adaptation and integration of Web Services standards, defined by leading organizations such as W3C, OASIS, Microsoft, IBM, Sun Microsystems, and the open-source community, with the OWS service framework in general and the set of core OGC specifications (WMS, WFS, CS-W, WCS, and WOS). This thread addresses issues of fundamental architectural significance to OWS, including metadata about resources (services and data) and how to adopt Web Service (specifically WSDL, SOAP, UDDI) technologies.

2.2 Technical Baseline Maturation

The OGC Technical Baseline contains a list of all OGC specifications and specification candidates, their type, title, current version, document number, description, and primary author/editor. This information provides the reader with a current snapshot of work within the OGC.

For OWS will develop conformance tests for the core set of the Technical Baseline that include: WMS, WFS, CS-W, WCS, and WOS.

2.3 Information Interoperability

The Information Interoperability (II) thread of OWS2 focuses on the interoperable expression, exchange and access of geographic information within and across information communities (Information communities are groups that share common geographic terms and common spatial feature definitions). The II activities will explore ways for enhancing common ways for representing, sharing and processing of information between collaborating entities to maximize opportunities for exchange and reuse of geographic information.

2.4 Image Handling for Decision Support

The growing availability of geographic imagery opens opportunities for applications in public and private sectors. Key to the exploitation of geographic imagery is the need to reduce the gap between the raw remote sensing data and the information needs of applications users. The Imagery Handling theme of OWS2 aims to radically increase the availability and usability of geographic imagery enabling the use of information derived from imagery to be used with all other forms of geographic information.

2.5 Open Location Services

The OpenGIS Location Services (OpenLS) represent an open (middleware) platform for location-based application services for mobile assets and terminals. The primary goal of the OpenLS initiative series is to define the specifications for the "Core Services and Abstract Data Types (ADT)" that comprise this platform. The services are defined in two specifications that are now in the RFC stage within OGC TC. Whereas the specifications are considered to be stable, there is ongoing discussion to enhance these specs (add minor features) during OWS 2. Also, there is a desire to add new specifications for traffic and tracking services.

3 OWS-2 Architecture

3.1 OpenGIS Reference Model

The OpenGIS Reference Model (ORM) provides an architecture framework for the ongoing work of the OGC. Further, the ORM provides a framework for the OGC Technical Baseline. The OGC Technical Baseline consists of the currently approved OpenGIS Specifications as well as for a number of candidate specifications that are currently in progress.

The ORM has the following purposes:

Provides a foundation for coordination and understanding (both internal and external to OGC) of ongoing OGC activities and the Technical Baseline;

Update/Replacement of parts of the 1998 OpenGIS Guide;

Describes the OGC requirements baseline for geospatial interoperability;

Describes the OGC architecture framework through a series of non-overlapping viewpoints: including existing and future elements;

Regularize the development of domain-specific interoperability architectures by providing examples.

The ORM is a living document that will be revised on a regular basis to continually and accurately reflect the ongoing work of the Consortium. (see http://www.opengis.org/specs/?page=orm. It is encouraged that respondents to this RFQ understand the concepts that are presented in the ORM.

3.2 Common Architecture (CA)

3.2.1 Scope

OpenGIS Web Services (OWS) are individual components of dynamic geospatial computing applications; they are also parts of an overall paradigm for building solutions to geospatial "problems". This paradigm - the Spatial Web - imposes both conceptual and implementation constraints on how OWS works.

- Conceptual constraints include service orientation, n-tier distribution capability, self-description, and stateless operation; these constraints generally address functionality.
- Implementation constraints include use of common XML encodings, HTTP transports, tightly defined interface syntax, specific information models for service descriptions and other metadata; these constraints generally address interoperability.

Taken together, these constraints form a common basis or architectural framework to guide the successful and interoperable implementation of OWS instances. From a practical viewpoint, the more of these common constraints can be satisfied once for all services, the less work it will be either to implement existing service types or to define new ones. As with any functional web, this should lead more quickly to the efficiencies of both scale and diversity (many nodes of many different capabilities) which give the Web much of its value.

The OWS Common Architecture defines a framework of guiding concepts, terminology, fundamental patterns and organizing principles for implementation and deployment. The OWS Common Architecture establishes common interfaces, exchange protocols, and services that can be utilized by any application. OpenGIS® Implementation Specifications provide guidance to application developers on how to build their applications to comply with the OWS Architecture. OpenGIS® Services are implementations of services that conform to OpenGIS® Implementation Specifications. Compliant applications, called OpenGIS® Applications, can then "plug into" the framework to join the enterprise operational environment. This loosely coupled approach to enterprise development results in very agile systems.

By building applications to common interfaces, each application can be built without build-time or run-time dependencies on other applications. New applications and services can be added, modified, or replaced without impacting any other applications. In addition, operational workflows can be changed dynamically, allowing rapid response to crises conditions.

Specifically for OWS 2, the CA thread will concentrate on adapting the set of core OGC Specifications to be conformant with the Simple Object Access Protocol (SOAP), the Web Services Definition Language (WSDL) and Universal Description, Discovery, and Integration (UDDI) XML standards from the W3C.

3.2.1.1 Problem and Objectives

The OGC interface specifications currently make use of the HTTP GET and HTTP POST invocation mechanisms (see the ORM) to communicate with OGC compliant servers. This situation applies a constraint for the acceptance of the OGC specifications within the general Web Services community, as defined by the use of the SOAP ,WSDLand UDDI XML standards.

The long-term goal is to create an infrastructure such that individuals can contribute components of a modular solution that can be reused by others in the community as part of a whole system implementation. Such an infrastructure accommodates new components (evolvable) and the upgrade or replacement of existing components (maintainable). The open systems approach is intended to create a marketplace for data services, enabling ‘plug and play’ components contributing to tailored geospatial analysis tools. Examples include, but are not limited to:

- Reference architecture descriptions, proof-of-concept prototypes, middleware components, and ‘use scenarios’ demonstrating end user benefits;
- Data and Service representation methods, languages and metadata techniques enabling Earth science community-centric services and open tools sets; and
- Technologies managing data integrity, authentication, and heritage, among others.

In an effort to help achieve this long-term goal the CA thread will concentrate on providing adapting the OGC core specifications to operate in a SOAP/WSDL environment in a consistent fashion.

3.2.1.2 Requirements

The proposals for the CA thread should address the respondent’s capabilities and understanding of implementing SOAP, WSDL and UDDIcomponents and how they can be consistently applied to the following OGC core specifications:

1. Web Map Service with Style Layer Descriptor (WMS)
2. Web Feature Service - Transactional (WFS-T)
3. Web Coverage Service (WCS)
4. Catalog Service for the Web (CS-W)
5. Web Object Service (WOS)

For more information on these OGC Specifications please see section 3.2.5 the CA Computational Viewpoint.

The proposals should also address the harmonization of the Service Information Model (SIM) and the Registry Information Model (RIM) (see the references in Appendix A) across all of the 5 Open Web Services mentioned above.

In addition, there is a requirement to provide implementations of the SOAP ,WSDL and UDDI enhancements to the servers as well as a demonstrable client that exercises each service using the SOAP, WSDL and UDDI implementations.

3.2.1.3 Deliverables

The deliverables for the CA thread are:

1. Interoperability Program Report (IPR) describing the overall approached used to apply the SOAP and WDSL standards to the OGC specifications and descriptions of the Technical Integration Experiments (TIEs) that were exercised.
2. Change proposals for corrections and enhancements to the WMS specification as tested and demonstrated in this initiative

3. Change proposals for corrections and enhancements to the WFS-T specification as tested and demonstrated in this initiative

4. Change proposals for corrections and enhancements to the WCS specification as tested and demonstrated in this initiative

5. Change proposals for corrections and enhancements to the CS-W specification as tested and demonstrated in this initiative

6. Change proposals for corrections and enhancements to the WOS specification as tested and demonstrated in this initiative

3.2.2 CA Enterprise Viewpoint

depicts a system concept that depicts a typical production environment where the OpenGIS Services Framework is the basis for enterprise-wide interoperability. There are six main functional parts to this production environment:

- Common Source Processing – where all source acquisition, assessment and processing occurs
- Feature Production – where feature production and management occurs
- Imagery Production – where imagery production and management occurs
- Other Information Production – where multi-source processing and specialized information production and management occurs
- Common Product Processing – the main product finishing and distribution capability
- Common Operations – common capabilities for customers to view and exploit deployed products

The OpenGIS Services Framework provides the common set of interfaces that spans these functional parts of the enterprise and provides enterprise-wide interoperability.

The elements of the architecture that are the focus of this test bed are as follows:

- *Common Architecture* – the common, foundational elements of the OGC Technical Architecture as described in the OpenGIS Service Framework and Interoperability Program Service Model (see the ORM).
- *Client Services*[1] – the client-side components of client applications that interact with users, and on the server-side interact with *Server-side Client Applications*, *Application Servers* and *Data Servers*. (Client Services are services of type Geospatial Human Interaction Services as defined in OGC Abstract Specification Topic 12.)
- *Registry Services* – provides a common mechanism to classify, register, describe, search, maintain and access information about Web resources (data and services).

[1] From ISO TC 211: The Client tier consists of the user environment. A client component (we refer to this as a Client Service, given a service perspective) contains the logic that presents information to an external source and obtains input from that source. In most cases the external source is a human end user working at their own computer, although the external source might also be process-oriented. The client logic generally provides menus of options to allow the user to navigate through the different parts of the

- *Processing Services* – the foundational application service building blocks that operate on geospatial data and provide value-add service.

- *Portrayal Services* – Portrayal Services provide specialized capabilities supporting visualization of geospatial information. Portrayal Services are components that, given one or more inputs, produce rendered outputs such as cartographically portrayed maps, perspective views of terrain, annotated images, views of dynamically changing features in space and time, etc.)

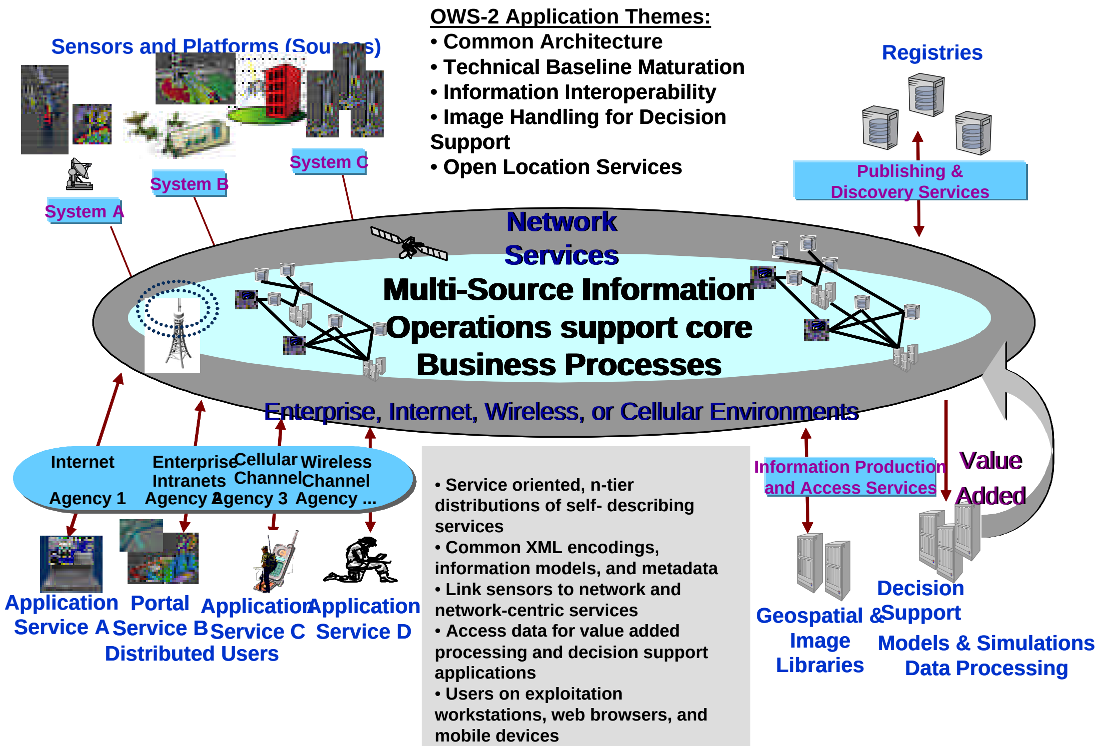


Figure 1. OWS1.2 Operational Architecture: System Concept

- *Data Services* – the foundational service building blocks that serve data, specifically geospatial data in OWS1.2.

- *Service Chaining* – the use of the Business Process Execution Language (BPEL) to orchestrate OWS services in a manner that meets a higher level business goal. While the use of a BPEL engine has a place in the OWES Architecture, the details of the orchestration is dependent on the domain. For this RFQ, the requirements for service chaining are discussed in section 3.5, Image Handling for Decision Support.

3.2.3 Use Cases

The following use cases exercise the services defined in Table 1 of the main body of the RFQ, and detailed in the section that follow, highlighting the use of the primary capabilities to be developed under this test bed.

3.2.3.1 Use of Registries for service publishing, finding, binding

The use case described here illustrates how the OpenGIS Framework can be used for service publishing, discovery and binding. First we consider how service *types* are defined and published. Next we describe how *service provider* organizations can "publish" their service *instances* to a registry. Finally we describe how *service consumer* organizations can find a service type of interest, construct software interfaces for a particular service type and then dynamically connect (bind) to any service instance of that type.

Define a service type. Consider how service types can be published. It is important to distinguish in this use-case between service types and instances. In the OpenGIS Framework, a service type is at least an interface signature type but may also include a set of service properties that contain information about computational aspects, such as the behavior and the environment of an interface, the types of data that can be handled by services of the type, and other semantic details of the service type. Service instances are "bind-able" access points on a network, conforming to an interface, having instance-specific metadata describing their operational capabilities (including, for example, information about the data they can return or process).

Usually an industry standards organization or consortium of service providers, such as OGC, will define standard service types, including their interfaces (specified as one or more WSDL service interface definition documents), descriptions of the semantics and behavior of the type and a reference to a taxonomy of service types. The WSDL interface part of the service type description specifies both the service interfaces (types, messages, port types) and protocol bindings, but not the access endpoints for an instance. As indicated, the service type description may also include a reference to a classification node in a service type taxonomy as well as other descriptive information about the behavior and semantics of services of given type.

Publish a service type description. Service developers (and standards consortia) make these service type descriptions publicly available by publishing a "service offer" to a *service type* registry. A service offer is a registry entry that advertises, in this case, a service type. The type offer consists of a description of the service type, an identifier for the interface(s) to which a binding can be established and possibly a reference to a classification node in a service taxonomy. The service offer may include the WSDL documents as part of the offer itself or reference them remotely in a service interface repository. The service offer may also be linked to a node in a service type taxonomy managed by the registry.

Find a service type description. Programmers at a service provider organization can use the service type registry to find, by searching the service type taxonomy or by searching for other descriptive properties of service types, and retrieve a complete description of the service type including its interface specifications (WSDL documents) and other behavioral characteristics and specifications. With this detailed information, programmers can construct industry-standard instances of the service type that implement the WSDL interfaces and have the behavioral characteristics of the type. Advanced software development tools can help to automate this process.

Publish service instances. Next, the newly constructed and deployed *service instance* must be published to a *service instance registry*. Either manually, or with advanced software development tools, a service instance description (i.e., "capabilities") document is constructed. The service capabilities document includes a WSDL instance (i.e., "service") section as well as any instance-specific message definitions. In addition to the WSDL instance information, the capabilities document includes operational information about the data the service handles, as well as references back to the service offer in the service type registry (and thus references to the WSDL interface descriptions) and to the service type taxonomy. The capabilities document is published as a service offer to a service instance registry. The service offer, in this case, is a registry entry that advertises the service instance. Some registry implementations may support automatic "harvesting" of service capabilities documents.

Construct a client. Finally, to build clients (service requestors) of a particular service type, programmers at a service consumer organization (e.g., an application developer) can use the service type registry to find, by searching the *service type* taxonomy or by searching for other descriptive properties of service types, and retrieve a complete description of the service type including its interface specifications (WSDL documents) and other behavioral characteristics and specifications. With this detailed information, programmers can construct “connector” modules used by service requestors (e.g., application clients) to interoperate with service instances. The connectors are constructed to use the industry-standard service interfaces, defined with WSDL, and according to the documented behavior and semantics of the service type. Advanced software tools can be used to help automate the construction of connectors. Once the connector for a service type is constructed, any instance of that type can be dynamically discovered and accessed. For example, a service instance registry can be used to dynamically discover, at runtime, service instances of a specified type having certain desired operational characteristics. Service instances that were found as a result of the search can be dynamically accessed at runtime using the previously constructed type connector. Advanced software tools and deployment environments may support dynamic (late) binding to services of any type, thus bypassing the need for programmers to construct connectors specific to a service type.

3.2.3.2 Feature Extraction

This use case focuses on using a Feature Production Client component for discovery of services and data, portrayal, feature editing, and feature data management. New service interfaces and encodings, to include Style Management Service(s), are employed. The emphasis is on using and extending Registry Services for service and information discovery, Web Feature Service (WFS), Styled Layer Descriptor (SLD) and Geography Markup Language (GML) 3.0. The Feature Production Client component also performs cartographic portrayal, imagery viewing, feature analysis and editing capabilities.

A Feature Production Client component designed to satisfy requirements of the OWS 2 use case might perform the following tasks:

- ***Registries as a means to find, access and process relevant services***.

  The integrated client discovers and binds with suitable services through a service registry. This is followed by runtime binding to these services.

- ***Registries as a means to find, access and process relevant data***.

  Similarly, the integrated client discovers suitable feature metadata data through a registry. This then leads to runtime use of these data in support of disaster response efforts.

- ***Use of Web Feature Server as a means to update feature data in support of disaster response***.

  The user employs recent imagery as a source for feature analysis and update. Again, the client employs the Image Catalog and Image Archive Service to access the imagery. Next, the client browses and queries Feature Collection Registries for Feature Collection (Product) Metadata. The user employs this metadata to select the appropriate feature data for use in disaster response. Having discovered the appropriate feature data, the client then employs Web Feature Service (WFS) to access the feature data. The client then uses feature extraction tools to update the data.

- ***Style Management Service and Styled Layer Descriptor in support of cartographic portrayal***

  The client provides the means to view, filter and interact with styled features and maps as overlays. An SLD-enabled Web Map Service is used to portray cartographic layers that are styled according to user-selected symbolization preferences. The user selects the desired SLD from a Style/Symbol Registry, which was previously constructed using a Style Editor component and a Style Management Service.

3.2.3.3 Imagery Exploitation

This use case illustrates the use of OWS services to support an Imagery Exploitation. This use case targets a scenario in which a user gets an overview of recent multi-spectral imagery for the disaster area, explores it using an Imagery Exploitation Client component, which is part of the integrated client, and then invokes a few service chains to conduct imagery exploitation.

- ***Use of Imagery Exploitation Client component***

  The user will use this component to find and use imagery data, and then find and use imagery application services to operate on the imagery data. The client must have search tools to specify, find and retrieve data. The client must also provide the user the means to view and interact with the data. The client must have tools to select and invoke imagery application services, and to invoke service chains (e.g. Image Catalog→Image Archive→Coordinate Transformation Service→Web Coverage Service). The client might access map data to depict their study area, view imagery footprints from an Image Catalog, select imagery coverage, etc. This also involves using Web Map Servers and Web Feature Servers.

- ***Image Catalog is used to access Image Metadata***

  The user wants a recent imagery over the disaster area. The user formulates a request based upon the well-known Imagery Metadata Model employed by the Image Catalog. The user employs the client to access an Image Catalog to find recent satellite, aerial and ground imagery of the area. (As described here, the client knows about the Image Catalog Service, but the client might also discover this service through a service registry that operates as a broker for several Image Catalogs.)

- ***Image Archive Service is used to access Imagery***

  The user finds the Image Metadata they want through the catalog search and now must access the appropriate Imagery Archive Service to fetch the imagery and imagery support data. The client formulates the request to the archive, stipulating where the data are to be delivered for the client to later exploit. This process might take some time, if for example the archive has to fetch the data from tape storage. The Imagery Archive Service completes its assignment by delivering the imagery data to the appropriate Web address. Optionally, the Imagery Archive Service might employ a Notification Service to alert the User about the availability of their requested data. The data is now available for exploiting, although it is still in its tiled archive format. (The archive service likely supports mosaicking, re-tiling, and re-sampling to deliver the imagery in a form that is ready for exploitation.)

- ***Service Registry and use of Web Coverage Service and Coverage Portrayal Service***

  The user now invokes a tool that uses service registries and service chaining to be able to view the imagery. This tool allows the user to specify a filter for how they want to view the imagery. Upon doing so, the client begins a sequence of operations that employ registries and service chains to accomplish the desired task. First, the client accesses a Service Registry to find a Web Coverage Service that will operate on one of the output formats produced by the Imagery Archive Service. It finds the right service. Next the client searches the Service Registry to find a matching Coverage Portrayal Service. It finds one that will do the job, but a Coordinate Transformation is required. The client now has the full sequence of necessary operations and constructs and invokes the service chain: Web Coverage Service→Coordinate Transformation Service→Coverage Portrayal Service. The client then binds to the Web Coverage Service, requesting bands 5, 6, and 27 of the imagery, as a reduced-resolution “browse thumbnail” for the overall area. The coverage data is chained through the chosen Coordinate Transformation Service and Coverage Portrayal Service, which renders the imagery as a simple JPEG picture. The user then browses the resulting image with their client imagery viewer. Having found a desirable image, now the user now fetches the full-resolution imagery using the same service chain: Web Coverage Service→Coordinate Transformation Service→Coverage Portrayal Service. The user browses and combines the different image bands to meet their exploitation needs.

3.2.4 CA Information Viewpoint

3.2.4.1 Geospatial Data

The following sections provide examples of the geospatial information types potentially required for OWS-2.

3.2.4.1.1 Raster Data

Raster data is the class of data items that represent a picture. This data consists of a single layer of pixels that can be readily visualized by a user. For purposes of CIPI-1, raster data is considered a separate data class from imagery or coverage data. Relevant specifications for CIPI-1 Raster data are:

- JPEG *–ISO/IEC 11544*
- PNG – Portable Network Graphics W3C Recommendation version 1.0
- TIFF – Tagged Image File Format version 6.0
- TBD

3.2.4.1.2 Vector Data

Vector data is the class of data that represents geospatial features. This data contains detailed information, both geometry and attributes, about individual geospatial features. As such, vector data lends itself to data processing applications instead of visualization (although that processing may enable visualization). Relevant specifications for CIPI-1 Vector data are:

- GML 2.0 OGC Geography Markup Language version 2.1
- GML 3.0 – OGC Geography Markup Language version 3.0
- VPF – Vector Product Format MIL-STD-2407
- TBD

3.2.4.1.3 Imagery Data

Imagery data is the class of data items that represent one or more pictures (bands) and the associated metadata. This data differs from Raster Data in its complexity. An Imagery data set may consist of one or more Raster data sets often taken using different parts of the electromagnetic spectrum. In addition, imagery data sets contain information on where and when the image was collected as well as supporting data describing the conditions when it was collected. Relevant specifications for CIPI-1 Imagery data are:

- NITF – National Imagery Transmission Format v 2.1 MIL-STD-2500A
- TBD

3.2.4.1.4 Gridded Data

Gridded data is the class of data items that contain a matrix of values representing measured phenomenon. This data differs from Raster Data in that the measured phenomenon is not inherently visual. A collection of elevation points, for example, is a common form of gridded data. Like Imagery data, however, a Gridded data set may contain information on where and when the data was collected as well as supporting data describing the conditions when it was collected. Relevant specifications for CIPI-1 Gridded data are:

- DTED – Digital Terrain Elevation Data MIL-PRF-89020B
- TBD

3.2.4.2 Management and Control Data

3.2.4.2.1 Query Languages

All of the services used in the CICE have a queryable interface. To use those interfaces, it is necessary that the client and server have a common understanding of what constitutes a valid query. Standardized query languages provide the common vocabulary and grammar needed to enable this sliver of interoperability. Relevant specifications for CIPI-1 Query Languages are:

- OGC Filter Encoding Language – Filter Encoding Specification (OGC discussion paper)

3.2.4.2.2 Service Description Languages

The CICE supports the concept of service publication, discovery, and binding. To enable this capability, there must be a common language for publishing the information necessary for a potential client to assess the suitability of a service and to understand how to bind to it. Service Description Languages provide a standard way of expressing that information. Relevant specifications for CIPI-1 Service Description Languages are:

- WSDL – Web Service Definition Language version 1.2 (http://www.w3.org/2002/ws/desc/)

3.2.4.2.3 Styling Description Languages

The Web Map Service provides users with the ability to customize the symbolization of the data they are requesting. In order to work across all WMS implementations, there must be a standard way of expressing the desired symbolization. Standardized Styling Description Languages address that need. Relevant specifications for CIPI-1 Styling Description Languages are:

- SLD – Style Layered Description Specification (OGC discussion paper)

3.2.5 CA Computational Viewpoint

The Information Technology environment in which the CIPI initiatives will take place is the Critical Infrastructure Collaborative Environment (CICE). The CICE will initially be based on the OGC's Interoperability Program Service Model (IPSM), OpenGIS® Service Framework, and relevant elements of the Canadian Spatial Data Infrastructure (CSDI).

The IPSM (see the ORM) describes a computational model for OpenGIS® Services. The objective of the IPSM is to detail how geospatial software services plug into broader interoperability infrastructures to use and extend diverse, loosely coupled sources of data and services. The IPSM draws on Topic 12 of the OGC Abstract Specification (Service Architecture-ISO 19119) but focuses more specifically on current technologies, platforms and mechanisms for enabling implementation of interoperable services.

The CICE will be based on the OpenGIS® Service Framework, providing a platform for geospatial interoperability between applications and critical infrastructure information resources (see the ORM). This Framework includes the following:

- The OpenGIS® Framework establishes common interfaces, exchange protocols, and services that can be utilized by any application.
- OpenGIS® Implementation Specifications provide guidance to application developers on how to build their applications to comply with this framework.
- OpenGIS® Services are implementations of services that conform to OpenGIS® Implementation Specifications.
- Compliant applications, called OpenGIS® Applications, can then "plug into" the framework to join the enterprise operational environment. This loosely coupled approach to enterprise development results in very agile systems.

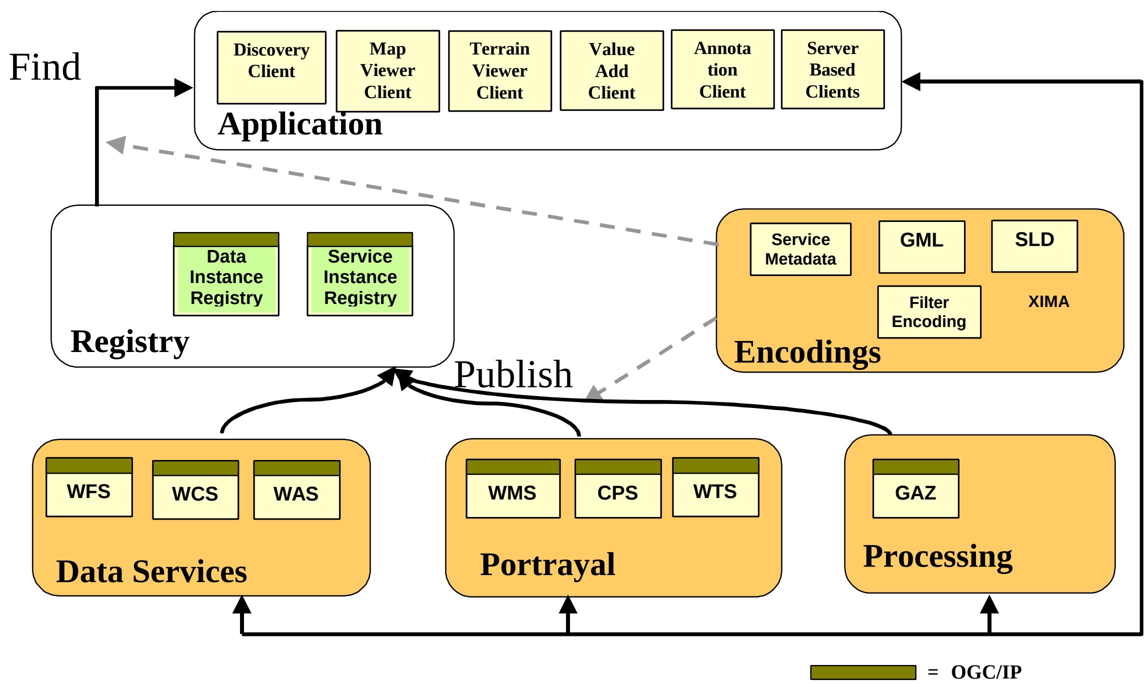


Figure 2 – Open GIS Service Framework elements for OWS-2

3.2.5.1 Web Map Servers

Relevant Specifications: *OpenGIS® Web Map Server version 1.1*

OpenGIS® Style Layer Descriptor discussion

A Web Map Server (WMS) generates "pictures" of georeferenced data. Independent of whether the underlying data are simple features (such as points, lines and polygons) or coverages (such as gridded fields), the WMS produces an image of the data that can be directly viewed in a graphical web browser or other picture-viewing software. A WMS labels its data as one or more "Layers," each of which is available in one or more "Styles." Upon request a WMS makes an image of the requested Layer(s), in either the specified or default rendering Style(s). The image request, called GetMap, indicates the Spatial Reference

System (SRS) and Bounding Box of the portion of the Earth to be mapped, and the output width, height and format of the picture. Typical output formats include Portable Network Graphics (PNG), Graphics Interchange Format (GIF), Joint Photographic Expert Group format (JPEG), and Tagged Image File Format (TIFF). When the data do not cover the entire field of view (such as a network of roads that includes the space between the roads), the background can be made transparent in some output formats.

An extension of the basic Web Map Server is the Styled Layer Descriptor (SLD) Web Map Server. The SLD enabled WMS inherits all of the attributes from the Web Map Server then adds support for the use of Styled Layer Descriptor documents to specify styling. Instead of generating maps of particular named layers in one or more predefined styles, an SLD Map Server extracts features from a data provider and renders them using a stylistic description encoded in XML.

3.2.5.2 Web Feature Server

Relevant Specifications: *OpenGIS® Web Feature Server version 1.0*

The Web Feature Service (WFS) supports the query and discovery of geographic features. In a typical Web-base scenario, Web Feature Service (WFS) delivers GML (XML) representations of simple geospatial features in response to queries from HTTP clients. Clients (service requestors) access geographic feature data through a WFS by submitting a request for just those features that are needed for an application. The client generates a request posts it to a WFS instance (a WFS server on the Web). The WFS instance executes the request, returning the results to the client (service requester) as GML. A GML-enabled client can manipulate or operate on the returned features.

The Transactional Web Feature Server extends the base WFS class with enhancements to allow the client to insert, update, and delete data.

3.2.5.3 Web Coverage Server

Relevant Specifications: *OpenGIS® Web Coverage Server 1.0*

The Web Coverage Services (WCS) allows access to geospatial "coverages" containing values or properties of geographic locations, rather than static maps (server-rendered as pictures). In a typical Web-based scenario, Web Coverage Services delivers coverage data (e.g., images or DTED) in response to queries from HTTP clients. Access to intact (unrendered) geospatial information is needed for client-side rendering, multi-valued coverages, and input into scientific models and other clients beyond simple viewers. The WCS interface supports delivery of images, multi-spectral imagery, elevation data (e.g., DTED) and other scientific data.

3.2.5.4 Web Object Server

Relevant Specifications: *OpenGIS®* Web Object Service Implementation Specification *version 0.0.3*

The Object Service Implementation Specification provides an interface to manage different types of objects, such as styles, symbols and images. The specification defines a set of base XML types that define the behavior of a Web Object Service. A Web Object Service is a generic web-based repository interface. The interface support the following operations: *GetCapabilities, DescribeObjectType, GetObjectById, GetObject, LockObject* and *Transaction.* The specification assumes that the distributed computing platform is HTTP and may define both XML (suitable for the POST method) and Keyword-Value Pair (suitable for the GET method) encodings of each operation.

3.2.5.5 Catalog Service for the Web

Relevant Specifications: *OpenGIS® Catalog Service version 2.0*

The OpenGIS® Catalog Service Interface Specification defines a common interface that enables diverse but conformant applications to perform discovery, browse and query operations against distributed and potentially heterogeneous catalog servers. Spatial Catalog servers typically contain metadata about spatially referenced information such as maps, schematics, diagrams, or textual documents. The specification uses metadata and spatial location to identify and select layers of interest, and provides for interoperability in catalog update, maintenance, and other Librarian functions. The specification is designed for catalogs of imagery, geospatial information, and mixtures of the two. (Future versions of the specification may also support services.) It specifies open APIs that provide discovery services, access services and interfaces for catalog managers, including a complete Catalog Query Language. Detailed implementation guidance is provided for establishing and ending a stateful catalog session to: query the catalog server properties, check the status of a request, cancel a request, issue a query, present the query results, and get the schema of a discovered collection.

The Data Registry Service extends the Stateless Catalog by providing a base schema and management interfaces (create, update, and delete) to support the discovery of data providers.

The Services Registry Service extends the Stateless Catalog by providing a base schema and management interfaces (create, update, and delete) to support the discovery of Web Services.

3.2.6 CA Engineering Viewpoint

3.2.6.1 OWS-2 Test Platforms

In support of the OWS 2 initiative there are a collection of t4est platforms that have been made available. These test platforms provide data and/or software that can be used by the respondents to the OWS 2 RFQ as part of their proposed technical approach. The respondents are encouraged to incorporate these test platforms into their respective technical approaches to the various OWS 2 threads where appropriate.

- MSN Terraserver - Microsoft Research offers the use of Microsoft TerraServer (http://terraserver-usa.com) as a platform to host selected OWS-2 prototypes in order to test and validate OWS-2 proposed standards and architecture and assess .Net frameworks. (POC: Tom Barclay, <tbarclay@microsoft.com>)

- CommerceOne - Composite Process Management Platform - Test Platform that extends OGC's understanding of composite process management function developed as part of CIPI-1 (Threat Response System). (POC George Haro, <George.Haro@commerceone.com>)

- Collaxa - BPEL Orchestration Server - Test Platform for increasing OGC’s understanding of automating business processes based on the BPEL4WS specification. (POC Doron Sherman <doron@collaxa.com>)

- SPOT Image (IHS) Selected and controlled access to the SPOT Image Programming Centre, to submit acquisition requests (possibly as OGC Image Handling Services) and acquire data over an OWS-2 area of interest (AOI) with specific support from SPOT Image Engineering & Management Personnel. Selected and controlled online access to the SPOT Image Catalogue with specific support from SPOT Image Engineering & Management Personnel. o Selected and controlled online access to the SPOT Image Production and Access Services. (POC Didier Giacobbo <Didier.Giacobbo@spotimage.fr>,)

- NASA (Image Archive) - Test Platform providing access to NASA-EOS data. This platform will provide access to the existing NASA Synergy Data Pools using the recently approved OpenGIS WCS 1.0. The Data Pools contain on-line access to data products from multiple instruments, e.g., MODIS, ASTER, MISR. (For further details see the Image Handling theme in the OWS2 RFT.) (POC George Percival <percivall@nasa.gov>)

- OGC Web-based Conformance Testing Engine (TBM) - OGC offers the use of the Conformance Testing Engine for Web-based specifications for the development and testing of Conformance Tests (URL: http://cite.occamlab.com/tsOGC) (POC: Kurt Buehler <kbuehler@opengis.org>)

- OGC Portal Reference Implementation - (POC: Kurt Buehler <kbuehler@opengis.org>)

3.3 Technical Baseline Maturation (TBM)

During the last several years, the Open GIS Consortium has experienced tremendous growth in terms of number of members, development activity and number of specifications and specification candidates. Therefore, the OGC adopted a framework for the communication of OGC's current technical baseline and future plans for the baseline.

The OGC Technical Baseline contains a list of all OGC specifications and specification candidates, their type, title, current version, document number, description, and primary author/editor. This information provides the reader with a current snapshot of work within the OGC.

The OGC Technical Plan provides, for elements of the baseline with future work identified, 6 and 12 month activity plans and an indication of whether a compliance test exists. Within the 6 and 12 month plans are indications which version will be available during that time frame and which tests will be available.

This section of the OWS-2 RFQ will indicate elements of the Technical Baseline that require work within the testbed either on the specification itself (in terms of specific technical issues of interest to the sponsors). In overview, these items are:

- Web Map Service 1.2 Specification Improvement, resulting in version 1.3
- Web Feature Service 1.0 Specification Improvement, resulting in version 1.1
- Web Coverage Service 1.0 Specification Improvement, resulting in version 1.1
- Catalog Service for the Web 1.1.1 Specification Improvement, resulting in version 2.0
- Location Services 1.0 Specification Improvement, resulting in version 1.1
- Geography Markup Language 3.0 specification improvement, resulting in 3.1

These improvements are mentioned in the reference portions of this RFQ (as they fit with the other OWS-2 threads). The remainder of this section deals with the primary focus of the TBM thread: compliance testing.

3.3.1 Compliance Testing Overview

The CITE Initiative established a program for testing software components for compliance to OGC Web services specifications. This section addresses the high level architecture of the system that was designed to fulfill these goals and to provide references for further reading concerning the details of writing tests. Annex A of the RFQ document provides the requirements specific to the development of tests within the OWS-2 initiative.

3.3.2 Goals for compliance testing

The value of compliance testing is in the extent to which it provides a measurement of the accuracy of an implementation of the underlying specification. In order to be useful, the compliance test must:

• Provide adequate coverage of the underlying specification

• Clearly indicate cases where the implementation is deficient

• Be available and affordable to the implementers

• Be impartial in the manner in which it implements tests

• Be easy to use and produce a clear result

• Provide a strategy for release control and appeal management

While a compliance test suite can never provide complete certainty that an implementation complies in all circumstances, a well-designed test suite can help the drive towards consistency and interoperability.

3.3.3 Types of Testing

3.3.3.1 Compliance for certification

The primary focus of the Technical Baseline Maturation element of the OWS-2 initiative is compliance testing for certification. The effort will develop test scripts and/or modify existing tests on selected OGC Web services. Tests typically use a standard, pre-defined dataset so that the responses to test requests will be relatively uniform and more easily verifiable. The intended goal of certification tests is to validate that a server complies with a published specification. In this context, compliance means that the server accepts all of the specification’s mandatory requests and responds to all requests in the manner mandated by the specification.

Note that certification tests *do not* verify that the server functions properly in other respects. For example, certification testing is not likely to reveal errors in spatial reference system conversion or color accuracy.

3.3.3.2 Compliance for user testing

While not the focus of the Technical Baseline Maturation element of OWS-2, a suite of tests can be developed and made available to the user community that goes beyond the strict guidelines of certification testing. The purpose of these tests is to allow users to verify their compliance to optional aspects of a specification and to test their servers against their own data. These testing regimes do not cover every possible feature of a server that may be of interest. They are mainly informative in nature and advance the goal of full interoperability. Organizations interested in working in this area can indicate so in their response to the RFQ.

3.3.3.3 Specifications covered

The OWS-2 initiative contemplates the development compliance tests following specifications:

- Web Map Service (the focus will be on version 1.2, extending and modifying the existing test for WMS 1.1.1). This is a Tier-2 requirement, meaning that cost-share funding is not available at this time. In the update to this RFQ, it is anticipated that this will be upgraded to a Tier 1 requirement.

- Web Map Service, using Styled Layer Descriptor (the focus will be either version 1.0 or 1.1 depending on the status of the SLD specification at the time of OWS-2 execution). This is a Tier-2 requirement, meaning that cost-share funding is not available. In the update to this RFQ, it is anticipated that this will be upgraded to a Tier 1 requirement.
- Web Feature Service (the focus will be on version 1.1, if it exists during the time of OWS-2 execution, extending and modifying the existing test for WFS 1.0). This is a Tier-1 requirement, meaning that cost-share funding is available.
- Web Coverage Service (the focus will be on version 1.1, or the DIPR corresponding to such a revision during the time of OWS-2 execution). This is a Tier-1 requirement, meaning that cost-share funding is available..
- Catalog Service – Web Profile (the focus will be version 2.0, if it exists during the time of OWS-2 execution). This is a Tier-1 requirement, meaning that cost-share funding is available.
- Location Services (the focus will be on the 1.1 revisions being worked during the time of OWS-2 execution). This is a Tier-2 requirement, meaning that cost-share funding is not available. In the update to this RFQ, it is anticipated that this will be upgraded to a Tier 1 requirement.
- Geography Markup Language 3.0 Schema and Instance Validation Capability(the focus will be on version 3.0, or the DIPR corresponding to such a revision during the time of OWS-2 execution). This is a Tier-2 requirement, meaning that cost-share funding is not available. In the update to this RFQ, it is anticipated that this will be upgraded to a Tier 1 requirement.

3.3.3.4 Geography Markup Language Validation

In addition to compliance testing for OGC Web Services, the OWS-2 initiative is interested in also making available a Geography Markup Language (GML) validation engine for version 3.0 of the GML specification. At this stage there are no certification plan for GML, however, a validation capability will definitely provide the basis for such a program. This is a Tier-2 requirement, meaning that cost-share funding is not available.

3.3.4 Design decisions

3.3.4.1 Use Cases

3.3.4.1.1 Implementer pursuing product certification

The primary use case under consideration is that of a software vendor wanting to advertise a software product as being compliant to a particular OGC specification. Interested parties in this scenario are: the OGC, which wants to protect the use and value of its intellectual property; the candidate vendor, who wants to make certain statements about their product’s functionality; and software buyers, who need accurate information about product functionality. OGC must continue to develop a testing regime that meets these basic needs. We see this as ensuring that an OGC-compliant product on the market implements all mandatory interfaces in the specification and emits responses that are valid against the specification’s XML schema or Document Type Definition(s).

3.3.4.1.2 Implementer testing pre-release software

The compliance engine should also be useful in a pre-release scenario, to aid developers in ensuring that their software is compliant as it is being developed, and also to test vendor-specific capabilities. These needs are addressed by allowing a user to write their own tests, as long as they comply with the test script schema. Pre-release testing is also facilitated by allowing accounts to be created that are not identified with a particular company. This allows companies to use the test engine without feeling that they are under evaluation in any way.

3.3.4.1.3 Implementer testing a service

Their will be scenarios where an entity wants to test software in situ, i.e. a particular software, hardware and data combination. While the certification test suite proposed under this project requires a particular data sets, compliance may still be tested by developing test scripts that work with different data. Proposers who wish to perform such tests must be explicit about this so that resources can be arranged to help in the installation of such tests within the OGC Web-based Compliance Testing Engine software.

3.3.4.2 Dynamic testing

During the CITE project it was decided not to pursue the ability to test compliance of a service in a situation where the data and the capabilities are not known beforehand. Meanwhile, the capability was developed as is ready for beta testing within the OWS-2 framework.

3.3.4.3 The Concept of Test Suites

The set of documents required to test that a server completely implements a particular OGC specification is called a test suite. This project is potentially developing test suites, for the OGC Web Services listed above. A test suite consists of many elements, which are described in the table below. The compliance engine, which is at the core of all compliance tests, knows how to execute tests that fit this architecture. It does not inherently know anything about particular specifications. Compliance scripts are written to test specifications, and they are executed “blindly” by the engine.

Table 1: CITE Terminology

Term	Explanation
Test Script	The set of Test Cases for testing against a single OGC specification, e.g. WMS. They will be organized in a hierarchy that matches the structure of the specification to be tested. A single test script when installed in the compliance test system acts as a test suite for the target specification. Several test scripts may be installed at the same time.
Test Case	This is the self contained XML document that holds all of the information that a single test needs in order to function within the test system. This will include such things as assertions, test strategies, test methods, expected results etc.
Assertion	An assertion is a statement of required behavior that is derived from the specification. A corresponding test method is designed to determine whether the statement is true or false for the implementation under test. All assertions will be written to POSIX 1003.3 standard (http://standards.ieee.org/reading/ieee/std_public/description/posix/13210-1994_desc.html).
Test Strategy	This is an explanation of the test method, how it works and what the expected result will be. This is in a human-readable form so that an operator can check what the device should

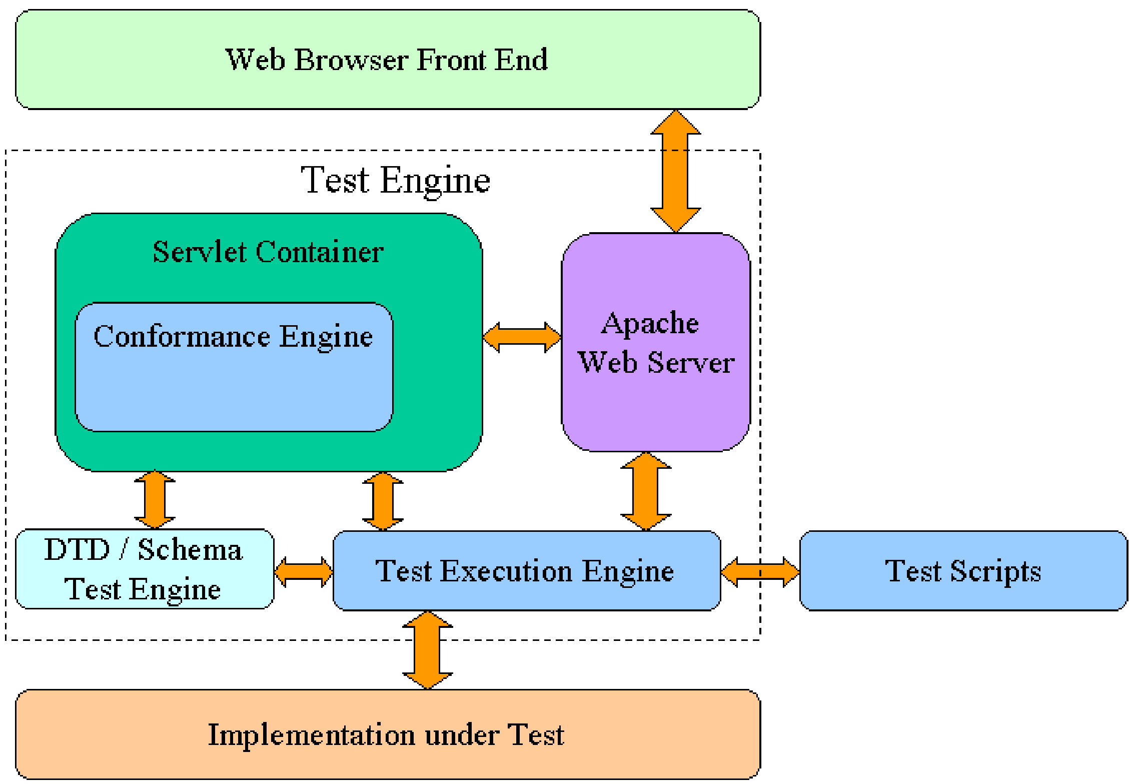
	actually be doing.
Test Method	This is the part of the test document that defines the actual test that will be executed by the device under test. It will use specially defined XML markup to allow the compliance engine to interpret the test correctly.
Expected Result	This is the result that the test method must get back from the result of the test in order for the test script to pass.

3.3.4.4 Test Data

A set of data should be designed that can be used for certification testing of OGC Web Services.

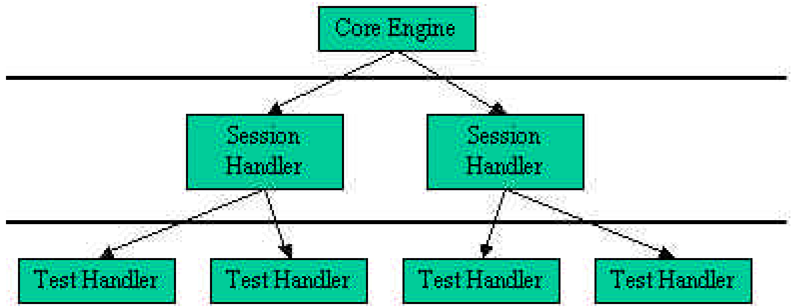
3.3.5 Program components

The Compliance Engine for an OGC specification will accept one or more Compliance Test Suites. These suites will be run against a candidate OGC-compliant software component. The results of the test will be codified in a Compliance Test Report, indicating success or failure and as much detail as possible regarding the reasons for such.



At the highest level the testing system architecture is broken into 3 layers as shown in Figure 3.

Figure 3: Compliance Test System Architecture



This diagram indicates that there are three distinct layers in the system:

3.3.5.1 Core Engine

The engine understands about Test Session handlers, it understands about the underlying file system, how to save Test Sessions, how to find Tests, how to save and load logs. It also contains an environment; this is global to all Test Session handlers. The core engine also includes the core Servlets that are used as the main interface for the User. The core engine also understands how to manage test sessions; this includes modifying information held within the test sessions.

3.3.5.2 Test Session Handlers

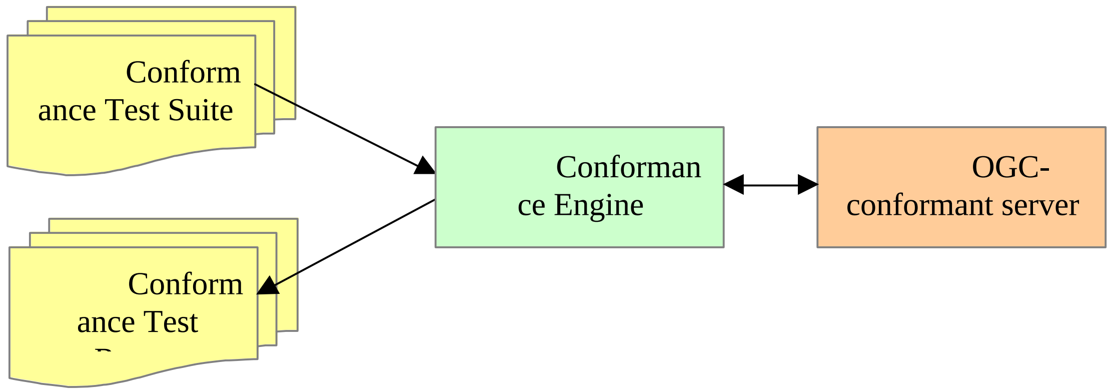
The purpose of the Test Session handler is to give the specifics of how the Test Session should be managed, for example, how the Test Session results should be displayed to the User, or how a Test Session is created. These are things that the Core engine does not and should not know about, each Test Session handler provides the "real" method of handling the Test Session for a particular type.

3.3.5.3 Test Handlers

Test Session handlers are used to manage the Test Sessions themselves but when it comes to executing actual Tests the concept of Test Handlers is introduced, these are specific pieces of code that understand about a specific test and understand how to transform that test so that it is suitable for passing and execution by a Client. They also serve the purpose of transforming the test source into a format that is suitable for displaying to the User, so when the User requests to see the source of the test a chunk of text is returned that nicely formats the test source into something suitable for reading. Test Handlers also give a way of dealing with the logs that are generated when a test is executed, the Core engine knows how to load and save the logs but the Test Handler provides the internals of what information is in the log and how it is displayed to the User. Finally when the User submits some test results back to the system the results processor part of the Test Handler processes the test results sent back and generates a log.

The thick horizontal lines on the diagram above indicate that there are interfaces between each layer. These are well-defined interfaces that Test Session handlers and Test Handlers must comply to—they are the contracts that allow the system to work. Each layer *only* goes through those interfaces, it does not know about the internals or each layer, only the interfaces.

As the diagram indicates, the core engine has a number of Test Session handlers "plugged into" it, and there are a number of Test Handlers "plugged into" each Test Session handler. For the purposes of the OGC test suite, there will only be the need for one session handler, but there may be multiple test handlers, for WMS and WFS for example.



3.3.5.4 Compliance engine

The compliance test engine is the part of the system that pulls all of the constituent parts of the test suite together. It controls what is sent to the web browser front end, which tests are run, what content is sent to the server, etc. It is written in Java, and runs within any compliant Servlet 2.2 container under the Apache web server. Both Apache and Servlet 2.2 containers are freely available. All tests and test information (test specifications, strategies, expected results, etc.) comply to a defined XML schema.

The engine exists to provide core services and functional items, for example, basic viewing and management of a Test Session, viewing of Test content, logging, helpdesk, email, etc. The engine calls separate test session handlers and test handlers (execution engines) that allow the different types of tests to be executed. For this, the core compliance engine defines certain interfaces so that it can act as a ‘plug and play’ engine. If, in the future, testing is required for another type of service, we can simply write new handlers that deal with this without the need to recompile the core compliance engine. We simply create the new handlers and update an XML file that defines all of the parameters that the core engine needs to be able to execute the new tests using the new handlers. The session handlers then read in the test documents and deal with them as appropriate.

The test engine allows users to have their own password protected testing area so that they can test their service in a private environment. This also allows multiple users to use the test suite concurrently.

The compliance test system is written to allow multiple operators, and test sessions can be used concurrently.

3.3.5.5 Compliance Test Scripts

A test script consists of a set of Test Cases organized as a hierarchy of separate files in a directory structure that maps directly to the specification under test. Each Test Case will be a self contained XML document so that it is simple to add, remove, and update individual tests as required. A test script consisting of a set of test cases will be developed and delivered for each specification to be tested. The test case XML schema will also be developed and delivered to facilitate subsequent test development by OGC and its technical working groups.

Test Cases use test assertions that follow the IEEE Standard 1000.3. Assertions are written against each identifiable compliance requirement in the specification and will be captured in the Test Cases in such a way that allows them to be filtered and displayed as a stand-alone document. All of the test cases for a given specification will be processed to create a single assertion document and hence one set of assertions will be produced for a given OGC Web Service specification.

The IEEE 1003.3 standard allows assertions to be classified as testable or un-testable as well as mandatory or conditional. This is referred to as the Class of the assertion. If an assertion is testable, the test case will also include a description of the strategy for the test method. If the assertion is un-testable, the test case will include rationale as to why the test is classified as un-testable. Un-testable assertions provide a way to explicitly capture information in the specification that is important, but is unable to be tested by the current system, offering guidance for future enhancements to the system, and/or modifications to the specification.

The OWS-2 initiative will define the test scripts required for certification. However, this is only a subset of the potential capabilities of a service. The system allows users to design their own test suites, combining existing scripts or those developed under OWS-2 with their own to provide more complete system compliance tests. While the actual test execution engine is commercial software, the API and XML schema for writing test scripts is public and open, so that extensibility is achieved.

3.3.5.6 Reference Implementations

The intent of this project is to spur the adoption and diffusion of OGC specifications into the market. The main method of accomplishing this goal is to develop a system through which users can be assured that the software they buy correctly implements the OGC specifications. A complementary strategy is to provide the community with a suite of software that can be used as a reference. An excellent way of providing reference implementations is via open source software under the GNU Public License, since the code can be examined in full without limiting the ability of companies to develop commercial products. For these reasons, an open source Web Feature server and client, and a Web Map server were developed for the CITE initiative and reference implementations for the same specifications mentioned above will potentially be developed under OWS-2.

This suite of software will be available for download from the CITE portal. It is envisioned that the community will use this software to:

- Quickly stand up data services
- Spur the development of other products
- Spur the development of new product features
- Help troubleshoot compliance problems in developing software

3.3.6 Compliance Testing References

Open GIS Consortium, Inc., Compliance Testing Program,
http://www.opengis.org/resources/testing/docs/03-085r1.pdf

The Open Group, Conformance and Interoperability Test and Evaluation
(CITE) Initiative: Test Script Notation V1.2,
http://cite.occamlab.com/instructions/files/TestScriptNotation_1_2.doc

The Open Group, CITE Test Engine User Guide,
http://cite.occamlab.com/instructions/files/OGCUserGuide.pdf

3.4 Information Interoperability (II)

3.4.1 II Scope

The *Information Interoperability* theme of OWS2 focuses on the interoperable expression, exchange and access of geographic information within and across information communities (Information communities are groups that share common geographic terms and common spatial feature definitions). The *Information Interoperability* activities will explore ways for enhancing common ways for representing, sharing and processing of information between collaborating entities to maximize opportunities for exchange and reuse of geographic information.

In order to support these goals, the OWS2 *Information Interoperability* activities will focus on enhancing current interface and encoding specifications while emphasizing practical implementations of these interfaces and their testing in a variety of clients (including OWS2 Integrated Clients).

This section of the RFQ outlines the service, client and encodings requirements needed to support the following *Information Interoperability* activities

- Development, testing and implementation of GML 3 and GML 2.1 application schemas to support NIMA Mission Specific Data Sets (MSDSs) as described in the attached fictitious scenario provided by NIMA (see Appendix B).
- Enhancement, testing and implementation of WFS services in support of the attached fictitious scenario provided by NIMA (including support for GML 3 Level 0).
- Evaluation of GML 3 schemas with respect to the development and transformation of community application schemas into public GML schemas conformant to the ISO 19109 Rules for Application Schema.
- Investigation and testing of the use of semantic repositories in support of schema transforms within information communities.

3.4.1.1 Problems and Objectives

To this date, many information communities are still having problems performing efficient and meaningful exchanges of information. These problems include

- Information communities often use different and proprietary data models and formats to encode their geographic information.
- Information communities face some difficulties when exchanging data and effectively trying interpret that exchanged data in meaningful ways.
- Members of information communities have difficulty locating schemas applicable to their applications and environments.
- Access to data is often provided through proprietary interfaces.
- There is sometimes a lack of agreement (and documentation) on semantic meaning of community-defined features.

In order to address these problems, this thread aims at

- Enabling the interoperable expression, exchange and access of geographic information across information communities.
- Enhancing common ways for representing, sharing and processing of information between collaborating communities.
- Maximizing opportunities for exchange and reuse of geographic information.

- Leveraging web services for finding, publishing and accessing that data.
- Testing the feasibility of creating simplified and semi-complex community application schemas based on GML 3, and simplifying the process of generating mission-specific application schemas.
- Ensuring that application schemas developed by information communities are compliant with ISO19109.

3.4.1.2 Planned Activities & Associated Requirements

3.4.1.2.1 Information Interoperability – Development, Testing and Implementation of GML 3 and GML 2.1 Application Schemas[2] in Support of the Attached Fictitious NIMA Scenario.

This section refers to the narrative fictitious scenario generated by NIMA (see Appendix B) in order to help demonstrate the interfaces and encodings that are to be worked on during OWS2. The goal is to begin using specifications enhanced and implemented during OWS2 to help NIMA/GeoScout mission in transforming the NIMA Information Technology Infrastructure over the next 5 to 10 years.

The goal of this activity is to test the feasibility and utility of using the Geography Markup Language (GML) as a transfer format for NIMA's Mission Specific Data Sets (MSDSs). GML is an XML encoding for the transport and storage of geographic information, including both the geometry and properties of geographic features. GML utilizes the OGC Abstract Specification geometry model which has been harmonized with the ISO geospatial geometry model. Unlike a simple geometry model, the GML specification also includes the ability to handle complex properties. Work is also underway to harmonize this complex property model with the ISO efforts in the same area. The GML 3 implementation specification was approved and released in February of 2003.

This test involves data with simple content (i.e. two-dimensional geometries and associated attribute data) as well as more complex data (i.e. 2.5-dimensinal data with selective geometries and associations, and data with temporal content). **The goal of this activity is to generate GML 3.0 Application Schemas that can support both simple and complex content.** In order to support the requirement of creating application schemas requiring differing levels of complexity simpler and less time-consuming, this activity should focus on **developing and testing an XSLT-based mechanism (a Schema Subsetting Tool) for generating profile schemas by selecting particular elements from a base schema as required in a mission-specific schema.** Additional mission-specific elements can then be added to meet the mission-specific needs. At a minimum, the following GML 3 schema modules should be exercised and assessed: Feature, Geometry, Coordinate Reference Systems, Topology, Temporal and XLinks.

The Mission-Specific Application Schemas should

- Accommodate the ability to add z-values and topology elements and utilize additional GML 3 capabilities such as XLinks, dynamic features, observations, roadway restrictions and temporal elements.
- Allow a feature to have any number of geometric properties (as in the Airport facility example in the scenario. In this case, each feature is encoded with multiple geometric representations with each representation containing more details than the previous one with the area representation containing the most detailed information. Changes to features are tracked based on time and changes).
- Handle simple geometric shapes 0 to 2.5 D (3D solids are out of scope).
- Support only LineString type curves (CurveSegments).
- Contain the Coordinate Reference System to be used for all features and geometries in both the base and secondary application schemas, the original set of 49 features and their attributes, and the geometry and topology schemas in use with the 49 features (see paragraph 3.4.3.3).

---

[2] An application schema provides the formal description of the data structure and content required by one or more applications. It contains the description of both geographic data and other data.

The attached scenario (see Appendix B) lists additional specific types of information that the application schemas are required to support for transferring the MSDSs using GML. A detailed look at that scenario and a validation of the underlying requirements will be performed at the initiative kickoff and during the first weeks of the initiative. NIMA will be providing data to be encoded using the defined application schemas.

Due to the infancy of GML 3 and the small number of current clients that support that encoding, this *Information Interoperability* activity shall also pursue the **development of equivalent GML 2.1 Application Schemas**. Insights on identifying automated and efficient approaches for transforming the schema from GML 3 Level 0 to GML 2.1 (including mapping the right subset of GML3 into GML 2.1, and dropping or transforming the GML3-only features) are solicited.

Deliverables

- GML 3 and GML 2 Base and Mission-Specific Application Schemas
- Data encoded using the developed application schemas
- GML change proposals (as needed)
- Working implementation of Schema Subsetting Tool
- DIPR detailing process of creating schemas and the associated instances
- Documentation for the Schema Subsetting Tool

3.4.1.2.2 Information Interoperability - Enhancement, Testing and Implementation of WFS Services In Support of Serving and Interacting with Data Encoded Using the Application Schemas Developed For the Fictitious NIMA Scenario, in addition to Support of Schema Translation Activities.

The fictitious NIMA Scenario requires that **the data encoded be made available through Web Feature Services**. Web Feature Services support the query and discovery of geographic features. In a typical scenario, a WFS delivers GML representations of simple geospatial features in response to queries from http clients. Clients (service requestors) access geographic features that are needed for an application. The client generates a request, posts it to a WFS instance (a WFS server on the web). The WFS instance executes the request, returning the results to the client as GML. A GML-enabled client can manipulate or operate on the returned features.

To support the NIMA fictitious scenario, the GML-enabled Integrated Client will be developed. This Integrated Client Reference Implementation will follow the design details of the Integrated Client DIPR (i.e., OGC Document Number 02-051r6) and the TIE work that was done on in OWS Phase 1.2. The Reference Implementation shall work correctly with the Reference Implementations for WMS, WFS, WCS, and Catalog Services – Web Profile. The integrated Client shall be provided as open source and shall be developed in a way such that it can be deployed on multiple platforms.

To support the Information Interoperability thread, the Integrated Client should be able to update the encoded data and upload the changes back to the WFS on a daily basis. **Robust implementations of such clients and WFS services are required for this activity**. Enhancements and documentation of best practices and lessons learned, particularly in the areas of XLink Processing (as proposed in OGC02-063) and WFS topology queries, are expected under this activity. These should be documented in the form of change proposals as needed.

An IPR (or set of IPRs) shall be developed that include the Architecture of the Integrated Client RI, Installation, and User Guide.

This activity also involves conducting a series of hands-on Technology Integration Experiments (TIEs) to **test the support for the Level 0 GML 3 Profile by vendors' WFS's implementations**. The Level 0 GML 3 profile was introduced as part of OGC's Critical Infrastructure Protection Initiative Phase I (CIPI1) in order to help lower the implementation bar for any organization wishing to commit time and resources for developing an interoperable and simple OGC WFS client application. It does that by rigidly defining a simplified representation of geographic features (albeit with limited expressiveness) using a restricted subset of GML3.

Finally, this activity will also involve **augmenting WFSs to support schema translation from a local schema to a standard application schema**. Such translation can contribute to minimizing the semantic divide between communities (also explored in semantic mapping section of the *Information Interoperability* theme). Such translation activity was first attempted as part of OGC's Geospatial One-Stop Transportation Pilot (GOS-TP) and OGC's CIPI 1.2 initiative, where approaches were developed to allow a Web Feature Service to accept requests for features in a national schema, connect to data defined by a local schema and return features in the national schema.

OWS 2 *Information Interoperability* activities in this area will focus on overcoming the difficulties identified during the GOS-TP and CIPI1.2 initiatives, including (but not limited to)

- Accommodating situations where features and properties in one schema aren't represented in another
- Accommodating cases where schema mappings were not one-to-one, necessitating the creation of multiple queries from a single query or mapping multiple properties to a single property (also accommodating cases where mandatory properties in the global schema do not have equivalents in the local schema)
- Mapping multiple features types into a single feature type, or transforming relatively flat data into more hierarchical data (while inferring missing properties)
- Accommodating cases where remote WFSs support different subsets of interfaces, or support only subsets of the global schema (for example, by not supporting some of the optional properties)
- Experimenting with approaches for consistently handling the absence of an optional attribute from a requested feature

Deliverables

- WFS services implementations (including XLink processing, translating WFSs, and support for Level 0 GML 3)
- Integrated client implementations
- WFS change proposals to support the inclusion of the Level 0 GML3 in the WFS suite of interfaces (as needed)
- DIPR detailing schema translation approaches and findings
- DIPR summarizing overall activity (including capturing changes to WFS and approaches to resolving topology queries)

3.4.1.2.3 Information Interoperability - Evaluation of GML 3 Schemas with Respect to the Development and Transformation of Community Application Schemas into Public GML Schemas conformant to the ISO 19109 Rules for Application Schema.

The objective of this OWS2 *Information Interoperability* activity is to evaluate GML 3 schemas with respect to the development and transformation of community application schemas that conform to the ISO 19109 Rules for Application Schemas.

ISO 19109 defines rules for creating application schemas in a consistent manner to facilitate the acquiring, processing, analyzing, accessing, presenting, and transferring of geographic data between different users, systems and locations. It contains concepts required to describe types of features, rules for the application modeling process, for application schema development in UML (Unified Modeling Language), as well as for creating domain profiles of standard schemas and use of metadata schema.

Compliance with ISO 19109 is required by ANSI/INCITS L1 for FGDC's Geospatial One-Stop Initiative. The Geospatial One-Stop initiative is one of the 25 OMB e-government initiatives intended to enhance government initiative. As part of it, a web-based portal (www.geodata.gov) was developed to accelerate the development implementation of the National Spatial Data Infrastructure (NSDI) and includes state, local, and tribal governments along with private sector and academia as participants. In addition, a collaborative process to develop data content standards was initiated to provide the building blocks for a National Geographic Information Network, ensuring consistency among data sets and allowing governments to share data and integrate multiple sources of information – a key value of geographic information. Led by individual agency leads, teams of federal, state, local and private representatives have been working to create draft standards to provide a common structure for easier access to data.

The FGDC framework is a collaborative effort to create a widely available source of basic geographic data. The framework's key aspects include seven themes of digital geographic data that are commonly used. For OWS2, the focus will be on two of these themes: the National Hydrography Draft Content Standard, and the Geographic Information Framework Data Content Standards for Transportation: Roads.

Using these two themes, the OWS 2 *Information Interoperability* activities will involve

- Understanding what ISO19109 entails and whether compliance is achievable for the two selected themes
- Identifying what needs to be done in order to make the selected themes compliant
- **Justifying** any changes that need to be applied to the existing models (this is an opportunity to make changes to those themes, but those changes need to be properly justified)
- Experimenting with directly translating the UML models of those themes to GML using tools such as the UML-to-GML Application Schema (UGAS) open source tool
- Identifying practical limitations and constraints, if any, through **implementation examples**

The process of making a community application schema compliant, and any changes that need to be applied should be justified and documented.

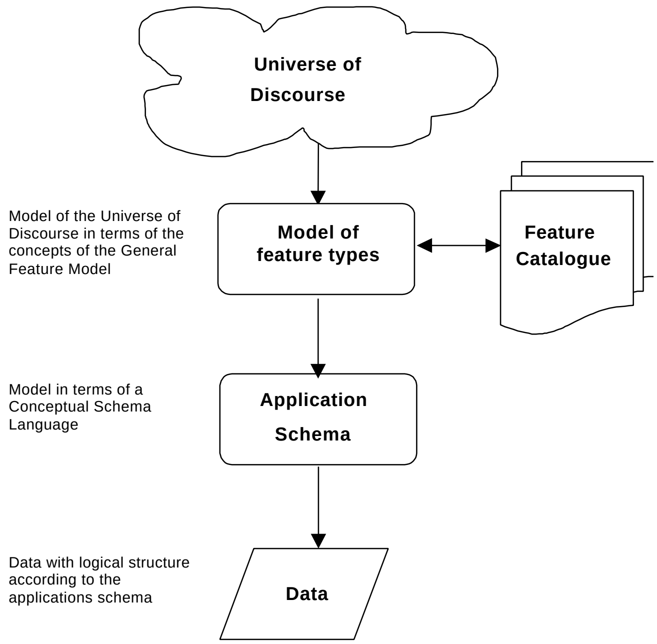
Deliverables

- DIPR documenting process of making a schema 19109 compliant
- Change proposals to existing models for at least Roads and Hydrology
- GML schema implementations and sample instances of new models

3.4.1.2.4 Information Interoperability - Investigation and Testing of the Use of Semantic Repositories in Support of Information Communities Schema Transforms, including Exploration of the Use of Feature Catalogues as Semantic Repositories, and Methods for Expressing Them.

For a set of data to be used or exchanged successfully, information community members must have a common understanding of that data. Key to this common understanding is the unambiguous identification of the feature types in the data, including how they are encoded.

This OWS 2 *Information Interoperability* activity involves **the investigation and testing of the use of semantic repositories to capture that common understanding of data within an information community**. Such repositories are needed before application schemas can be developed by communities to make sure the semantics of the application domain are agreed on and well-captured. The goal is to find a way for expressing such semantic repositories to enable information community members to have a shared understanding of the contents of geographical data sets, thereby promoting the dissemination and interoperability of data, and reducing confusion and misinterpretation of data.



As indicated in the above diagram, this can be done by using feature catalogues for such semantic repositories. Hence, **this activity involves investigating methods for expressing such feature catalogues in different communities**, **and making them web-accessible and readily available to the geographical information community.** A feature catalogue defines the types of geographic features one would find in a data set, together with their attributes, relationships and behavior.

ISO 19110 Methodology for Feature Cataloguing describes a standard framework for organizing and reporting the classification of real world phenomena in a set of geographic data. It also specifies that a feature catalogue shall present its abstraction of reality as a defined classification of phenomena. The catalogues may consist of unstructured, unordered list of feature types, but it will then be difficult to conceptualize all the feature types simultaneously. A classification scheme provides some structure and order to the catalogue by grouping together related feature types, normally into some form of a hierarchy. It makes it easier to find the required feature type by drilling down through a well structured hierarchy than it is to find in a long list of feature types.

Since ISO 19110 only focuses on a standard organization framework and NOT on the specific encodings or access interfaces needed for standing up a feature catalogue, **this activity will focus on specifying these encodings and interfaces**. In the process, **it is required to leverage existing OGC encodings and service interfaces** (such as the Catalog Service or Web Object Service).

OWS 2 implementations should allow information community members to build, view and edit feature catalogues and classification schemes, and all the elements in them, and to export and import feature catalogues and classification schemas using a simple structure.

Deliverables

- Feature catalogue implementations (populated with FGDC Framework feature types)
- DIPR documenting approach to designing feature catalogue (rules, encodings, interfaces based on existing OGC interfaces)
- Change proposals to WOS or Catalog service (if needed)

3.4.2 II Enterprise Viewpoint

3.4.2.1 Use Cases

Use Case 1

A fictitious use case narrative scenario has been provided by NIMA, and is appended to this *Information Interoperability* section of the OWS2 RFQ Annex B.

Use Case 2

This *Information Interoperability* use case revolves around facilitating the process followed by information community members to develop local application schemas for their community. These members first discover one or more feature catalogues that house the feature types (and their relationships) pertaining to the FGDC framework theme(s) that are the most relevant to their community. The feature catalogue(s) enable those community members to browse through master lists of feature types, each described using unique unambiguous feature type names, numeric feature type codes, concise unambiguous definitions, agreed-upon attributes, relationships with other features, etc. The feature catalogue(s) also enable community members to structure relevant features into classification systems that can best meet their particular needs. Using the feature catalogue(s), the information community members can pick the subset of features they need to build their local application schemas. The catalogs may also provide a framework for these community members to add their own specialist feature types.

For a real-world application of feature catalogues, the reader is referred to the South African National Spatial Feature Catalogue study available at
http://www.nsif.org.za/standards/SA_Catalogue_Standard_final.pdf.

3.4.3 II Information Viewpoint

3.4.3.1 Geography Markup Language (GML)

GML is an XML encoding for the transport and storage of geographic information, including both the geometry and properties of geographic features. GML utilizes the OGC Abstract Specification geometry model which has been harmonized with the ISO geospatial geometry model. In addition to the simple geometry model, the GML specification also includes the ability to handle complex properties. Work is currently underway to harmonize this complex property model with the ISO efforts in the same area. The GML 3 implementation specification was approved and released in February of 2003.

GML2.1 (http://www.opengis.org/docs/02-069.pdf) is concerned with representing simple features (features whose geometric properties are restricted to simple geometries for which coordinates are defined in two dimensions and the delineation of a curve is subject to linear interpolation).

GML3 (http://www.opengis.org/docs/02-023r4.pdf) addresses certain needs that were not addressed or adequately met by GML2.1, including

- representing geospatial phenomena in addition to simple 2D linear features, including features with complex, non-linear, 3D geometry, features with 2D topology, features with temporal properties, dynamic features, coverages, and observations;
- providing more explicit support for properties of features and other objects whose value is complex;
- representing spatial and temporal reference systems, units of measure and standards information;
- using reference systems, units and standards information in the representation of geospatial phenomena, observations, and values;

- representing default styles for feature and coverage visualization;
- conforming with other standards, including ISO DIS 19107 Geographic Information – Spatial Schema, ISO DIS 19108 Geographic Information – Temporal Schema, ISO DIS 19118 Geographic Information – Encoding, ISO DIS 19123 Geographic Information – Coverages

According to the Web Feature Services (WFS) specification, GML and XML-Schema must be used to describe feature types returned by a WFS. This implies that any client application must be able to parse and interpret schemas in XML-Schema. However, since XML-Schema is a large and complex specification designed to satisfy a large set of requirements, implementing such parsers has been found to be a significant implementation hurdle for organizations.

Consequently, as part of OGC's Critical Infrastructure Protection Initiative Phase I (CIPI1), a Level 0 GML 3 Profile has been proposed to address that hurdle. GML 3 Level 0 is a basic GML 3 application schema that helps in lowering the implementation bar for any organization wishing to commit time and resources for developing an interoperable and simple OGC WFS client application. It does that by rigidly defining a simplified representation of geographic features (albeit with limited expressiveness) using a restricted subset of GML3. The proposed Level 0 GML 3 Profile is documented in the Interoperability Program Report OGC03-003r7.

3.4.3.2 FGDC Framework Data

The FGDC framework is a collaborative effort to create a widely available source of basic geographic data. As part of that effort, a collaborative process to develop data content standards was initiated to provide the building blocks for a National Geographic Information Network, ensuring consistency among data sets and allowing governments to share data and integrate multiple sources of information – a key value of geographic information. Led by individual agency leads, teams of federal, state, local and private representatives have been working to create draft standards to provide a common structure for easier access to data.

The FGDC framework's key aspects include seven themes of digital geographic data that are commonly used. The Framework themes (http://www.fgdc.gov/framework/framework.html) are being developed using the ISO TC211 Standardized Conceptual Schemas (the ISO 19100 series of standards). Table 2 provides a brief description of each Framework theme.

Cadastral	Right, title, and interest in real property, including above, surface, and below ground and water.
Digital Ortho Imagery	Georeferenced images of the Earth's surface, with the geometric characteristics of a map, and image qualities of a photograph
Elevation	Georeferenced representations of terrestrial and bathymetric surfaces, natural or manmade, which describe vertical position above or below a datum.
Geodetic Control	Common, consistent, and accurate reference system for establishing coordinates for all geographic data.
Governmental Unit	Content requirements for the collection and interchange of governmental unit (GU) and other legal entity boundary .
Hydrography	Surface water features such as lakes, ponds, streams and rivers, canals, oceans, and coastlines. Also, network connectivity and direction of flow.

Transportation	Geographic locations, interconnectedness, and characteristics of the transportation system: road, railroad, transit, and waterway networks, plus airport facilities

Table 2 FGDC Framework Themes

3.4.3.3 NIMA Framework Data

3.4.3.3.1 USIGS Conceptual Data Model

As part of the United States Imagery and Geospatial Information System (USIGS) effort, NIMA has developed the USIGS Conceptual Data Model (UCDM) which describes the common information used by the business elements of USIGS. UCDM contains three components: a fully attributed entity relationship diagram, a data dictionary, and a representation of business rules and relationships between data. Table provides a brief description of the eight major themes (or volumes) of UCDM.

Metadata	Metadata and model views of events, coordinate reference systems, feature representations, names, geometry, topology, security and user profiles
Imagery	Images, sensors/platforms, capture information, imagery targets, and image processing information.
Air Transportation	Transporting people and cargo through the air and on the ground at airports.
Ground Transportation	Cultural features fixed to the earth’s surface that provide for the transportation of cargo and people.
Water Transportation	Navigating over the portion of the Earth’s surface that is covered with water.
Water Features	Water that is inland of the coastline and shorelines, including shore protection structures.
Cultural Features	Non-transportation, man-made features on the Earth.
Physiography	Naturally occurring features such as rock formations, snow and ice, relief and bottom characteristics, and vegetation.

Table 2 USIGS Conceptual Data Model Themes/Volumes

The UCDM documentation contains an appendix that maps UCMD to the NIMA Profile (the subset of the FAAC used by NIMA).

3.4.3.3.2 NIMA FAAC Profile

NIMA’s Profile of the Feature and Attribute Coding Catalogue (FAAC) is a subset of the FAAC that defines a common coding glossary for use in NIMA Vector Product Format (VPF) data sets (products). The

FAAC consists of part 4 of the Digital Geographic Information Exchange Standard (DIGEST)[3] available at http://www.digest.org/html/gp4a-toc.htm. The FAAC is a dictionary of features, attributes and attribute values and is designed to provide a means for encoding real-world entities or objects and their associated properties or characteristics for the purpose of an orderly exchange of digital geospatial information between organizations. FAAC organizes features in the following categories

A Culture	B Hydrography	C Hypsography
D Physiography	E Vegetation	F Demarcation
G Aeronautical	I Cadastral	S Dataset-specific

The attached NIMA Scenario lists the following extended set of FAAC features and attributes that need to be supported for this theme:

- NIMA enhanced extraction list- Features
  - Buildings (school, hospital, police station, fire station)
  - Utilities (water & electric)
  - General Miscellaneous Feature (ZD019)
  - Geographic Information Point (ZD012)
  - Land Subject to Inundation (BH090)
  - Safety Fairway (FC170)
- NIMA enhanced extraction list- Attributes for
  - Roads
  - Bridges
  - Buildings
  - General Miscellaneous Features
  - Geographic Information Points
  - Land Subject to Inundation

3.4.3.4 ISO 19109

ISO 19109 defines rules for creating application schemas in a consistent manner to facilitate the acquiring, processing, analyzing, accessing, presenting, and transferring of geographic data between different users, systems and locations. It contains concepts required to describe types of features, rules for the application modeling process, for application schema development in UML (Unified Modeling Language), as well as for creating domain profiles of standard schemas and use of metadata schema.

3.4.3.5 ISO 19110

ISO 19110 Methodology for Feature Cataloguing describes a standard framework for organizing and reporting the classification of real world phenomena in a set of geographic data. It also specifies that a feature catalogue shall present its abstraction of reality as a defined classification of phenomena. A feature catalogue contains definitions and descriptions of the feature types, feature attributes, and feature associations occurring in one or more sets of geographic data, together with any feature operations that may be applied. The feature catalogue provides the definition of geographic features at the type level, not the recording and representation of individual instances of each type.

[3] DIGEST is an international exchange standard for digital geographic information issued under the authority of the Digital Geographic Information Working Group (DGIWG).

3.4.4 II Computational Viewpoint

3.4.4.1 Application services

OGC Service Framework (OSF) services are accessible from Application Services operating on user terminals (e.g., desktop, notebook, handset, etc.) or servers that have network connectivity. Users may use Application Services to access Catalog, Portrayal, Processing and Data Services, depending upon the requirements and designed implementation of the application. Application Services commonly, but not necessarily, provide user-oriented displays of geospatial content and support user interaction at the user terminal.

Examples of OSF applications services include: Discovery Application Services, Map Viewer Application Services, Value-Add Application Services, Imagery Exploitation Application Services, Sensor Web Application Services, Location Organizer (LO) Application Services, Mobile Location Services. More information about OSF and Application services can be found in OGC's Reference Model (http://www.opengis.org/docs/03-040.pdf).

3.4.4.2 Web Feature Service

The Web Feature Service (WFS) supports INSERT, UPDATE, DELETE, QUERY and DISCOVERY of geographic features. WFS delivers GML representations of simple geospatial features in response to queries from HTTP clients. Clients access geographic feature data through WFS by submitting a request for just those features that are needed for an application. A WFS can either be a basic WFS (a READ-ONLY WFS), which implements the GetCapabilities, DescribeFeatureType and GetFeature interfaces, or a transaction WFS, which, in addition to supporting all the interfaces of a basic WFS, implements the Transaction interface (and optionally the LockFeature interface).

The latest WFS approved specification can be found at http://www.opengis.org/docs/02-058.pdf.

3.4.4.3 Catalog Services for the Web

OSF Catalog Services for the Web (CS/W) provide a common mechanism to classify, register, describe, search, maintain and access information about resources available on a network. Resources are any network addressable instances of typed data or services.

A discussion paper about the Catalog Services for the Web profile can be found at http://www.opengis.org/docs/02-087r3.pdf.

3.4.4.4 Web Object Service

The Web Object Service (WOS) defines a set of base XML types from which object access and management services, such as the Web Feature Service or the Catalog Service for the Web, may be derived. The Web Object Service essentially represents a standard interface to an object repository.

Within the OGC, the Web Feature Service (WFS) Implementation Specification has been used as the basis for defining of a number of library or data management service interfaces. These include the interface for the Stateless Catalog (CS/W) and the Gazetteer Specification. Each derived service modified the WFS interface and the payload that the interface operated upon slightly to satisfy its requirements. After the initial work on the Stateless Catalog, it was realized that it would be beneficial if a standard set of base types could be defined from which data management service interfaces could be derived by restriction or extension or both. The WOS discussion paper represents the first attempt to define such base types, and to define an interface for managing different types of objects including styles, symbols and images.

The WOS discussion paper can be found at http://www.opengis.org/docs/03-013.pdf.

3.5 Image Handling for Decision Support (IH4DS)

3.5.1 IH4DS Scope

The growing availability of geographic imagery opens opportunities for applications in public and private sectors. At the same time, there are ever-increasing volumes of complex geographic imagery data being collected by new sensor systems. Reducing the gap between raw remote sensing data and the information needs of applications users is key to the broader use and more effective exploitation of geographic imagery.

For purpose of this theme, coverages are geographic features and an image is a type of gridded coverage, with the coverage function relating locations on the ground to their representation on in an image surface. The images are collected by variety of sensors (e.g., photogrammetric, multi-spectral, hyper-spectral, video, radar) and analyzed by a variety of processing software algorithms and systems. A gridded Digital Elevation Model (DEM) may be considered an "image."

Decision support systems are interactive computer-based systems designed to help people and organizations retrieve, summarize, analyze data and conduct predictive analysis on scenarios that enable enhanced capacity to make better decisions.

This Image Handling for Decision Support (IH4DS) theme of OWS2 aims to use and, where appropriate, define and enhance interoperability specifications to increase the availability and usability of geographic imagery, enabling the use of information derived from imagery to be used with all other forms of geographic information by decision support tools and systems to derive information that enhances the human capacity to make decisions.

3.5.1.1 Problems and Objectives

Table 4 – IH4DS Problem & Solution Summary

Problem Statement	Description of solution and solution inputs, outputs, constraints, variables and alternatives
1. Very large data volumes	• Leverage Web Services technologies for constructing highly distributed, scalable image archive and handling capabilities
2. Automated processing is required	• Web Services for greater distribution and specialization of image handling services
3. Proprietary/closed systems, networks and formats inhibit broader accessibility	• Web Services for standards-based access to self-describing services and data.
4. Proprietary/closed systems, networks and formats inhibit broader accessibility	• Web Services for standards-based access to self-describing services and data.
5. Inability to flexibly automate information discovery, access, analysis, visualization and modeling activities; these are often human-intensive activities that take time, are error-prone and require	• Web Services enabling greater distribution, specialization and automation of services supporting data discovery, access, management and analysis activities for assisting decision-makers.

Problem Statement	Description of solution and solution inputs, outputs, constraints, variables and alternatives
specialized skills to accomplish	decision-makers.
6. Inability to accurately model and predict complex multivariate and multi-dimensional phenomena.	• Standards-based Web Services for performing specialized predictive modeling
7. Lack of information sharing for collaborative multi-participant decision-making activities.	• Standards-based Web Services and service infrastructures for enabling multi-participant collaboration.

Table 4- IH4DS Objectives

IH4DS Objectives
1. Speed availability of information through standard data formats, open and available protocols, and standard validation and verification information
2. Enable systems that integrate/merge applications for multi-disciplinary solutions
3. Span gaps between the raw remote sensing data and the information needs of applications users, e.g., decision support systems
4. Describe coverages in support of automated image handling
5. Develop image processing services as loosely-coupled Web services, e.g., data mining on imagery, enabled by service chaining.
6. Speed availability of relevant information with standard data formats, open and available protocols, and standard validation and verification information
7. Enable systems that integrate/merge applications and multi-disciplinary solutions
8. Develop standard mechanisms and services for automated generation of decision aids for decision-makers
9. Develop standard mechanisms for enabling and automating information sharing and collaboration among multiple parties.

The overarching objective for accomplishing these is to further mature the OGC Technical Baseline (TB) specifications, particularly those directly related to this theme.

3.5.1.2 Requirements

Proposals for the IH4DS theme should specifically address, in whole or in part, the following requirements:

1. A workstation-based image handling application must access image archives using the WCS 1.0 specification. Such an application client must, at a minimum, be able to:

- Demonstrate access to two or more WCS 1.0 implementations
- Obtain image data from WCS 1.0 implementations available for NASA Data Pool (EOSDIS) and Spot Image.
- Get a list of image data products using the WCS interface
- Specify a list of product types (e.g., ASTER, MODIS, aerial photo, DEM, etc) to access using the WCS interface
- Subset an image by geographic area of interest using the WCS interface
- Subset an image by time instants or intervals, radiance band or other properties of image data products using the WCS interface.
- Perform portrayal and/or local modification of image data obtained from WCS.

2. Support for XML-based image metadata content specifications

2.1. WCS client applications must demonstrate support for image metadata specifications for describing images accessible via WCS. All of these XML-based metadata schema should be supported:

2.1.1. ISO TC22 19139 extended in this initiative with FGDC Metadata Extensions for Remote Sensing

2.1.2. ESML (version 3, June 4 2003) (http://esml.itsc.uah.edu/index.jsp)

2.1.3. SensorML (OGC Discussion Paper, Project Document 02-0264r4)

2.2. WCS implementations must support (in GetCapabilities and DescribeCoverage responses) one or all of these metadata schema:

2.2.1. ISO TC22 19139 extended in this initiative with FGDC Metadata Extensions for Remote Sensing

2.2.2. ESML (version 3, June 4 2003)

2.2.3. SensorML (OGC Discussion Paper, Project Document 02-0264r4)

3. Define and implement new Image Manipulation Services:

3.1. Image Classification Service (ICS) as informed by OGC Project Document 02-012. Note: implementations of such a service shall be built on existing and proven image classification tools that can be wrapped with OWS-compatible interfaces and made accessible on the Web.

3.2. Web Coordinate Transformation Service (WCTS) for performing image coordinate transformations. Note: this service should be based OGC Discussion Paper 02-061r1 (or suitable update) and support emerging Coordinate Reference System models defined in OGC Project Documents 03-009 (Abstract Specification change proposal “Revised UML model for Abstract Specification Topic 2: Spatial referencing by coordinates”) and 03-010 (“Recommended XML encoding of coordinate reference system definitions”).

3.3. Image Manipulation services must be able to participate in image processing service chains (see requirement below).

4. Support for service chaining middleware for image processing

   4.1. Enable an image processing workflow (i.e., a chain of two or more services for manipulating an image) to be defined using BPEL.

   4.2. Enable client applications to define and invoke an image processing workflow encoded with BPEL and executed using Collaxa BPEL Orchestration Services (http://www.collaxa.com) or equivalent software.

   4.3. Enable an image processing workflow involving a client application, one or more WCS instances and one or more image manipulation service instances.

5. All service implementations used in this thread must support definition of service operations using WSDL and provide mechanisms for service consumers to obtain WSDL documents. Client applications must be able to use WSDL as required to support the IH4DS requirements.

3.5.1.3 Deliverables

Proposals for the IH4DS theme should specifically address, in whole or in part, the ability to lead or contribute to the following deliverables:

1. An Image Handling client application that successfully:
   - interoperates with more than one WCS implementation to access image data
   - invokes a service chain for remote execution and accesses the resulting image data
2. Change proposals for corrections and enhancements to the WCS 1.0 specification as tested and demonstrated in this initiative.
3. Change proposals for ISO 19139 and/or ESML metadata content specifications (for consideration by their respective administering organizations) as tested and demonstrated in this initiative.
4. Interoperability Program Report (IPR) for implementation specification of Image Classification Service.
5. Interoperability Program Report (IPR) for implementation specification of Web Coordinate Transformation Service.
6. Interoperability Program Report (IPR) describing the use of BPEL for definition and invocation of Web Services for image processing as defined, tested and demonstrated in this initiative.

3.5.2 IH4DS Enterprise Viewpoint

The IH4DS Enterprise Viewpoint captures the capabilities that must be present in support of Image Handling for Decision Support activities and operations. The capabilities identified here describe the requirements to be met by the OWS computation and information models. The Enterprise Viewpoint is defined by a high-level system concept and use-cases. The system concept illustrates the operational setting, major system components and key interfaces. The use cases provide descriptions of the behavior of the system from the point of view of enterprise users.

A distinguishing feature of geographic imagery, versus other geographic features, is the data volume. OWS Image Handling requirements are based on current large imagery enterprises, with heavy usage and frequent additions to repositories of images. For example, $\sim 10^7$ images might be archived, with $\sim 10^4$ additional images per day, $\sim 10^6$ image retrievals per day, and $\sim 10^6$ image queries per day. In addition, there are these challenges to consider:

- National imagery archives are of petabyte size; ingesting a terabyte per day
- Individual application data centers archive ~100 terabytes of imagery
- Tens of thousands of datasets have been catalogued but are not yet on-line.

Most geographic imagery will never be directly accessed by humans. Human attention is the scarce resource as there are insufficient human resources to view petabytes of data. Information technology allows the creation of geographic information products through processing of geographic imagery. Imagery archives must be accessed by software to extract information of value to a community for a specific business activity.

The ultimate challenge (and opportunity) is to enable the geographic imagery collected from different sources to become an integrated digital representation of the Earth that is widely accessible, through open and interoperable technologies, for humanity's critical decisions.

3.5.2.1 System Concepts

Figure 4 depicts the basic elements of the IH4DS concept. The figure represents a conceptual deployment (i.e., one of many possible mappings) of software components to run-time processing nodes and their dependencies.

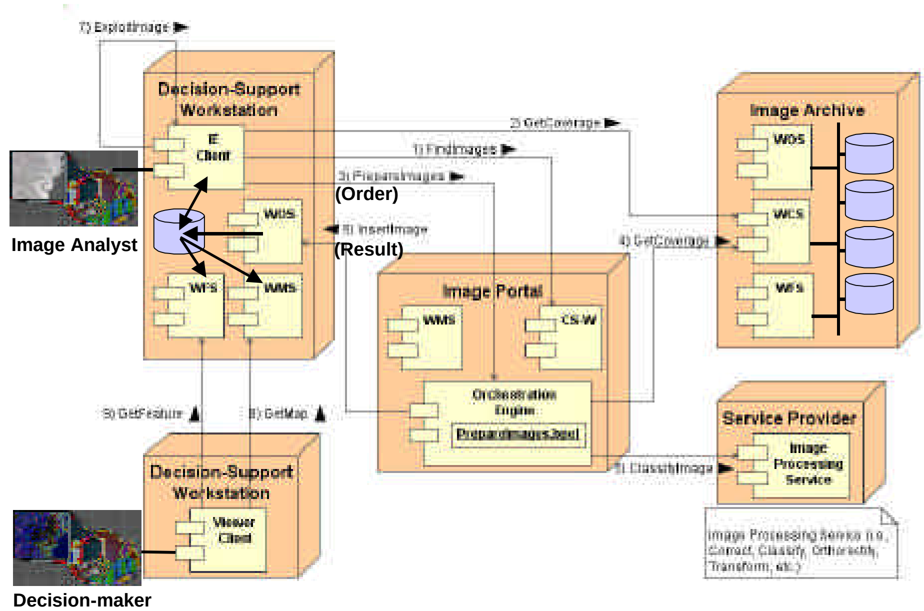


Figure 4. Concept Image Handling Services for Decision Support

The nodes and components in Figure 4Figure are described briefly here:

- Data sources, primarily for imagery (raw and processed) and metadata are provisioned as Web-accessible Image Archive Services (IAS) of an Image Archive node. The Image Archive repositories are populated by external systems (e.g., sensors, communications links, ground stations, planning and collection systems, etc.), the architecture details of which are out of scope in the IH4DS context. In the OWS-2 architecture, the data sources at the Image Archive are published and accessible via implementations of OpenGIS® Web Service (OWS) interoperability specifications such as GML, SensorML, CS-W, WFS, WCS, WOS, WMS described in this and other supporting documents.

- An Image Portal provides the ability for users (and client applications via Web Service interfaces) to discover, view, access and process images from the Image Archive. A BPEL Orchestration Engine component is also integrated with the Image Portal to enable specialized processing of data from the Image Archive to be defined, invoked and controlled by the portal on behalf of portal users and client applications.

- An Image Exploitation Client ("IE Client" in the diagram) application on a Decision-Support Workstation node may act as a client to the Image Portal, issuing requests to search for images of interest held within the Image Archive (Step 1) and subsequently access the Image Archive to access images directly (Step 2). In addition, the IE Client may "order" specialized preparation of images from the Image Archive to be performed (Steps 3-5) and direct the results to be saved to a local repository (Step 6). The Image Analyst using the IE Client may perform additional processing of imagery including, for example, feature extraction, saving the results of the analysis (including newly extracted features and images) to the local repository (Step 7).

- A Viewer Client application on a Decision-Support Workstation node may be used by Decision-makers to access the results of the newly developed analysis, visualize, report and further analyze the data (Steps 8 and 9).

The physical distribution of components and nodes in the diagram are notional. There is more than one way to distribute these elements of the architecture and control their interactions. Of particular importance is the ability to flexibly and transparently distribute these services across wide-area (e.g., the Internet) and local-area networks. Except perhaps for issues of network security and throughput, all nodes and their components could be distributed for access over the Web just as transparently as they could on a single private LAN.

Note about diagram: Deployment diagrams show the physical layout of the various nodes (typically, networked computers and other hardware devices) that compose a system as well as the mapping of components (e.g., applications, services and repositories) to nodes. A node is a physical object that represents a processing resource, generally having at least a memory and often processing capability as well. Components are run time objects that may reside on node instances. A component represents all kinds of elements that pertain to piecing together systems. They can be simple files, documents, programs, applications, databases, Web services, etc. Nodes and their components are shown with the connections they have to each other, thus specifying system topology.

3.5.2.2 Use Cases

The following are representative examples where Earth Science Enterprise (ESE) data and scientific processes have been and/or are being used to demonstrate solutions for decision support needs of other Federal Agencies, states, local communities, and/or industry applications.[4] They are included here to provide further context for the system concept described above and the architectural elements described in greater detail in the sections that follow.

3.5.2.2.1 Wild Fire Management

This scenario illustrates how geospatial data from a wide array of sources could be integrated with powerful computational models, enabling us to predict more accurately the onset and behavior of wildfires. The size and severity of wildfires depend upon how quickly and effectively firefighting resources are deployed and how successfully the areas of high risk can be evacuated. In this hypothetical future, a wildfire hazard system is in constant operation: (Excerpted from NAS Study: IT Roadmap to Geospatial Future, 2003.)

Typical of this application would be the use of information products and computing services to enable U.S. Forest Service (USFS) and firefighters with awareness of the trend analyses of wildfire extent, expansion, and direction. To accomplish this, National Aerospace Science Administration (NASA) MODIS terrestrial and cloud mapping/monitoring data sources would be used to perform analytical functions such as image calibration, processing, and extraction of smoke signatures in order to determine the location and extent of smoke and fire.

Decision-making activities requiring support of information products and services include:

- Monitoring

The wildfire hazard system automatically monitors the national landscape to ensure early detection of fire outbreaks. Although dry fuel load (biomass with low water content) is the most direct indictor of potential fire severity, it is too difficult to measure over large areas, because remote optical instruments respond to the radiation reflected from the leaves rather than the dry fuel. Because ground-based sensors are

---

[4] From: EARTH SCIENCE ENTERPRISE APPLICATIONS STRATEGY *for* 2002-2012, National Aeronautics and Space Administration, January 2002.

impractical over vast areas, the new system monitors data (e.g., lighting strikes, Doppler weather radar, soil surface properties and wind data) harvested from satellites. A wide array of satellites – some of them engaged in classified or proprietary reconnaissance – has been deployed in recent years, making it possible to acquire data updates at coarse spatial resolution almost continuously, with higher-resolution (~1 km) data available at intervals of several hours. The wildfire hazard system warns of the possibility of fires by combining these measures with spatially distributed models of plant growth and drying (as functions of energy and water inputs, which vary at the synoptic scale as well as locally with elevation and slope orientation) and with spatio-temporal data about historical wildfire occurrences. Once a fire starts, satellites sensing radiation in the infrared portion of the spectrum can detect small hot areas, as long as their view is not obscured by clouds. Not all of these hot targets are fires, however, so to avoid false alarms, the hazard system must integrate, mine, analyze, and cross-compare data to reliably identify wildfire outbreaks.

- Detection and Evaluation

When an apparent wildfire is detected, a standby alert is issued to emergency response authorities. The measurements from the remote sensing instruments are passed to a system component that calculates the geographic boundaries of the fire itself and of the area affected by smoke. The system automatically identifies potentially relevant data sets, and it harvests data on vegetation/biology, wildfire-spread factors (vegetation flammability, location of natural and man-made fire barriers, etc.), and meteorological conditions. Weather prediction and chemical plume diffusion models are activated to forecast how the fire and smoke /debris will spread. A wildfire is especially complicated because its behavior depends on the three-dimensional flow of air over terrain, which in turn depends on both synoptic weather conditions and the convections that the fire itself causes. The hazard system combines models of the airflow with the Doppler wind profilers to estimate the state of the overlying atmosphere. As the wildfire spreads, the hazard system rapidly updates the models to predict the future behavior of the fire.

- Response

An emergency response component is activated to cross-analyze the simulation results with data on the locations of population centers, remote dwellings or businesses, and evacuation routes. Results are presented to a distributed control team that reviews the data, evaluates the risks and collaboratively selects a plan of action. Public agencies are alerted to begin the evacuation process, with detailed routing information provided automatically to all cell phones, pagers, PDAs, and other location aware devices in the affected area. Meanwhile a fire control component is activated. This cross-analyzes the original simulation results – the wildfire-spread prediction model continues to run, using constantly updated sensing data – with data on access paths for firefighting equipment and personnel. The component proposes strategies for combating the fire and predicts the relative effectiveness of each strategy in containing damage to natural resources and property. As firefighting crews are dispatched, they are provided with strategic scenarios and routing information. Real-time updates flowing through the system make it possible to adjust strategies and routing as conditions change.

In this scenario, a number of new challenges arise because predictive models have been coupled with the time-critical analysis of extremely large amounts of data:

- Development of systems that can harvest classified and proprietary data, with appropriate barriers to unlawful access;
- Methods for integrating computational, observed, and historical data in real time;
- Methods for dynamically coupling independent numerical models and infusing external data into them to develop, evaluate, and continuously refine strategies for emergency response;
- Algorithms capable of tracking moving and evolving objects and predicting their future state;

- Methods for automatically identifying and communicating with persons in the affected area via wired and wireless communication mechanisms (household and cellular telephone numbers, pagers, PDAs, satellite TV and radio, cable TV, and the Internet) based on geographic location; and

- User interfaces empowering a range of users (from emergency responders to local government officials) with little or no training to collaboratively evaluate proposed plans and coordinate actions.

3.5.2.2.2 Coastal Beach Mapping for Sustainable Management

The US has approximately 95,000 miles of coastline. The coastal zone provides resources for human and natural populations, unique ecosystems, and a range of economic activities. Currently, more than half the US human population lives in coastal counties. The carrying capacity of the coastal zone, effects from human impacts, alternations of food webs, and the mitigation of natural and anthropogenic hazards are of vital importance to the nation. Several interagency programs are developing operational observing systems to identify and monitor changes and potential threats within the coastal zone. Earth science-based solutions to this application involve the use of NASA's measurements, scientific knowledge, data assimilation techniques, and modeling into public and private decision tools to support coastal management, especially issues concerning harmful algal blooms, hypoxia, and coastal inundation. Measurements and information products include temperature, salinity, phytoplankton, hydrology, shoreline changes, bathymetry, and soil moisture. Assessments and predictive capability are needed to predict onset of events that may significantly effect human health, critical wetlands and ecosystems, and economic development

Typical of this application would be the use of information products and computing services to enable NOAA and coastal community managers knowing the precise terrain for beach management. To accomplish this, NASA airborne Wallops Test Facility lidar data sources are used to perform analytical functions such as derivation of elevation data and change detection in order to determine the location and extent of beach morphology in coastal areas.

3.5.2.2.3 Agriculture Crop Greenness and Production Assessment

A primary factor impacting production and yield on a given field is weather. Weather is a regional phenomenon that can be predicted from global indicators (i.e., El Nino). Improvements in agricultural competitiveness require better understanding of weather and climate, especially prediction of events with increasing accuracy and longer lead times. Solutions serving this application will draw directly from the results of NASA research and development of Earth science and technology that have potential to address weather and climate predictions and observations that can be integrated into local and regional decision support systems used in agriculture management. This national application draws upon, and contributes to, other information solutions associated with early warning for homeland security, water management and conservation, air quality management, carbon management, and invasive species management.

Typical of this application would be the use of information products and computing services to enable U.S. Department of Agriculture (USDA) and over 55,000 subscribers knowing the information provided in the Green Report® and RangeReport® products delivered by the Great Plains. To accomplish this, NASA MODIS, AVHRR, and LandSat data sources are used to perform analytical functions such as calibration, processing, and extraction of crop signatures in order to determine crop greenness and production.

3.5.2.2.4 Hurricane Track Prediction

Community preparedness for disaster management involves assessments of vulnerability, risk, and response to short-lived phenomena in the Earth's atmosphere, land and oceans. Particular episodic events are of concern such as severe weather (thunderstorms, tornadoes, and hurricanes), as well as tsunamis, river flooding, plain/coastal flooding, volcanic ash, earthquakes, harmful ocean blooms and human-made disasters such as petroleum releases in rivers and oceans. Communities need an increased understanding of the effects of short-term events on the physical, chemical, and biological processes that interact to affect human safety, the environment, and the economy. Improved decision support systems are being developed that may address human life and property damage, meet the requirements of planners, early warning systems, first responders, and contribute to impact assessments, risk communication, mitigation, and implementation of relief efforts. Disaster management applications will evolve in cooperation with federal agencies such as the Federal Emergency Management Agency (FEMA). The applications will draw upon, and impact other national applications including coastal management, community growth, homeland security, public health and water management.

Typical of this application would be the use of information products and computing services to assist NOAA National Weather Service daily delivery of weather products. To accomplish this, NASA QuickScat data sources are used to perform analytical functions such as calibration, processing, verification and validation in order to determine 24-hour forecasts of global wind profiles.

3.5.2.2.5 Homeland Security

Federal, State, and local governments are cooperating to prevent and reduce America's vulnerability to terrorism, minimize possible damage, and recover from attacks that do occur. Agencies are strengthening aviation and border security, preparing the defense against bioterrorism, improving information sharing, and deploying more resources to protect our critical infrastructure. NASA's measurements, observations, and modeling can provide data and information to Homeland Security networks to support risk assessments, vulnerability assessments, and mitigation assessments. Data and information can support decision making to ensure the adequacy of preparing for, preventing against, responding to and recovering from terrorist threats or attacks. Earth science-based solutions can serve this application by drawing on developments in several other applications, such as air quality, water management, public health, and disaster management. This application will focus especially on providing NASA data, information, and models to support governmental decision tools that identify, track, and forecast agents from anthropogenic disasters and terrorism introduced into the air and water. Prediction of events, hazardous situations, and impacts with increasing accuracy and longer lead times is a significant part of this application.

3.5.3 IH4DS Information Viewpoint

The information viewpoint is concerned with the information processing semantics, information system constructs (information model, semantics, data models, schemas), concepts, rules and structures of the architecture independent of distribution and implementation details.

3.5.3.1 Sensor Instrument Description

- SensorML - provides an XML schema for defining the geometric, dynamic, and observational characteristics of a sensor. Sensors are devices for the measurement of physical quantities. There are a great variety of sensor types from simple visual thermometers to complex electron microscopes and earth observing satellites. The purpose of SensorML is to provide: provide general sensor information in support of data discovery, support the processing and analysis of the sensor measurements, support the geolocation of the measured data, provide performance characteristics (e.g. accuracy, threshold, etc.), archive fundamental properties and assumptions regarding sensor.

  References:

  - **Sensor Model Language (SensorML) Discussion Paper**, OpenGIS® Project Document 02-026r4. Available at: http://www.opengis.org/specs/?page=baseline

3.5.3.2 Image Representation

Representations (i.e., encodings, formats) of image data to be supported for transport between producer and consumer service components include the following:

- HDF-EOS (http://hdfeos.gsfc.nasa.gov/hdfeos/index.cfm)
- GeoTiff (http://remotesensing.org/geotiff/geotiff.html)
- Dimap (http://www.spotimage.com/dimap.html)

Other representations may be supported but those listed here are the priorities for the IH4DS architecture.

3.5.3.3 Image Description

To support processing of coverages by client applications and services, a detailed description of the contents of a Coverage must be available. Coverages can be extremely complex in their native organization and their packaging for interoperable transport and subsequent processing can introduce more variety. The following content description schema describe coverage data and are distinguished primarily by their utility in describing coverage data at different levels of abstraction and for different purposes.

- SensorML – Refer to Section 3.5.3.1 for general description. A description of specific sensor properties (e.g., camera models) that SensorML affords is particularly essential when computationally rigorous image processing operations are applied to coverage data.

- ESML – A technology that is relevant to describing coverages is Earth Science Markup Language (ESML). ESML is an XML-based markup language used to describe the structure, semantics and content of any earth science dataset in any data format. The ESML contains two parts: 1) ESML files written using a number of XML-style elements and attributes to describe Earth Science data sets, and 2) ESML schema which defines the elements and attributes used in ESML files. These elements and attributes are used to describe various Earth Science data sets. The ESML schema is also used to validate an ESML file. To facilitate ESML applications, an ESML library is designed and implemented to help data reading using ESML files. The ESML files conformed to the ESML schema are the valid ESML files and are supported by the ESML library.

  References:

  - http://esml.itsc.uah.edu/index.jsp

- 19105/19139/FGDC-RSE – these XML schema support coverage descriptions for more general purposes of publishing, cataloging and discovery:

  - FGDC-STD-001-1998. Content standard for digital geospatial metadata (revised June 1998). Federal Geographic Data Committee.
  - ISO FDIS 19115 Geographic Information – Metadata
  - ISO Project 19139 (XML) Implementation of 19115
  - FGDC Metadata Extensions for Remote Sensing (this is the subject of a future ISO TC211 Project)
  - OWS1.2 IPR Imagery Metadata, OGC Project Document 03-017

  The FGDC's Content Standard for Digital Geospatial Metadata defines the content (but not the encoding or presentation) of metadata describing geospatial data. This was the starting point for the ISO 19115 Metadata standard that provides a UML model of metadata, based on the FGDC's Content Standard. Its chief purpose is to support profiles, using a small set of required elements and many optional ones.

The FGDC Content Standard for Digital Geospatial Metadata: Extensions for Remote Sensing Metadata (FGDC–RSE) provides extensions to version 2 of the FGDC standard. Production rules are used to express the relation between individual elements and the compound elements of which they are a part. The new elements in the *Remote Sensing Extensions* are organised under the production rules structure of the base FGDC standard. The data dictionary for the *Extensions* contains, in addition to the information in the data dictionary of the base standard, a description of parent and any child elements, the obligation and the repeatability (both of which can also be inferred from the production rules) and the source of the definition.

The ISO 19139 project intends to amend the 19115 standard with normative XML schema for ISO 19115 compliant metadata. The schema will not be part of 19115 but will be a technical amendment to it.

References:

- Content Standard for Digital Geospatial Metadata: http://www.fgdc.gov/metadata/contstan.html
- FGDC Extensions for Remote Sensing Metadata: http://www.fgdc.gov/standards/documents/standards/remote_sensing/Extensions_PublicReviewDraft.pdf
- ISO 19139 version 0.6, http://metadata.dgiwg.org/metadata/public/iso19115/impl/model/webver/index.html

3.5.3.4 Service Description

To support discovery and access to services, a detailed description of the type of service, its capabilities, and content must be supported. IH4DS seeks implementations supporting the ability to publish and respond (when requested) with appropriate descriptive information. The following service description schema may be used to accomplish this.

- Service Information Model (SIM)

OpenGIS® Web Services are designed to provide geographic information over the Internet. As such, they share many capabilities and characteristics of more generally defined Web Services, but also have features unique to geospatial needs. In particular, OGC Web Services are, by definition, self-describing and support the ability for consumers to discover the valid operations and content of service types and instances. SIM defines the semantics and structure for packaging metadata about services necessary for a client to make use of (i.e., "consume") a service. SIM is an XML vocabulary comprised of several parts for describing different aspects of a service. The first unit describes the service interface in sufficient detail so that an automated process can read the description and invoke an operation that the service advertises (using WSDL). A second optional unit describes the data content of the service, or the data it operates on, in a way (using metadata content standards) that enables service requestors to dynamically compose requests for service. Additional description units provide information specific to particular types of services as well as specific instances of services. The SIM specification defines a complete service information model, expressed in UML, XML Schema, and as XML sample instances.

References:

Service Information Model Discussion Paper, OpenGIS® Project Document 03-026. Available at: http://www.opengis.org/specs/?page=baseline

- WSDL

Web Services Description Language (WSDL) is a specification from W3C to describe networked services. WSDL is used to describe *what* a web service can do, *where* it resides, and *how* to invoke it. It provides a simple way for service providers to describe the basic format of requests to their systems. WSDL is a key complement to the OpenGIS SIM as it provides descriptive information about services that is key for publishing descriptions of on-line services to registries and subsequent discovery and access of services by client applications and services. WSDL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define an endpoint. Related concrete endpoints are combined into abstract endpoints (services). The ability for Web-accessible services to be described, at least in part, by WSDL is a key requirement for defining and then executing "orchestrated" interactions of services specified using encoding languages such as BPEL and executed using service orchestration engines.

References:

Web Services Description Language (WSDL) Version 1.2. W3C Working Draft (9 July 2002). World Wide Web Consortium (W3C). Available [online]: http://www.w3.org/TR/wsdl12/

3.5.3.5 Process Management and Control Description

The IH4DS architecture supports Business Process Management (BPM) technologies to facilitate adaptive enterprise functionality. It is a goal is to maximize the use of BPM infrastructure and limit adding general business process functionality to OGC Web Services. For IH4DS, the focus is on producing the key OGC interoperability components needed to support interoperation of image discover, image access and processing in support of workflows for decision-making activities. Another goal is to be able to change the business process without impacting the interfaces of the OGC Web Services (i.e., having to support multiple versions of service interfaces). It is expected that additional specifications for Web Services will emerge as new use cases are considered, where these new Web Services will "fill in" required components for the BPM layer to form viable service chains that are needed to achieve desired results.

- BPEL4WS

The Business Process Execution Language for Web Services (BPEL4WS or BPEL for short) defines a notation for specifying business process behavior based on Web Services. It is a standard promoted by Microsoft, IBM, Siebel, SAP and BEA for orchestrating discrete services into end-to-end business processes. Processes defined in BPEL can export and import functionality by using Web Service interfaces exclusively. BPEL provides a language for the formal specification of business processes and business interaction protocols. By doing so, it extends the Web services interaction model and enables it to support business transactions. BPEL defines an interoperable integration model that should facilitate the expansion of automated process integration in both the intra-corporate and the business-to-business spaces.

Business processes can be described in two ways. Executable business processes model actual behavior of a participant in a business interaction. Business protocols, in contrast, use process descriptions that specify the mutually visible message exchange behavior of each of the parties involved in the protocol, without revealing their internal behavior. The process descriptions for business protocols are called abstract processes.

Languages for encoding models of both executable and abstract processes define abstract processes.

References:

- **BPEL4WS 1.1 Specification**:
- **OASIS Web Services Business Process Execution Language Technical Committee**. (http://www.oasis-open.org/committees/wsbpel)

3.5.4 IH4DS Computation Viewpoint

The Computation Viewpoint is concerned with the functional decomposition of the system into a set of objects that interact at interfaces– enabling system distribution. It describes an interaction framework including application objects, service support objects and infrastructure objects. It focuses on identifying and describing objects that interact at interfaces and the constraints on their actions and interactions. It captures component and interface details without regard to their distribution.

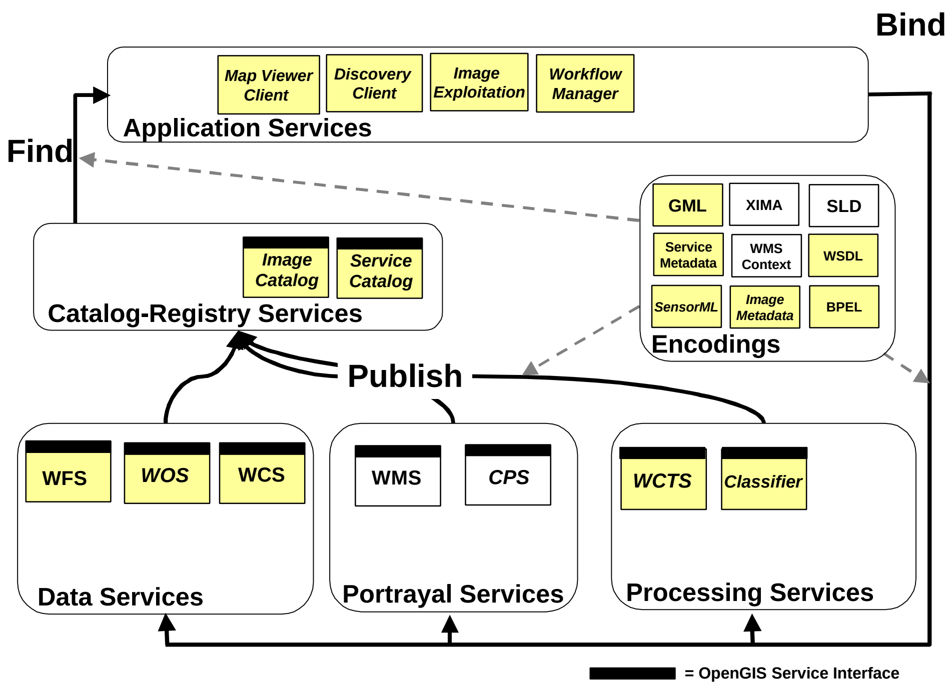


Figure 5. IH4DS Services

Figure 5 identifies the IH4DS Framework, a profile of the OpenGIS Service Framework, for OWS2. All components shaded (yellow) in the above figure are key to the IH4DS Framework. Italicized components are anticipated to require enhancement based on OWS2 requirements and previous work to enhance the specifications on which they are based. Elements of this computational framework are described in greater detail below.

3.5.4.1 Application Services

- Map Viewer Client

A Web Map Viewer Client, including server-based clients, can issue GetMap requests for different maps to several independent Web Map Servers. If each map has the same geographic area and physical dimensions, and if their backgrounds are transparent, then they can be overlaid in a single window to produce a combined map. For example, server A might produce a topography image, server B a map of rivers and lakes, and server C a map of watershed boundaries. Each server maintains the type of data in which it specializes, but the end user can obtain a combined presentation of the three Layers. The Web Map Viewer Client may itself perform the portrayal process, acting as a tightly coupled Portrayal Service, or it may delegate to a loosely coupled Portrayal service, such as a WMS, to produce a map.

For OWS2, Web Map Viewer Clients must support the ability to use the IH4DS components when requested by a user to generate cartographically rendered pictures. The Web Map Viewer Client is a kind of Decision-Support tool for performing portrayal and visual analysis of imagery and geospatial information.

- Discovery Client

A Discovery Client provides the tools to enable the user to build services or data queries, issues those queries to the appropriate registry(s) and display the returned result set.

- Image Exploitation Client

Figure 4 shows the Image Exploitation Client (IE Client) as a component deployed at a Decision-Support Workstation node. The IE Client is a "thick" (i.e., robust, full-featured) client application or suite of applications. Examples of IE Client applications include commercial-off-the-shelf systems from organizations such as PCI-Geomatics, Intergraph, Polexis, ERDAS, Boeing/Autometric (EDGE), BAE (Socet Set), Analytical Graphics (STK) and government-developed or open-source systems such as CATS, CHARTS, MATLAB, etc.

For IH4DS, the IE Client should be fully interoperable with components deployed at Image Archive Service and Image Portal nodes that implement OpenGIS Service interfaces (such as WCS, WOS, WFS and CS-W) and Orchestration Engine components described elsewhere in this document.

A goal of an IH4DS framework is to enable collaborative geospatial decision-making and group work with geographic information. The current goal of state of the art implementations of collaborative geospatial decision-making is to have display-enabled decision-making equal to that of paper map-enabled decision-making. The Image Exploitation Client should be extensible to support enahanced decision-support capabilities such as:

- Distributed portrayal-- multiple clients at distributed locations with a shared composite view coming from multiple servers.

- Collaboration-- Extensions should support simultaneous users, each having input to the shared workspace. A future goal is to enable what-if scenarios, e.g., group control of geo-process simulations.

- Partial views-- Supporting users with different display technologies and computing platforms, e.g., mobile, will have different views of the same geospatial data.

- Workflow Manager

The Workflow Manager executes workflow processes and correlates and coordinates synchronous interactions into collaborative and transactional business flows. It is an infrastructure service for modeling, connecting, deploying and managing and executing business processes. The Workflow Manager:

  - Allows composite web services to be defined
  - Integrates asynchronous services
  - Coordinates multi-step business processes

- Publishes business processes as Web Services

For OWS2, the Workflow Manager must support BPEL4WS as the encoding language for specification of workflow processes.

3.5.4.2 Image Archive Services

- WCS

The Web Coverage Service (WCS) supports the networked interchange of geospatial data as "coverages" containing values or properties of geographic locations. Unlike the Web Map Service, which filters and portrays spatial data to return static maps (i.e., server-rendered pictures), the Web Coverage Service provides access to raw (unrendered) geospatial information and multi-valued coverages (such as multi-spectral images and terrain models), typically for input into scientific models and other client applications including simple viewers.

For IH4DS, the WCS interface is part of a set of service interfaces comprising the Image Archive Service. WCS instances must be capable of providing operational metadata associated with image datasets (refer to Sections 3.5.3.1and 3.5.3.3 above) as well as specific image data encodings (refer to Section3.5.3.2).

- WFS

The Web Feature Service (WFS) supports the query and discovery of geographic features. In a typical Web-based scenario, Web Feature Service delivers GML representations of simple geospatial features in response to queries from HTTP clients. Clients (service requestors) access geographic feature data through a WFS by submitting a request for just those features that are needed for an application. The client generates a request and posts it to a WFS instance (a WFS server on the Web). The WFS instance executes the request, returning the results to the client (service requester) as GML. A GML-enabled client can manipulate or operate on the returned features.

For IH4DS, the WFS interface is part of a set of service interfaces comprising the Image Archive Service. WFS instances may also be used to serve geospatial data for analysis and decision-support activies.

- WOS

The Web Object Service (WOS) defines a standard interface to an object repository. It defines a set of generic XML types from which object access and management services, such as WFS and WRS, may be derived. It also describes an unspecialized instantiation of the types defined with XML Schema to define a Web Object Service. Like the WFS and WRS, the WOS supports INSERT, UPDATE, DELETE, QUERY and DISCOVERY operations on object instances other than (but not excluding) GML features. Object instances may be encoded directly into a WOS request message, using XML, or they may be referenced using other mechanisms described in this document.

For IH4DS, the WOS service will be used to enable Web-based publication of and access to image data and image metadata held within repositories. The WOS interface is also part of a set of service interfaces comprising the Image Archive Service.

- CS-W

Catalog-Registry services implemented with the emerging “Catalog Service-Web profile” (CS-W) interface specification (formerly known as Web Registry Service (WRS)) provide a common mechanism to classify, register, describe, search, maintain and access information about network resources. Resources are network addressable instances of typed data or services. Registries may be differentiated by their role such as registries for cataloging data types (e.g., types of geographic features, coverages, sensors, symbols), online data instances (e.g., features, images, measurements), service types (e.g., portrayal, processing, data services) and online service instances.

The metadata content published to the registry, while conforming to the same Registry Information Model (RIM), describes different kinds of resources using metadata that may be structurally and semantically different than metadata for resources of other types or for other purposes or organizations. The OpenGIS Web Registry Service defines a common information model and the service interfaces to access resource offers, regardless of the type of resource and the content of the metadata.

For IH4DS, CS-W service instances must support the ability to publish image metadata and discover image data from the published image metadata. The ability to publish and share image metadata is an essential requirement for distributed applications to share and exploit these resources.

3.5.4.3 Image Processing Services

An aim of OGC Web Services is to enable functionality in a distributed network environment similar to that currently available workstation based image processing. The OWS Service Architecture defines a multi-tier framework for services on geographic data. To date, the emphasis has been on discovery, access and portrayal services. The IH4DS extends the existing baseline of OWS service types with image processing services. For IH4DS, the following services are to be accessed as OWS-style Web Services:

- WCTS

A Web Coordinate Transformation Service (WCTS) transforms digital geospatial data from one Coordinate Reference System (CRS) to another. The geospatial data transformed is digital feature data, including digital coverages. Such transformations include all the types of coordinate operations, including both “transformations” and “conversions”. This service inputs digital features in one CRS and outputs the same features in a different CRS. The service inputs include identifications of the input and output CRSs, and optionally the coordinate transformation between these CRSs. A WCTS can be used by many different geospatial applications and other services. Transformation of geospatial data from one CRS to another is a frequent requirement when using data from different sources in one application. Not all applications or services are capable of directly performing such transformations.

- Classification Service

In Topic 15 of the OpenGIS Abstract Specification, an Image Modification Service is defined as one that that modifies version of images, providing access to all or selected sections of the modified image. Most of these services modify one image. However, some services must or can combine two or more images. Some of these service types also allow a client to change a selected section of the modified image, with the corresponding changes being made to the original image(s).

A Classification Service is a kind of Image Modification Service that performs supervised and/or unsupervised classification of images to identify or discriminate phenomena of interest based on their spectral signatures. The inputs and output of the service are Coverages.

3.5.5 IHS4DS Engineering Viewpoint

The Engineering Viewpoint is concerned with the infrastructure required to support system distribution. It focuses on the mechanisms and functions required to: a) support distributed interaction between objects in the system and b) hide the complexities of those interactions. It exposes the distributed nature of the system, describing the infrastructure, mechanisms and functions for object distribution, distribution transparency and constraints, bindings and interactions.

3.5.5.1 Orchestration Engine

The IH4DS architecture supports Business Process Management (BPM) technologies to facilitate adaptive enterprise functionality. The Orchestration Server executes workflow processes and correlates and coordinates synchronous interactions into collaborative and transactional business flows. It is an infrastructure service for modeling, connecting, deploying and managing and executing business processes.

For each process, the orchestration engine takes in a BPEL document that describes the process to be executed, a WSDL document (without binding information) that describes the interface that the process will present to clients (*partners* in BPEL terms), and WSDL documents that describe the service instances that the process may invoke during its execution. From this information, the process is made available as a Web Service. A WSDL file that describes the process's interface may be retrieved from the run-time.

It is a goal of IH4DS to maximize the use of BPM infrastructure and limit adding general business process functionality to OGC Web Services. For IH4DS, the focus is on producing the key OGC interoperability components needed to support discovery, accessing and processing of images in support of workflows for decision-making activities. Another goal is to be able to change the business process without impacting the interfaces of the OGC Web Services (i.e., having to support multiple versions of service interfaces). It is expected that additional specifications for Web Services will emerge as new use cases are considered, where these new Web Services will “fill in” required components for the BPM layer to form viable service chains that are needed to achieve desired results.

For IH4DS, it is the intent of Collaxa Inc. (http://www.collaxa.com) to provide their BPEL Orchestration Server software, a kind of Orchestration Engine component as described in Section 3.5.2.1, free of charge for use by participants during the OWS2 initiative.

3.5.5.2 The Open IH4DS Web Portal

The IH4DS Web Portal is envisioned as middleware for integrating information sources available at an Image Archive into a single mechanism for interacting with the user and applications over the Web. The IH4DS Web Portal facilitates providing these services in a secured manner. Multiple applications can share an IH4DS portal, and multiple portals can be linked to integrate even more information sources and applications. OWS interfaces and encodings are used to implement an open and extensible portal.

The IH4DS Web Portal, an Image Portal node as described in Section3.5.2.1, specifically provides the ability for users (and client applications via Web Service interfaces) to discover, view, access and process images from the Image Archive. An Orchestration Engine component, such as provided by Collaxa, is also integrated with the Image Portal to enable specialized processing of data from the Image Archive to be defined, invoked and controlled by the portal on behalf of portal users and client applications.

For IH4DS, it is desired that OWS2 participants define and make available an open “IH4DS Web Portal” for use by all project stakeholders.

3.5.5.3 Image Archives

The NASA Synergy Data Pools, a kind of Image Archive node as described in Section3.5.2.1, are planned to be accessible through a Web Coverage Service interface. This is a step toward making the Data Pools accessible as an OGC Image Archive Service, allowing client access to tens of terabytes of data in NASA Data Pools using OGC Web Services.

NASA's Earth Observing System Data and Information System (EOSDIS) is an archive and distribution service for multi-petabytes of earth observation data. This data is held in robotic tape archives. The typical access method is for a user to place an order, the data is written to media, and then delivered to the user. Recently, select EOSDIS data has been made available on-line for ftp access to data hosted on spinning disk storage. This new access method to EOSDIS data is known as Data Pools. Some relevant URLs are provided here:

Access to EOS Data Pools at the DAACs:

GSFC Earth Science Data Pool:	http://daac.gsfc.nasa.gov/data/datapool/
Data Pool for the Land Process DAAC at the Eros Da ta Center:	http://edcdaac.usgs.gov/tutorial/datapool.html
Langley DAAC data pool	http://eosweb.larc.nasa.gov/HPDOCS/datapool/
NSIDC data pool	http://nsidc.org/data/data_pool/

Equipping Data Pools with a WCS server will provide access to the data within the Data Pool in its native HDF-EOS format. Access to the native HDF-EOS (unrendered) geospatial information is needed for client-side rendering, multi-valued coverages, and input into scientific models and other clients beyond simple viewers. The GetCoverage operation has within it a variety of ways in which the coverage data can be manipulated. This manipulation or processing of the data before it is sent back to a WCS client can take one of many forms including, spatial subsetting, temporal subsetting, band/layer subsetting, reduced resolution, and data format (HDF, GeoTIFF).

EOSDIS Data is archived in the HDF-EOS format. The HDF-EOS Format Conversion Tool will reformat, re-project and perform operations such as subsetting and stitching on a selectable HDF-EOS object. The output file produced by the tool will be ingestible into commonly used GIS applications. The Tool also allows the user to select other output formats, binary and HDF-EOS Grid. The output data can be in other projections selected from the USGS General Coordinate Transformation Package (GCTP). The use of this tool to make EOS Data Pool data available through a WCS interface is under consideration.

For further information about the HDF-EOS Format Conversion Tool see:
http://hdfeos.gsfc.nasa.gov/hdfeos/details.cfm?swID=55

Sponsors and participants may offer additional implementations of Image Archive nodes for use during the OWS2 initiative.

3.5.6 Potential WCS-accessible Data Pool Products

The following list of NASA EOS Data Products are potentially available for access through the WCS-equipped Data Pools in OWS2. Specific products to be hosted will be chosen based on the needs in OWS2. RFQ responders are urged to identify specific data products that would be useful.

The Product IDs, e.g., AST_L1B.3 and MOD09A1.4, can be used to find detailed descriptions. Detailed information about the data products can be found via a "text search" using the product ID (ShortName) in the EOS Data Gateway. http://redhook.gsfc.nasa.gov/~imswww/pub/imswelcome/

Table 5. Available data from NASA EOS Data Pools

ShortName	Versions
MISR Products	
MI1B2E	3,4
MI1B2T	3,4
MIL2ASLS	3,4
MIL2ASAE	3,4
MIL2TCAL	3,4
ASTER Products	
AST_L1B	3,4
AST_04	3,4
AST_05	3,4
AST_06V	3,4
AST_06S	3,4
AST_06T	3,4
AST_08	3,4
AST_09	3,4
AST_09T	3,4
AST14DEM	3,4
MODIS Products	
MOD021KM	3,4
MOD02HKM	3,4
MOD02QKM	3,4
MOD09GQK	3,4
MOD09GHK	3,4
MOD09GST	3,4
MOD10_L2	3,4

MOD11_L2	3,4
MOD09A1	3,4
MOD09Q1	3,4
MOD10A1	3,4
MOD10A2	3,4
MOD11A1	3,4
MOD11A2	3,4
MOD11B1	3,4
MOD12Q1	3,4
MOD13A1	3,4
MOD13A2	3,4
MOD13Q1	3,4
MOD14A1	3,4
MOD14A2	3,4
MOD15A2	3,4
MOD17A2	3,4
MOD29	3,4
MOD29P1D	3,4
MOD29P1N	3,4
MOD43B1	3,4
MOD43B3	3,4
MOD43B4	3,4
Digital Elevation Model	
DEM_1KM	3,4

3.6 Open Location Services (OpenLS)

The OpenGIS Location Services (OpenLS) represent an open (middleware) platform for location-based application services for mobile assets and terminals. The primary goal of the OpenLS initiative series is to define the specifications for the "Core Services and Abstract Data Types (ADT)" that comprise this platform. The services are defined in two specifications that are now in the RFC stage within OGC TC. To date, seven services have been defined:

1) Directory Service - A network-accessible service that provides access to an online directory (e.g. Yellow Pages) to find the location of a *specific* or *nearest* place, product or service.
2) Gateway Service - A network-accessible service that fetches the position of a known mobile terminal from the network. This interface is modeled after the Mobile Location Protocol (MLP), Standard Location Immediate Service, specified in LIF 3.0 (see Open Mobile Alliance).
3) Geocoder Service - A network-accessible service that transforms a description of a location, such as a place name, street address or postal code, into a normalized description of the location with a Point geometry (see GML Specification for OGC geometry).
4) Navigation Service - An enhanced version of the Route Service, which is a network-accessible service that determines routes between two or more points and the associated navigation information.
5) Presentation Service - A network-accessible service that renders a base map made up of any geospatial data, with a set of ADT's as overlays (e.g., one's current position and the location of a desired Point of Interest displayed on a local map).
6) Reverse Geocoder Service - A network-accessible service that transforms a given position into a normalized description of a feature location (Address with Point), where the address may be defined as a street address, intersection address, place name or postal code.
7) Route Service - A network-accessible service that determines travel routes and navigation information between two or more points.

Whereas the specifications are considered to be stable, the objectives of this OpenLS Thread are:

1) To enhance the existing specs by adding new features to existing services,
2) To harmonize the OpenLS ADTs with GML 3.0
3) To enable SOAP messaging.
4) To enable seamless indoor-indoor location services.
5) To add new specifications for traffic and tracking services

3.6.1 OpenLS Enterprise Viewpoint

An Enterprise Viewpoint provides a high-level system Concept and Use Cases. The System Concept illustrates the operational setting, major system components, and major interfaces. The Use Cases provide descriptions of the behavior of the system from the point of view of Users. For OpenLS , the System Concept is in this section. We will provide high-level use cases in this section. However more detailed Use Cases may be developed during OpenLS.

OpenLS markets are wide and varied. They include any markets applications for consumers, business and government users where location-enabled mobile assets or mobile terminals are or may be employed, including:

- Wireless Location-Based Services (LBS)
- Automatic Vehicle Location (AVL) systems
- Telematics
- Asset tracking and management
- Seamless indoor-outdoor navigation service
- Emergency services and homeland security
- Mobile workforce applications (field operations, field service, field inventory, etc.)

Figure 6**Error! Reference source not found.**, illustrates a typical operational configuration for a wireless mobile phone service provider. Subscribers receive their services over the network via a Portal Platform, which handles authentication, billing and other basic service needs. Application services run on a designated Service Platform. The OpenLS platform is called upon to be perform location-based services, as needed. A GMLC/MPC determines positions of mobile terminals. The OpenLS platform may also call upon Web-based geospatial data and/or processing capabilities via OGC Web Services.

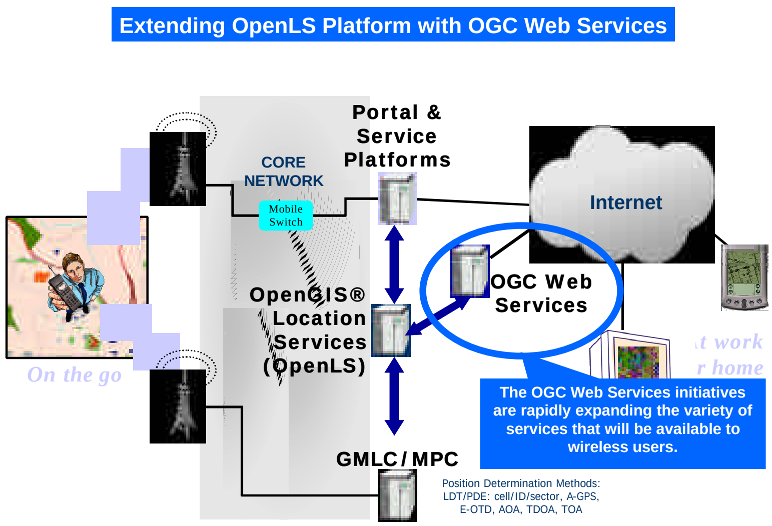


Figure 6: Geospatial One-Stop Portal Concept

3.6.2 OpenLS Information Viewpoint

- GML 3.0 for OpenLS geometries
- Abstract Data Types ADT: The basic information construct used to encode OpenLS information. Consists of well-known data types and structures for location information.

3.6.3 OpenLS Computational Viewpoint

The Core Services are location-based application services that form the Services Framework for the GeoMobility Server.

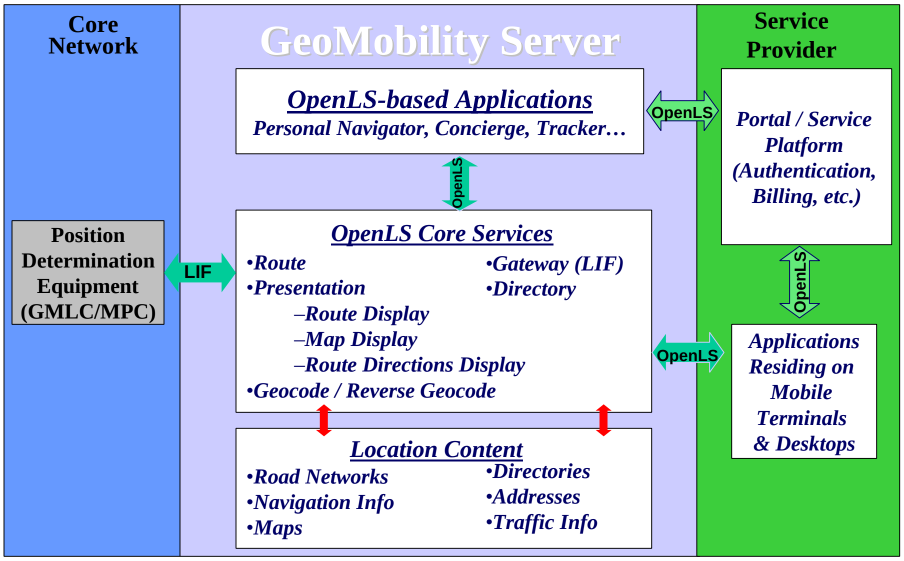


Figure 7. The GeoMobility Server

3.6.3.1 Directory Service

This service provides subscribers with access to an online directory to find the nearest or a specific place, product or service. Through a suitably equipped OpenLS application, the subscriber starts to formulate the search parameters in the service request, identifying the place, product or service that they seek by entering the name, type, category, keyword, phone number, or some other 'user-friendly' identifier. A position must also be employed in the request when the subscriber is seeking the nearest place, product or service, or if they desire a place, product or service at a specific location or within a specific area. The position may be the current Mobile Terminal position, as determined through the Gateway Service, or a remote position determined in some other manner. The directory type may also be specified (e.g. yellow pages, restaurant guide, etc). Given the formulated request, the Directory Service searches the appropriate online directory to fulfill the request, finding the nearest or specific place, product or service, depending on the search criteria. The service returns one or more responses to the query (with locations and complete descriptions of the place, product, or service, depending upon directory content), where the responses are in ranked order based upon the search criteria.

3.6.3.2 Gateway Service

This is the interface between the GeoMobility Server and Location Server that resides in the GMLC or MPC through which OpenLS services obtain position data for Mobile Terminals. This interface is modeled after the MLP specified in LIF 3.0 for Standard Location Immediate Service.

3.6.3.3 Location Utility Service (Geocoder/ Reverse Geocoder)

This service performs as a geocoder by determining a geographic position, given a place name, street address or postal code. It also returns a complete, normalized description of the place (which is useful, say, when only partial information is known). The service also performs as a reverse geocoder by determining a complete, normalized place name/street address/postal code, given a geographic position. Both the geocoder and reverse geocoder may return zero, one, or more responses to a service request, depending on subscriber request information, the algorithm being employed, and the match criteria.

3.6.3.4 Presentation Service

This service renders geographic information for display on a Mobile Terminal. Any OpenLS Application may call upon this service to obtain a map of a desired area, with or without map overlays that depict one or more OpenLS ADTs, such as Route Geometry, Point of Interest, Area of Interest, Location, Position and/or Address. The service may also be employed to render route directions from Route Maneuver List ADT and/or Route Instructions List ADT.

3.6.3.5 Route Service

This service determines a route for a subscriber. The subscriber must use a navigation application to set up the use of the service. They must indicate the start point (usually the position acquired through the Gateway Service, but this could be a planned trip from a specified location, say, from their home), and the endpoint (any location, like a place for which they only have the phone number or an address, or a place acquired through a search to a Directory Service). The subscriber may optionally specify waypoints, in some manner, the route preference (fastest, shortest, least traffic, most scenic, etc.), and the preferred mode of transport. The subscriber may optionally store a route for as long as needed, thus requiring the means to also fetch a stored route.

3.6.4 OpenLS Engineering Viewpoint

Figure 8 conveys the general architecture. An end-user application service such as a Buddy Finder, Personal Navigator or Concierge Service running on a host Service Platform calls upon OpenLS services for location-based content and processing, as needed. In this figure we also show a possible configuration for the new traffic and tracking services, with back-office applications. Tracking Service and Traffic Service will need timely access to mobile device location information. These services can access the GMLC / MPC directly using OMA/LIF MLP. The challenge OGC addresses through OpenLS is to define the foundational components and related specifications for near-time tracking, and navigation services.

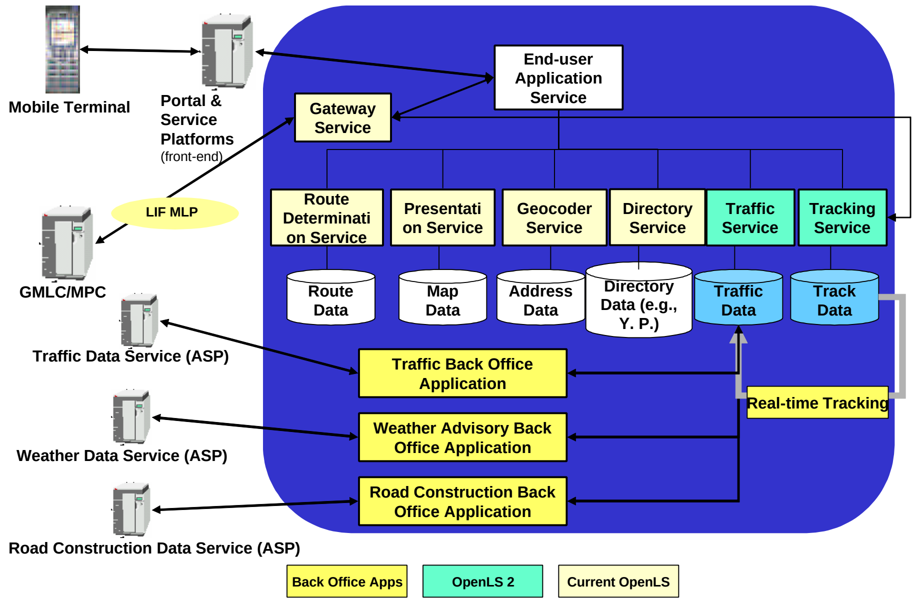


Figure 8 OpenLS Engineering View

3.6.5 *OpenLS Specific Activities During OWS2*

3.6.5.1 *Enhance Existing Specifications*

3.6.5.1.1 *Directory Services*

1) Provide a mechanism for directory service requests to specify the list of properties to return with each feature. This includes ensuring that all types of properties (spatial and non-spatial) can be returned with features.
2) Enhance filtering mechanism to limit or qualify features returned from a directory service query. In a manner similar to WFS, it needs to be able to filter based on an arbitrary set of properties and specify the filter matching operation. Matching operations include: Exact match, begins with, contains, like, less then, greater then, etc.
3) Enhance Directory Service to deal with enhanced query capabilities and a greater variety of directory information types.
4) Add the ability to provide access to POIs where they can be obtained from navigable network “links” directly. This activity is intended to enhance performance.

3.6.5.1.2 *Route and Navigation Services*

1) Bring Navigation Service functionality into the Route service so that there is just one service, with optional components.
2) Enrich the Navigation capability of the Route Service.

3.6.5.1.3 *Gateway Service*

1) Upgrade the Gateway service to meet the needs of the new Tracking Service.

3.6.5.1.4 *Presentation Service*

1) Enhance Presentation Service to deal with the new Abstract Data Types for Tracking and Traffic services and to account for enhancements to existing ADTs.
2) Enhance the Presentation Service to support SVG, or formulate a new service.
3) Add capabilities and options to map-display part of Presentation Service

3.6.5.1.5 Positioning Refinement Service

Allow a current GPS position, to return not just an x, y on the map, but actually find the appropriate relevant map feature, regardless of if it is actually the nearest one geographically.

To overcome the errors associated with GPS coordinates, which may lead to returning irrelevant map feature, an interactive exchange between man and Positioning Refining Service would allow several back and forth interaction to finally zero on a single feature, even if not via "tradition" mechanisms.

3.6.5.2 GML 3.0/ADT Harmonization

The ADTs are XML constructs designed to work with software applications that run on wireless devices. As part of a operational system, it may be necessary to transfer information from a wireless device used by a field worker, to an analyst’s workstation that interfaces with a store/production system. To enable this transfer, this effort will harmonize the Abstract Data Types developed in the OpenLS OGC IP initiative with the base schemas of GML 3.0. This harmonized work was started in the OLS 1.1 Initiative and needs to be completed. Once done, the appropriate Change Proposal(s) to the OLS Specifications, Open Location Services Core Services Request for Change (RFC) need to be generated.

3.6.5.3 Messaging Framework

Develop OpenLS Messaging Framework based on SOAP 1.2.

3.6.5.4 Indoor-Outdoor Navigation

Indoor-outdoor Navigation refers to the means for clients with mobile terminals to receive location and navigation guidance indoors or outdoors, as well as receive navigation guidance across indoor-outdoor and indoor-indoor transition points (e.g., doorways). Indoor location concepts must be supported for how people identify location for indoor environments, e.g., building, floor, room, etc. Indoor navigation concepts must also be supported for how people negotiate their way around indoor environments, e.g., park on level P1-P4, elevator to 3rd floor, right hall to room 310.

The present OpenLS services and information model are limited to outdoor navigation (i.e., the concepts of ‘location’ and ‘navigation’ are confined to outdoor activities.) An enhancement to the OpenLS services and information model is to support seamless indoor-outdoor navigation. The OpenLS services and information model will have to be modified to accommodate indoor location and navigation constructs.

3.6.5.5 Develop Tracking Services

1) The means to specify when (time period) the device or mobile device should be tracked.
2) The means to specify the criteria (temporal & spatial) that should be used to generate a notification or alert. This includes periodic updates and notifications when a mobile device enters or exits a region, starts, stops, moves a specific distance or when a mobile device interacts with another mobile device.

3) A well-defined mechanism and message format used to send a notification or alert to the client application.
4) The means to access position/time/speed/direction information for one or more mobile terminals/assets through the Gateway Service.
5) The means to store position/time/speed/direction information for one or more mobile terminals/assets for a designated time period and sampling periodicity.
6) The means to fetch position/time/speed/direction information tracks for one or more mobile terminals/assets for a designated time period.

3.6.5.6 Develop Traffic Services

1) The means to specify when (the time period) the traffic incidents or conditions should be monitored and reported.
2) The means to specify the criteria (temporal & spatial) that should be used to generate a notification or alert. This includes traffic for a specific area of interest, or along a route as determined by the Route Service.
3) A well-defined mechanism and message format used to send a notification or alert to the client application.
4) The means to report traffic incidents and/or conditions for a specified area of interest, road, or road segment for a designated time period.
5) The means to report traffic incidents and/or conditions for a designated route (that has been determined by a Route Service or Navigation Service) for a designated time period.
6) The system extracts existing real-time traffic data from public and private agencies, generates a graphic map depicting the traffic conditions, formats the graphic data for depiction on a web page, and transfers the page to a web server.

Appendix A: OWS 2 Architecture References

Refer to the OGC website (http://www.opengis.org/specs/?page=baseline) for the authoritative listing of adopted documents.

Note: Please contact the OGC Tech Desk if you need assistance in gaining access to these documents (techdesk@opengis.org).

OGC Specifications and Supporting Documents Relevant to EMS-1:

1. Style Management Services (SMS) Discussion Paper, OpenGIS® Project Document 03-031. Available at: http://www.opengis.org/specs/?page=discussion
2. Style and Symbol Management Services Requirements Interoperability Program Report, OpenGIS® Project Document 03-030. Available at: ***http://member.opengis.org/tc/archive/arch03.htm***
3. OpenGIS® Style Layered Description (SLD) Implementation Specification, version 1.0, available at: ***http://www.opengis.org/specs/?page=specs***
4. OpenGIS® Geography Markup Language (GML) Implementation Specification (version 3.0), available at: http://www.opengis.org/specs/?page=specs
5. OpenGIS® Web Feature Server (WFS) Implementation Specification, version 1.0, available at: ***http://www.opengis.org/specs/?page=specs***
6. OpenGIS® Filter Encoding Implementation Specification, version 1.0, available at: ***http://www.opengis.org/specs/?page=specs***
7. OpenGIS® Web Map Service (WMS) Implementation Specification, version 1.2, available at: ***http://www.opengis.org/specs/?page=specs***
8. OpenGIS® Map Context Documents Implementation Specification, version 1.0, available at: ***http://www.opengis.org/specs/?page=specs***
9. Coverage Portrayal Service Specification (CPS) Interoperability Program Report. OpenGIS Project Document 02-019r1, available at: ***http://member.opengis.org/tc/archive/arch02.htm***
10. OpenGIS® Catalog Service Implementation Specification, version 1.1.1, available at: ***http://www.opengis.org/specs/?page=specs***
11. OpenGIS® Web Registry Service Interoperability Program Report. OpenGIS® Project Document 03-024**.** Available at: ***http://member.opengis.org/tc/archive/arch03.htm***
12. Registry Service Requirements Interoperability Program Report. OpenGIS® Project Document 03-027. Available at: ***http://member.opengis.org/tc/archive/arch03.htm***
13. Web Object Service Discussion Paper, OpenGIS® Project Document 03-013. Available at http://www.opengis.org/specs/?page=discussion
14. Critical Infrastructure Collaborative Environment (CICE) Architecture: Enterprise Viewpoint Discussion Paper, OpenGIS Project Document 03-061. Available at http://www.opengis.org/specs/?page=discussion
15. Critical Infrastructure Collaborative Environment (CICE) Architecture: Information Viewpoint Discussion Paper, OpenGIS Project Document 03-062r1. Available at http://www.opengis.org/specs/?page=discussion
16. Critical Infrastructure Collaborative Environment (CICE) Architecture: Computation Viewpoint Discussion Paper, OpenGIS Project Document 03-063r1. Available at http://www.opengis.org/specs/?page=discussion

17. Critical Infrastructure Collaborative Environment (CICE) Architecture: Engineering Viewpoint Discussion Paper, OpenGIS Project Document 03-055r1. Available at http://www.opengis.org/specs/?page=discussion

Other OGC Specifications and Supporting Documents

18. OpenGIS® Abstract Specification, Topics 1-17), available at: http://www.opengis.org/specs/?page=abstract

19. OpenGIS® Reference Model, version 0.1.2, available at: http://www.opengis.org/specs/?page=orm

20. OpenGIS® Web Service Architecture Discussion Paper. OpenGIS Project Document 03-025, available at: http://www.opengis.org/specs/?page=discussion

21. OpenGIS® Grid Coverages Implementation Specification, version 1.0, available at: http://www.opengis.org/specs/?page=specs

22. OpenGIS® Coordinate Transformation Services Implementation Specification, version 1.0, available at: http://www.opengis.org/specs/?page=specs

23. Web Coverage Service Implementation Specification, Version 0.0. OpenGIS® Project Document 02-076r3, available at http://www.opengis.org/specs/?page=specs

24. OpenGIS® Location Services (OpenLS): Core Services [Parts 1-5], OpenGIS® Project Document 03-006r1. Available at: http://www.opengis.org/specs/?page=specs

25. OpenGIS® Location Services (OpenLS): Navigation Service [Part 6], OpenGIS® Project Document 03-007r1. Available at: http://www.opengis.org/specs/?page=specs

26. Service Information Model Discussion Paper, OpenGIS® Project Document 03-026. Available at: http://www.opengis.org/specs/?page=discussion

27. Web Terrain Service Discussion Paper, OpenGIS® Project Document 01-061. Available at: http://www.opengis.org/specs/?page=discussion

28. XML for Imagery and Map Annotations (XIMA) Discussion Paper, OpenGIS® Project Document 01-019. Available at: http://www.opengis.org/specs/?page=discussion

29. Web Coordinate Transformation Service Discussion Paper, OpenGIS® Project Document 02-061r1. Available at http://www.opengis.org/specs/?page=discussion

30. Recommended Definition Data for Coordinate Reference Systems and Coordinate Transformations, OGC Recommendation Paper, version 1.1.0, available at: http://www.opengis.org/specs/?page=recommendation

ISO Specifications

31. ISO 19101:2002 (Reference Model): http://webstore.ansi.org/ansidocstore/product.asp?sku=ISO+19101:2002

32. ISO 19107 (Spatial Schema) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19107_(E).pdf

33. ISO 19108 (Temporal Schema) : http://www.isotc211.org/protdoc/DIS/DIS19108.pdf

34. ISO 19109 (Rules for Application Schema) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19109_(E).pdf

35. ISO 19110 (Methodology for Feature Cataloguing) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19110_(E).pdf

36. ISO 19111 (Spatial Referencing by Coordinates) : http://www.isotc211.org/protdoc/DIS/DIS19111.pdf

37. ISO 19112 (Spatial Referencing by Geographic Identifiers) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19112_(E).pdf

38. ISO 19115 (Metadata) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19115_(E).pdf

39. ISO 19117 (Portrayal) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19117_(E).pdf

40. ISO 19119 (Services) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19119_(E).pdf

41. ISO 19123 (Schema for Coverage Geometry and Functions): http://www.isotc211.org/protdoc/211n1227/readme.htm

42. ISO 19125-1 (Simple Features Access - Part 1: Common Architecture): http://www.isotc211.org/protdoc/DIS/DIS19125-1.pdf

43. ISO 19125-2 (Simple Features Access - Part 2: SQL option): http://www.isotc211.org/protdoc/DIS/DIS19125-2.pdf

44. ISO 19128 (Web Mapping): http://www.isotc211.org/protdoc/211n1331/211n1331.pdf

Other Related Specifications:

45. Uniform Resource Identifiers (URI): Generic Syntax (RFC 2396) T. Berners-Lee, R. Fielding, L. Masinter, available at: http://www.ietf.org/rfc/rfc2396.txt

46. Extensible Markup Language (XML) 1.0, Second Edition, Tim Bray et al., eds., W3C, 6 October 2000. See http://www.w3.org/TR/2000/REC-xml-20001006

47. XML Schema Part 1: Structures. World Wide Web Consortium (W3C).. W3C Recommendation (2 May 2001). Available [online]: http://www.w3.org/TR/xmlschema-1/

48. XML Linking Language (XLink) Version 1.0, DeRose, S., Maler, E., Orchard, D., available at http://www.w3.org/TR/xlink/

49. Web Services Description Language (WSDL) Version 1.2. W3C Working Draft (9 July 2002). World Wide Web Consortium (W3C). Available [online]: http://www.w3.org/TR/wsdl12/

50. Simple Object Access Protocol (SOAP) 1.1, Box, D., et. al., available at http://www.w3.org/TR/SOAP/

51. UDDI – Universal Description, Discovery, and Integration, see http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=uddi-spec

Appendix B: Scenario Narrative to Demonstrate Functionality to be Developed and Demoed in OWS-2

See attachment.