OGC®

Open Geospatial Consortium

Technology Office

4899 North Old State Road 37
Bloomington, IN 47408

Telephone: +1-812-334-0601
Facsimile: +1-812-961-2053

Request for Quotation (RFQ)

And

Call for Participation (CFP)

OGC Web Services Initiative - Phase 6 (OWS-6)

Annex B

OWS-6 Architecture

RFQ Issuance Date: July 21, 2008

Proposal Due Date: September 5, 2008

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Overview

The architectures presented in this Annex are based upon a collaborative effort between OGC Web Services 6 (OWS-6) 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-6 development threads.

Section 3 discusses the architectural approach and technical baseline for OWS-6.

Section 4 discusses the architectural approaches and issues for each of the OWS-6 development threads.

The OGC portal provides a Glossary of Terms at the following URL that may be useful to aid in understanding and interpretation of terms and abbreviations contained throughout this RFQ:

http://www.opengeospatial.org/resources/?page=glossary

2 OWS-6 Initiative Threads

The OGC is engaged in an Interoperability Program which is a global, hands-on and collaborative prototyping program designed for rapid development and delivery of proven candidate specifications into OGC's Specification Program which can then be formalized for public release. In OGC's Interoperability Initiatives, international technology developers and providers team together to solve specific geo-processing interoperability problems posed by the initiative's sponsoring organizations. OGC Interoperability Initiatives include test beds, pilot projects, interoperability experiments, and interoperability support services – all designed to encourage rapid development, testing, validation and adoption of open, consensus based standards specifications.

The policies and procedures that define the OGC Interoperability Program are available here:

http://www.opengeospatial.org/about/?page=ippp

In March of 2008, the OGC issued a call for sponsors for an OGC OWS-6 Interoperability initiative testbed activity to advance OGC's open framework for interoperability in the geospatial industry. Three meetings were conducted with potential OWS-6 sponsors to review the OGC technical baseline, discuss OWS-5 results, and identify OWS-6 requirements. Sponsors have expressed keen interest in advancing standards for sensor webs, semantics, 3-D routing and visualization, avionics support, and geospatial processing. After analyzing the sponsors input, the OGC Interoperability Team recommended to the sponsors that the content of the OWS-6 initiative be organized around the following six threads:

1) Sensor Web Enablement (SWE)
2) Geo-Processing Workflow (GPW)
3) Aviation Information Management (AIM)
4) Geospatial Decision Support Services (DSS)
5) Compliance and Interoperability Test and Evaluation (CITE)

An introduction to each of these six threads is given below, followed by a detailed discussion of the architectural implications of the initiative threads.

2.1 Sensor Web Enablement (SWE)

The OGC Sensor Web Enablement framework has achieved a degree of maturity through previous OWS interoperability initiatives and deployments worldwide. OWS-6 will focus on integrating the SWE interfaces and encodings into cross-thread scenarios and workflows to demonstrate the ability of SWE specifications to support operational needs.

Emphasis for SWE during this phase of the OWS test-bed will be on:

- CCSI-Enabled CBRN Sensors into the SWE Environment
- Build on Georeferenceable imagery accomplishments of OWS-5
- Harmonize SWE-related information models: SensorML, GML, UncertML, MathML
- Apply GeoRM, Trusted Services, and security models in SWE environment
- Events-based architecture including WNS

2.2 Geo-Processing Workflow (GPW)

This GPW thread aims to build on the progress of previous testbeds with a focus on maturing the interoperability and capabilities of OGC web services in a service-oriented architecture with particular emphasis to address OGC web service security issues. To satisfy mission-critical goals, the architecture must not only provide for integration of a wide variety of service capabilities and resources, it must do so and ensure authenticity, integrity, quality and confidentiality of services and information.

To meet these goals, the following task areas have been identified:

- Asynchronous Workflow and Web Services Security
- Data Security for OGC web services
- Data Accessibility
- WPS Profiles - Conflation; and Grid processing
- GML Application Schema Development & ShapeChange Enhancements

2.3 Aviation Information Management (AIM)

The Aviation Information Management (AIM) subtask is a new thread within OWS to develop and demonstrate the use of the Aeronautical Information Exchange Model (AIXM) in an OGC Web Services environment. The AIM subtask shall focus on evaluating and advancing various AIXM features in a realistic trans-Atlantic aviation scenario setting by devising and prototyping a Web Services Architecture for providing valuable aeronautical information directly to flight decks, Electronic Flight Bags (EFB) and hand-held devices (such as PDAs and Blackberries) while the airplane is at the gate or en-route to its destination (for the purposes of OWS-6, the aeronautical information in the latter case does not depend on the knowledge of the airplane's location).

AIXM was developed by the Federal Aviation Administration (FAA) and Eurocontrol as a global standard for the representation and exchange of aeronautical information. It was designed as a basis for digital aeronautical information exchange and for enabling the transition to a net-centric, global aeronautical management capability. AIXM has been developed using the ISO 19100 modeling framework and has two major components: a conceptual model presented in the form of an UML class model and a data encoding specification which was developed using the OGC Geography Markup Language (GML). Both have been tailored to the specific requirements for the representation of aeronautical objects, especially the temporality feature that allows for time-dependent changes affecting AIXM features. More information about AIXM is available on www.aixm.aero.

In support of the above objectives, the OWS-6 AIM thread shall perform tasks in the following areas while ensuring that the integrity of data is preserved throughout all data exchange operations:

- Use and enhancement of Web Feature Service and Filter Encoding specifications in support of AIXM features and 4-dimensional flight trajectory queries,
- Prototype of Aviation client for retrieval and seamless visualization of AIXM, Weather and other aviation-related data, emphasizing time and spatial filtering in order to present just the right information into a given user context anytime, anywhere,
- Architecture of standards-based mechanism to notify users of changes to user-selected aeronautical information.

2.4 Geospatial Decision Support Services (DSS)

Decision Support Services having an emphasis on applications of geospatial and temporal information has been a recurring thread in previous OWS testbeds. This thread focuses on presenting and interacting with data obtained from the sensor web and geoprocessing workflows in the most effective ways to support analysis and decision making. The focus for DSS in OWS-6 builds on portrayal, WMS Tiling, and integrated client work from OWS-4, with additional work on 3D visualization and integration of the built environment and landscape.

This thread will encompass these capabilities and task areas:

- ISO 19117 and OGC SLD Portrayal
- 3D Portrayal of GML with Fly-through
- Hosting CityGML data with WFS
- Outdoor and indoor 3D route and tracking services
- WMS performance (tiling)
- Integrated Client for multiple OWS services

2.5 Compliance and Interoperability Test and Evaluation (CITE)

Validating compliance with an OGC specification means verifying that a software product has implemented the specification correctly by testing the software interface for response and behavior that is outlined in the specification. Verifying compliance to the standard is necessary in order to achieve interoperability. As a result, geospatial application vendors desire to provide their potential costumers a means to verify adherence to OGC standards as a measurable discriminator for the interoperability of software products. Similarly, users desire assurance that acquired software components will interoperate with their existing investments in OGC-compliant technology. The Conformance and Interoperability Test and Evaluation (CITE) thread is intended to provide the geospatial industry (consumers and vendors) a methodology and tools that will test compliance with OGC web services.

The OGC Interoperability Program and the OGC Specification Program have achieved a great deal of momentum as a result of the multiple OGC web service specifications that have recently been published. Key consumers in the geospatial industry are modernizing their enterprises based on the applicability and interoperability of OGC web services. The major geospatial industry consumers require verifiable proof of compliance with OGC specifications in order to reach the desirable outcome of interoperability. Furthermore, as the OGC technology stack has matured, a group of interfaces has emerged that represents a baseline of technology needed to implement a fully interoperable, end-to-end *spatial data infrastructure*. The CITE threads in previous OWS projects have made significant progress towards having a complete suite of compliance tests for this baseline of interfaces.

A major focus of OWS-5 was on achieving consensus on the format and content of an Abstract Test Suite (ATS). The OWS-5 CITE participants agreed to follow the ISO guidance for writing Abstract Test Suites. The ATS are used to develop Executable Test Suites which are the scripts that the TEAM Engine runs to conduct an automated compliance test. A major focus of OWS-6 CITE will be in clearly documenting the approach to defining Abstract Test Suites. This will be a great benefit to the OGC community as the OGC Architecture Board (OAB) requires that new specifications be published with an accompanying ATS. In addition, a focus of OWS-6 CITE will be to expand the usability of the existing OGC compliance tests by "tailoring" these tests for specific schema profiles and/or data.

3 OWS-6 Baseline

3.1 OpenGIS® Reference Model

Relevant Specifications: OpenGIS® Reference Model version 0.1.3

(http://portal.opengeospatial.org/files/?artifact_id=3836)

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 Standards Baseline. The OGC Standards Baseline consists of the member approved Implementation/Abstract Specifications as well as for a number of candidate specifications that are currently in progress.

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. It is encouraged that respondents to this RFQ understand the concepts that are presented in the ORM.

The structure of the ORM is based on the Reference Model for Open Distributed Processing (RM-ODP). This Annex of the OWS-6 RFQ will deal with the upper four views; Enterprise, Information, Computational, and Engineering as shown in the figure below. Each thread of the initiative will be described in the annex using any or all of these four views.

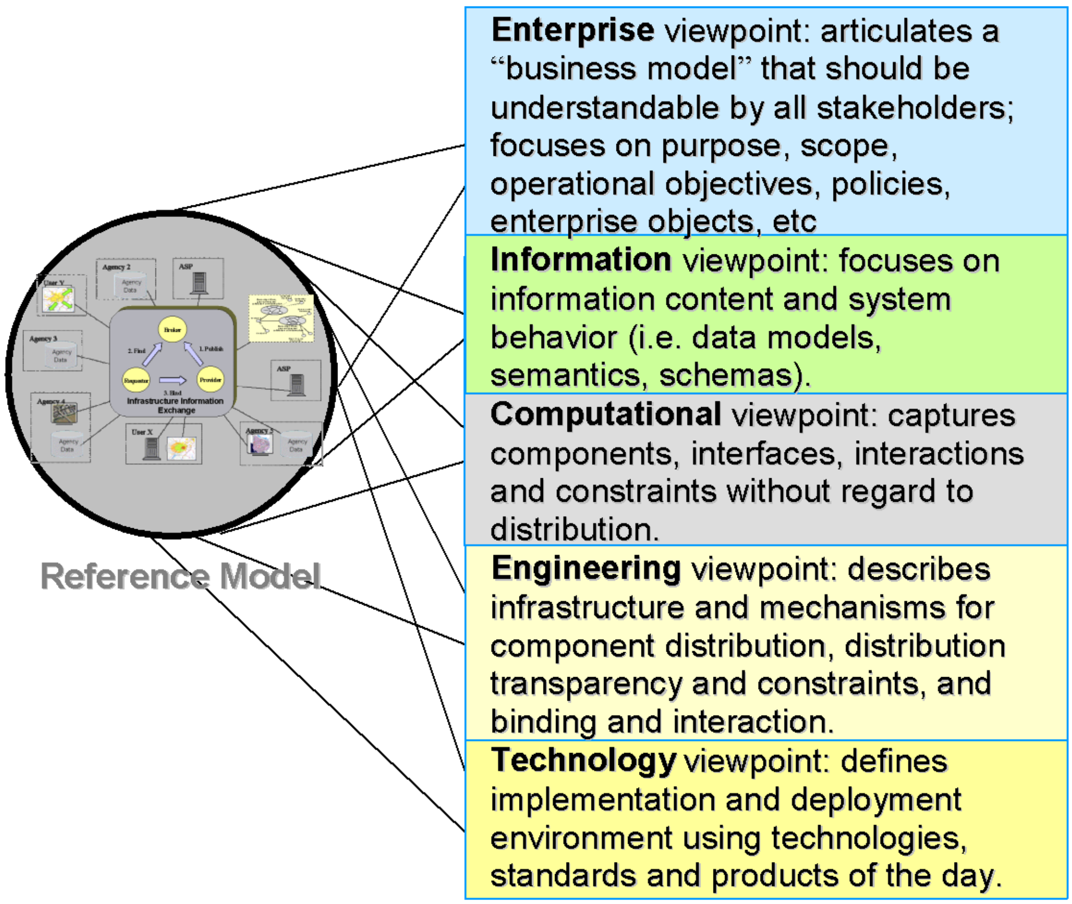


Figure 3-1. OGC Reference Model (ORM)

3.2 Geospatial Digital Rights Management Reference Model

Relevant Specifications: OpenGIS® Geospatial Digital Rights Management Reference Model version 1.0.0, OGC Document 06-004r3 (2006-02-28) http://portal.opengeospatial.org/files/?artifact_id=14085

In March, 2007 the OGC membership approved the Geospatial Digital Rights Management Reference Model (GeoDRM RM), an abstract specification for the management of digital rights in the area of geospatial data and services. The GeoDRM RM is Topic 18 of the OpenGIS® Abstract Specification.

The goal of the GeoDRM effort in the OGC is to make sure that a larger market has access to geospatial resources through a well understood and common mechanism that enables more than today's "all or nothing" protection. A major motivation for this effort is the need to manage the "ownership obstacle to data sharing" in spatial data infrastructure scenarios.

The GeoDRM RM defines the framework for web service mechanisms and rights languages to articulate, manage and protect the rights of all participants in the geographic information marketplace, including the owners of intellectual property and the users who wish to use it. A key aspect of the GeoDRM RM is that it is abstract, or general, rather than specifying implementation details about types of agreements. Such agreements might range from an open content sharing model to a cost-recovery program of a public or government organization or a full commercial vendor license model.

The OGC membership will use the GeoDRM RM in developing OpenGIS Implementation Specifications for open interfaces and encodings that will enable spatial data infrastructures (SDI) and diverse systems to participate in transactions involving data, services and intellectual property protection. Maintenance and further development of the GeoDRM RM is the responsibility of the GeoRM Working Group in the OGC Technical Committee.

3.3 OGC Standards Baseline

The OGC Standards Baseline, at any point in time, is the set of all Adopted Specifications plus all other technical documents that have been made available to the public by the OGC Technical and Planning Committees. The Standards Baseline all member approved implementation/abstract specifications and best practices documents. These specifications and other documents are freely available to the public at this website: http://www.opengeospatial.org/standards

With the exception of the CITE thread, OWS-6 will use GML version 3.2.1 which is available as OGC Document 07-036, subject to corrections and revisions documented in OGC 07-061. This version of GML is also adopted as ISO 19136:2007. Many, if not most, specifications that depend on GML are now based on GML 3.1.1. We shall allow a migration path here.

Several documents were approved for public release at the recent OGC TC/PC meeting in June 2008 and are being processed for public release. These documents will become available over the next few weeks. If a specific document is needed by a proposer and it has not yet been published to the public link above, contact OGC (techdesk@opengeospatial.org).

The following table lists the approved OGC Specifications that are relevant to OWS-6. In some cases OWS-6 may specify a version which is different from the approved specifications in this table.

Table 3-1. Approved OGC Specifications Related to OWS-6

IMPLEMENTATION SPECIFICATIONS			
Title	**Version**	**Document #**	**Date**
Catalog Service	2.0.2	07-006r1, 07-010, 07-110r2, 07-144r2, 07-045	2007-02-23
Web Coverage Service (WCS)	1.1.2	07-067r5, 07-066r5	2008-03-19
Web Feature Service (WFS)	1.1	04-094	2005-05-03
Web Map Service (WMS)	1.3.0	06-042	2006-03-15
Web Map Context (WMC)	1.1	05-005, 08-050	2005-01-19, 2008-03-14
Web Processing Service (WPS)	1.0	05-007r7	2007-06-08
Web Service Common	1.1	06-121r3	2007-02-09
Geography Markup Language (GML)	3.2.1	07-036, 07-061	2007-08-27
Styled Layer Descriptor (SLD)	1.1	05-078r4	2007-06-29
Symbology Encoding (SE)	1.1	05-077r4	2007-1-18
Filter Encoding (FE)	1.1	04-095	2005-05-03

Geospatial eXtensible Access Control Markup Language (GeoXACML)	1.0	07-026r2, 07-098r1, 07-099r1	2008-02-20
KML	2.2	07-147r2	2008-04-14
Open Location Services (OpenLS)	1.1	05-016	2005-05-02
Observations and Measurements - Part 1: Observation schema	1.0	07-022r1, 08-022r1	2007-12-08, 2008-02-25
Observations and Measurements - Part 2: Sampling Features	1.0	07-022r3	2007-12-08
SensorML with corrigendum	1.0.1	07-000, 07-122r2	2007-07-17, 2007-10-25
Sensor Observation Service	1.0	06-009r6	2007-10-26
Sensor Planning Service	1.0	07-014r3	2007-08-02

3.4 OGC Candidate Standards

The following documents are currently being voted on for adoption as OpenGIS Implementation Specifications. To request a copy of these documents, please email the OGC Technology Desk (techdesk@opengeospatial.org).

Table 3-2. Candidate Standards Related to OWS-6

Title	Version	Document #	Date
CityGML Candidate Implementation Specification	1.0	08-007r1, 08-093	2008-05-19
Web Coverage Service (WCS) - Transaction Operation Extension	1.1.4	07-068r3	2008-05-15
Web Coverage Service (WCS) - Processing Extension	1.0.0	08-059r1	2008-04-29

3.5 GML Profiles

Attention is called to the fact that numerous GML Profiles (functional subsets of GML designed for specific applications) have been developed, which are currently based on GML 3.1.1. Presumably these may need only minor changes to become compliant with GML 3.2.1, and in some cases such as GML Simple Features Profile, an OGC Revision Working Group is already working on a draft Change Request to bring this Profile into compliance with GML 3.2.1. The following profiles should be used wherever appropriate, and any changes required for compliance with GML 3.2.1 should be documented as a Change Request to the appropriate OGC Working Group.

Table 3-3. GML 3.1.1 Profiles which may be of use in OWS-6

Title	Version	Document #	Date
GML 3.1.1 common CRSs profile	1.0.0	05-095r1	2006-07-18

Title	Version	Document #	Date
GML 3.1.1 common CRSs profile Corrigendum	1.0.1	06-113	2006-07-19
GML 3.1.1 grid CRSs profile	1.0.0	05-096r1	2006-07-18
GML 3.1.1 grid CRSs Profile Corrigendum	1.0.1	06-111	2006-07-19
GML 3.1.1 CRS support profile	1.0.0	05-094r1	2006-07-18
GML 3.1.1 simple dictionary profile	1.0.0	05-099r2	2006-07-18
GML 3.1.1 simple features profile	1.0.0	06-049r1	2006-05-08

3.6 OGC Best Practices Baseline

Best Practice Documents contain discussion of best practices related to the use and/or implementation of an adopted OGC document and for release to the public. Best Practices Documents are an official position of the OGC and thus represent an endorsement of the content of the paper. These Best Practice Documents have been made available at the following website: http://www.opengeospatial.org/standards/bp.

Table 3-4. Approved OGC Best Practice Documents Related to OWS-6

Title	Version	Document #	Date
A URN namespace for OGC	0.4	07-107r3	2008-05-02
Binary Extensible Markup Language (BXML) Encoding Specification	-	03-002r9	2006-01-18
City Geography Markup Language	0.4.0	07-062	2007-08-14
Definition identifier URNs in OGC namespace	1.1.2	07-092r1	2007-11-14
FGDC CSDGM Application Profile for CSW 2.0	-	06-129r1	2006-12-26
Gazetteer Service - Application Profile of the Web Feature Service Implementation Specification	0.9.3	05-035r2	2006-07-27
GML Application Schema for EO Products	0.9.0	06-080r2	2007-08-16
GML Encoding of Discrete Coverages (interleaved pattern)	0.2.0	06-188r1	2007-05-17
GML PIDF-LO Geometry Shape Application Schema for use in the IETF	0.1.0	06-142r1	2007-05-17
Ordering Services for Earth Observation Products	0.9.0	06-141r2	2007-08-15

Title	Version	Document #	Date
Reference Model for the ORCHESTRA Architecture	2.1.0	07-097	2007-10-05
Sensor Alert Service	0.9.0	06-028r5	2007-05-16
Sensor Planning Service Application Profile for EO Sensors	0.9.5	07-018r2	2008-01-21
Specification best practices	1.0.0	06-135r1	2007-01-29
Units of Measure Recommendation	1.0.0	02-007r4	2002-08-19
Web Coverage Processing Service (WCPS)	-	06-035r1	2006-07-26
Web Map Services - Application Profile for EO Products	0.2.0	07-063	2007-08-15
Web Notification Service	-	06-095	2007-01-25

The following OGC Best Practice Document was approved for Public Release at the last Technical and Planning Committees in June 2008 pending minor revisions. To request a copy of these documents, please email the OGC Technology Desk (techdesk@opengeospatial.org).

Table 3-5. OGC Best Practices Documents Pending Revisions

Title	Version	Document #	Date
EO Products Data Model for ebRIM Profile of CSW 2.0.2	0.1.9	06-131r4	2008-05-14
OGC Sensor Web Enablement Architecture ER	1.x.x	06-021r2	2008-03-31

3.7 OGC Discussion Papers Baseline

OGC Discussion Papers are documents that present technology issues being considered in the Working Groups of the Open Geospatial Consortium Technical Committee. Their purpose is to create discussion in the geospatial information industry on a specific topic. These papers do not represent the official position of the Open Geospatial Consortium nor of the OGC Technical Committee. These discussion papers have been made available at this website: http://www.opengeospatial.org/standards/dp.

Table 3-6. Discussion Papers Related to OWS-6

Title	Version	Document #	Date
A URN namespace for the Open Geospatial Consortium (OGC)	2.0.0	06-166	2007-01-30

Title	Version	Document #	Date
Access Control & Terms of Use (ToU) "Click-through" IPR Management "	1.0.0	05-111r2	2006-05-09
Catalog 2.0 Accessibility for OWS3	-	05-084	2006-05-09
Cataloguing of ISO Metadata (CIM) using the ebRIM profile of CS-W	0.1.7	07-038	2007-06-06
Compliance Test Engine Interoperability Program Report	1.0.0	07-012	2007-09-04
Compliance Test Language (CTL) Discussion Paper	0.4.0	06-126	2006-10-18
Discussions, findings, and use of WPS in OWS-4	0.9.1	06-182r1	2007-06-06
EO Application Profile for CSW 2.0	1.4.0	06-079r1	2006-06-06
Feature Portrayal Service	-	05-110	2006-04-19
Feature Styling IPR	0.4.1	06-140	2007-06-08
Frame image geopositioning metadata GML 3.2 application schema	-	07-032	2007-06-06
Geo Video Web Service	-	05-115	2006-03-28
GeoDSS Mass Market	-	07-004	2007-05-07
GeoDRM Engineering Viewpoint and supporting Architecture	0.9.2	06-184r2	2007-08-14
Geographic information - Rights expression language for geographic information - Part xx: GeoREL	0.9.0	06-173r2	2007-01-25
GEOINT Structure Implementation Profile (GSIP) Schema Processing	0.5.0	07-028r1	2007-05-17
Geolinked Data Access Service	0.9.1	04-010r1	2004-05-04
Geolinking Service	0.9.1	04-011r1	2004-05-04
Geospatial Portal Reference Architecture	0.2.0	04-039	2004-09-22
Geospatial Semantic Web Interoperability Experiment Report	0.5.0	06-002r1	2006-08-21
GML Performance Investigations by CubeWerx	1.0.0	05-050	2006-05-02
GML Point Profile	0.4.0	05-029r4	2005-08-29

Title	Version	Document #	Date
Imagery Metadata	1.0.0	05-015	2005-01-27
Local MSD Implementation Profile (GML 3.2.1)	0.7.0	07-027r1	2007-05-25
OGC Web Services (OWS) 3 UML to GML Application Schema (UGAS) Tool	-	05-118	2006-04-28
OGC Web Services Architectural Profile for the NSG	1.3.0	07-009r3	2007-08-13
OGC Web Services Architecture for CAD GIS and BIM	0.9.0	07-023r2	2007-05-16
OGC Web Services SOAP Experiment Report	0.8.0	03-014	2003-01-15
OGC Web Services UDDI Experiment	0.5.0	03-028	2003-01-17
OWS 2 Common Architecture: WSDL SOAP UDDI	1.0.0	04-060r1	2005-02-17
OWS 3 GML Investigations - Performance Experiment by Galdos Systems	-	05-101	2006-04-19
OWS 5 SOAP/WSDL Common Engineering Report	0.1.0	08-009r1	2008-02-21
Integrated Client for Multiple OGC-compliant Services	0.1.18	03-021	2003-01-20
OWS Integrated Client (GeoDSS Client)	-	05-116	2007-03-08
OWS Messaging Framework	-	03-029	2003-01-20
OWS-2 Application Schema Development	-	04-100	2005-04-13
OWS3 GML Topology Investigation	-	05-102r1	2006-05-09
OWS-3 Imagery Workflow Experiments: Enhanced Service Infrastructure Technology Architecture and Standards in the OWS-3 Testbed	0.9.0	05-140	2006-03-30
OWS4 - Topology Quality Assessment Interoperability Program Report	0.3.0	07-007r1	2007-06-06
OWS-4 CSW ebRIM Modelling Guidelines IPR	-	06-155	2007-06-06
Requirements for some specific simple solid, plane and line geometry types	0.5.0	07-001r3	2007-05-02
Schema Maintenance and Tailoring	-	05-117	2006-05-02
Some image geometry models	1.0.0	04-071	2004-10-04

Title	Version	Document #	Date
Temporal Standard Recommendations	-	06-022r1	2006-04-21
Tiled WMS Discussion Paper	0.3.0	07-057r2	2007-10-10
Trusted Geo Services IPR	0.9.0	06-107r1	2007-05-07
Web 3D Service	0.3.0	05-019	2005-02-02
Web Coordinate Transformation Service (WCTS)	0.4.0	07-055r1	2007-10-09
Web Coverage Processing Service (WCPS)	-	06-035r1	2006-05-02
Web Image Classification Service (WICS)	0.3.3	05-017	2005-02-10
Web Object Service Implementation Specification	-	03-013	2003-01-15
Web Services Architecture	0.3.0	03-025	2003-01-18
Web Services Summaries	0.3.0	07-095r2	2007-11-14
WFS Temporal Investigation	0.1.0	06-154	2007-08-14
WMS - Proposed Animation Service Extension	0.9.0	06-045r1	2006-07-27
WMS Change Request: Support for WSDL & SOAP	0.1.0	04-050r1	2005-04-22
WMS Part 2: XML for Requests using HTTP Post	-	02-017r1	2002-08-24
Workflow Descriptions and Lessons Learned	-	06-187r1	2007-05-07
Wrapping OGC HTTP-GET/POST Services with SOAP	0.1.0	07-158	2008-01-02
XML for Image and Map Annotation	0.4.0	01-019	2001-02-06

The following OGC Discussion Papers were approved at the last Technical and Planning Committees in June 2008. These documents are awaiting minor revisions and have not yet been posted to the public site. To request a copy of any of these documents, please email the OGC Technology Desk (techdesk@opengeospatial.org).

Table 3-7. Recently Approved OGC Discussion Papers Relevant to OWS-6

Title	Version	Document #	Date
Loosely Coupled Synchronization of Geographic Databases in the CGDI	0.0.9	08-001	2008-01-02
CGDI WFS and GML Best Practices ER	0.0.9	08-002	2007-12-31
OWS-5 SOAP-WSDL Common ER	0.1.0	08-009r1	2008-01-16

Title	Version	Document #	Date
OWS-5 GeoRM License Broker Specification ER (See Note 1)	0.5	08-076	2008-05-15
OWS-5 GSIP Schema Processing ER	0.0.2	08-077	2008-05-15
OWS-5 Local MSD Profile (see Note 2)	0.0.2	08-078	2008-05-15
OWS-5 CITE Summary ER	1.0.0	08-084	2008-05-20

Notes:

(1) Document 08-076 adoption as an OGC Discussion Paper is contingent on a modification of the document to add sufficient requirements and examples to demonstrate a license as defined by this document is always consistent with figure 5 General License Model, in OGC Document 06-004r3 GeoDRM Reference Model.

(2) Document 08-078 adoption is contingent on completion of Annex A and Annex B.

3.8 Non-OGC Standards Related to OWS-6

These are placed here for reference-specific requirements mentioned in some of the threads. These are not all required or normative.

Table 3-8. Non-OGC Standards Related to OWS-6

Name	Specification	Description
WSDL	Web Services Description Language v 2.0 W3C Recommendation http://www.w3.org/TR/wsdl20/	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.
SOAP	Simple Object Access Protocol (SOAP) 1.1 http://www.w3.org/TR/soap11/ ; SOAP 1.2 http://www.w3.org/TR/soap/	Simple Object Access Protocol (SOAP) is a protocol specification from W3C for exchange of information in a decentralized, distributed environment.
BPEL	Web Services Business Process Execution Language 2.0 – OASIS Standard http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html	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.
ebXML	OASIS Standard 2.0 http://www.oasis-open.org/specs/index.php#ebxmlbp2.0.4 , see also ISO/TS 15000-5:2005	Defines a standards-based business process foundation that promotes the automation and predictable exchange of Business Collaboration definitions using XML.

Name	Specification	Description
ebXML RIM	ebXML Registry Information Model 2.0 – OASIS Standard http://www.oasis-open.org/committees/regrep/documents/2.0/specs/ebrim.pdf	Defines what information is in the Registry and how that information is organized. This leverages as much as possible the work done in the OASIS and the ISO 11179 Registry models.
BPMN	Business Process Modeling Notation 1.1 - OMG Specification http://www.bpmn.org/	Provides businesses with the capability of understanding their internal business procedures in a graphical notation and will give organizations the ability to communicate these procedures in a standard manner.
Wf-XML	Workflow-XML 1.1 and 2.0 - Workflow Management Coalition (WfMC) Standard http://www.wfmc.org/standards/wfxml.htm	Wf-XML is designed and implemented as an extension to the OASIS Asynchronous Service Access Protocol (ASAP). ASAP provides a standardized way that a program can start and monitor a program that might take a long time to complete. Wf-XML provides additional standard web service operations that allow sending and retrieving the "program" or definition of the service which is provided. Wf-XML is an ideal way for a BPM engine to invoke a process in another BPM engine, and to wait for it to completed.
Wf-XML-R	Workflow-XML (RESTful Binding) Draft 0.4 - WfMC Standard http://www.wfmc.org	
XPDL	XML Process Definition Language 2.1 – WfMC Standard http://www.wfmc.org/standards/xpdl.htm	XPDL provides a file format that supports every aspect of the BPMN process definition notation including graphical descriptions of the diagram, as well as executable properties used at run time.
CAP-V1.1	Common Alerting Protocol 1.1 – OASIS Standard http://www.oasis-open.org/committees/download.php/15135/emergency-CAPv1.1-Corrected_DOM.pdf	The Common Alerting Protocol (CAP) is a simple but general format for exchanging all-hazard emergency alerts and public warnings over all kinds of networks.
WS-Addressing	Web Services Addressing – W3C Recommendation, http://www.w3.org/TR/ws-addr-core, and SOAP binding, http://www.w3.org/TR/ws-addr-soap	Web Services Addressing provides transport-neutral mechanisms to address Web services and messages.
WS-Coordination	Web Services Coordination 1.1 – OASIS Standard http://docs.oasis-open.org/ws-tx/wstx-wscoor-1.1-spec/wstx-wscoor-1.1-spec.html	The WS-Coordination specification describes an extensible framework for providing protocols that coordinate the actions of distributed applications.

Name	Specification	Description
WS-Atomic Transaction	Web Services Atomic Transaction 1.1 – OASIS Standard http://docs.oasis-open.org/ws-tx/wstx-wsat-1.1-spec.pdf	This specification provides the definition of the Atomic Transaction coordination type that is to be used with the extensible coordination framework described in WS-Coordination.
WS-Eventing	W3C submission http://www.w3.org/Submission/WS-Eventing/	Describes a protocol that allows Web services to subscribe to or accept subscriptions for event notification messages. This is a public draft for review and evaluation only.
WS-Notification	Web Services Notification 1.3 OASIS Standard http://www.oasis-open.org/committees/documents.php?wg_abbrev=wsn	These specifications provide a standardized way for a Web service, or other entity, to disseminate information to a set of other Web services, without having to have prior knowledge of these other Web Services. They can be thought of as defining "Publish/Subscribe for Web services".
WS-Topic	Web Service Topic – OASIS Standard http://docs.oasis-open.org/wsn/wsn-ws_topics-1.3-spec-os.pdf	Part of WS-Notification
WS-Policy	Web Services Policy – W3C Recommendation http://www.w3.org/TR/ws-policy	The Web Services Policy Framework provides a general purpose model and corresponding syntax to describe the policies of entities in a Web services-based system.
WS-Federation	Identity Federation specification v1.1, http://www.ibm.com/developerworks/library/specification/ws-fed/	Defines mechanisms for allowing disparate security realms to broker information on identities, identity attributes and authentication.
WS-Reliable Messaging	Reliable Messaging 1.1 – OASIS Standard http://docs.oasis-open.org/ws-rx/wsrm/200702/wsrm-1.1-spec-os-01.pdf	Describes a protocol that allows SOAP messages to be delivered reliably between distributed applications in the presence of software component, system, or network failures.
WS-Security	Web Services Security 1.1 – OASIS Standard http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf	This specification and associated token profiles (Username, X.509, SAML, Kerberos, REL, and SOAP with Attachments) provide the technical foundation for implementing security functions such as integrity and confidentiality in messages implementing higher-level Web services applications.

Name	Specification	Description
WS-Trust	Web Services Trust 1.3 – OASIS Standard http://docs.oasis-open.org/ws-sx/ws-trust/v1.3/ws-trust.html	This specification defines extensions that build on [WS-Security] to provide a framework for requesting and issuing security tokens, and to broker trust relationships.
SAML	Security Assertion Markup Language 1.1 – OASIS Standard http://www.oasis-open.org/specs/index.php#samlv1.1 SAML 2.0 – OASIS Standard http://www.oasis-open.org/specs/#samlv2.0	This specification defines the syntax and semantics for XML-encoded assertions about authentication, attributes, and authorization, and for the protocols that convey this information.
XACML	eXtensible Access Control Markup Language 2.0 – OASIS Standard http://www.oasis-open.org/specs/#xacmlv2.0	This specification, together with associated schemas and resource profiles, defines the syntax and semantics for access control.
XML Digital Signature	W3C Recommendation http://www.w3.org/TR/xmldsig-core/	Specifies XML digital signature processing rules and syntax. XML Signatures provide integrity, message authentication, and/or signer authentication services for data of any type, whether located within the XML that includes the signature or elsewhere.
XML Encryption	W3C Recommendation http://www.w3.org/TR/xmlenc-core/	Specifies a process for encrypting data and representing the result in XML. The data may be arbitrary data (including an XML document), an XML element, or XML element content.
PKI	Public Key Infrastructure – IETF Standard http://www.ietf.org/html.charters/pkix-charter.html	Internet standards to support X.509-based Public Key Infrastructures (PKI) for data encryption.
XKMS	XML Key Management System – W3C Note http://www.w3.org/TR/xkms/	Specifies protocols for distributing and registering public keys, suitable for use in conjunction with the proposed standard for XML Signature. This document is a NOTE made available by the W3C for discussion only.
UncertML	Uncertainty Markup Language: http://www.intamap.org/pub/UncertML.pdf http://wiki.intamap.org/index.php/INTAMAP_Wiki	XML vocabulary for communicating about uncertainty.
RSS 2.0	Web syndication system http://www.rssboard.org/rss-specification	RSS is a family of Web feed formats to publish frequently updated content.
Atom 1.0	Atom Syndication Format is IETF RFC 4287 http://tools.ietf.org/html/rfc4287 while Atom Publishing Protocol is IETF RFC 5023 http://tools.ietf.org/html/rfc5023	Alternative to RSS to ease the development of applications with web syndication feeds.

Name	Specification	Description
GeoRSS GML	Geographically Encoded Objects for RSS Feeds as GML Application Schema, http://georss.org/gml	Encoding of GeoRSS' objects in a simple GML version 3.1.1 profile. Compatible with RSS and Atom.
ISO 19117:2005	ISO TC211 Document n1578 http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=40395	Geographic Information - Portrayal
IHO S-52	The International Hydrographic Organization (IHO) Special Publication S-52, Jan 2008 http://www.iho.shom.fr/	International standard for maritime navigation chart symbols used in Electronic Navigation Chart and ECDIS.
ISO/IEC 21000-5: 2004/Amd 2:2007	Rights Expression Language, REL http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44341	ISO/IEC 21000-5:2004 specifies the syntax and semantics of a Rights Expression Language.
ISO/IEC 15408: 2005	Part 1 - http://standards.iso.org/ittf/PubliclyAvailableStandards/c040612_ISO_IEC_15408-1_2005(E).zip ; Part 2 - http://standards.iso.org/ittf/PubliclyAvailableStandards/c040613_ISO_IEC_15408-2_2005(E).zip ; Part 3 - http://standards.iso.org/ittf/PubliclyAvailableStandards/c040614_ISO_IEC_15408-3_2005(E).zip	Information technology – Security techniques – Evaluation criteria for IT security.
ISO/IEC TR15443: 2005	Information technology -- Security techniques -- A framework for IT security assurance http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39733	Technical Report to guide the IT security professional in the selection of an appropriate assurance method when specifying, selecting, or deploying a security service, product, or environmental factor such as an organization or personnel.
ISO/IEC 10181: 1996	ISO catalogue link for ordering: http://www.iso.org/iso/search.htm?qt=10181&published=on&active_tab=standards	Security Framework for Open Systems; Part 1-Overview, Part 2-Authentication framework, Part 3-Access control framework, Part 4-Non-repudiation framework, Part 5-Confidentiality framework, Part 6-Integrity framework, Part 7-Security audit and alarms
ISO 19134	ISO/TC211 N2045, 2006-07-17 – Geographic Information – Location based services – Multimodal routing and navigation	This International Standard provides a conceptual schema for describing the data and services needed to support routing and navigation application for mobile clients who intend to reach a target position using two or more modes of transportation.
INFOD	www.ogf.org	Open Grid Forum (OGF) specification for metadata registry system for use in grid computing.

Name	Specification	Description
CSM TRD	Community Sensor Model (CSM) Technical Requirements Document (TRD) from Community Sensor Model Working Group (CSMWG), http://www.csmwg.seicorp.com/CSM2Doc.htm	The CSM Program will provide Government and Industry with the capability to create and maintain a standard program for developing, testing, and evaluating a collection of current and future sensor models. The models support Sensor Exploitation Tools (SETs) and other application tools that require a precise understanding of the image (data) and ground coordinate relationships. The CSMs are dynamically linked (or loaded) libraries that do not require re-compilation of the SET.

4 OWS-6 Architecture

4.1 Sensor Web Enablement (SWE)

The Sensor Web Enablement (SWE) architecture was designed to enable the creation of web-accessible sensor assets through common interfaces and encodings. Sensor assets may include the sensors themselves, observation archives, simulations, and observation processing algorithms. The role of SWE is depicted in Figure 4-1. The purpose of the OGC Sensor Web Enablement framework is to provide interoperability among disparate sensors and models, as well as to serve as an interoperable bridge between sensors, models and simulations, networks, and decision support tools.

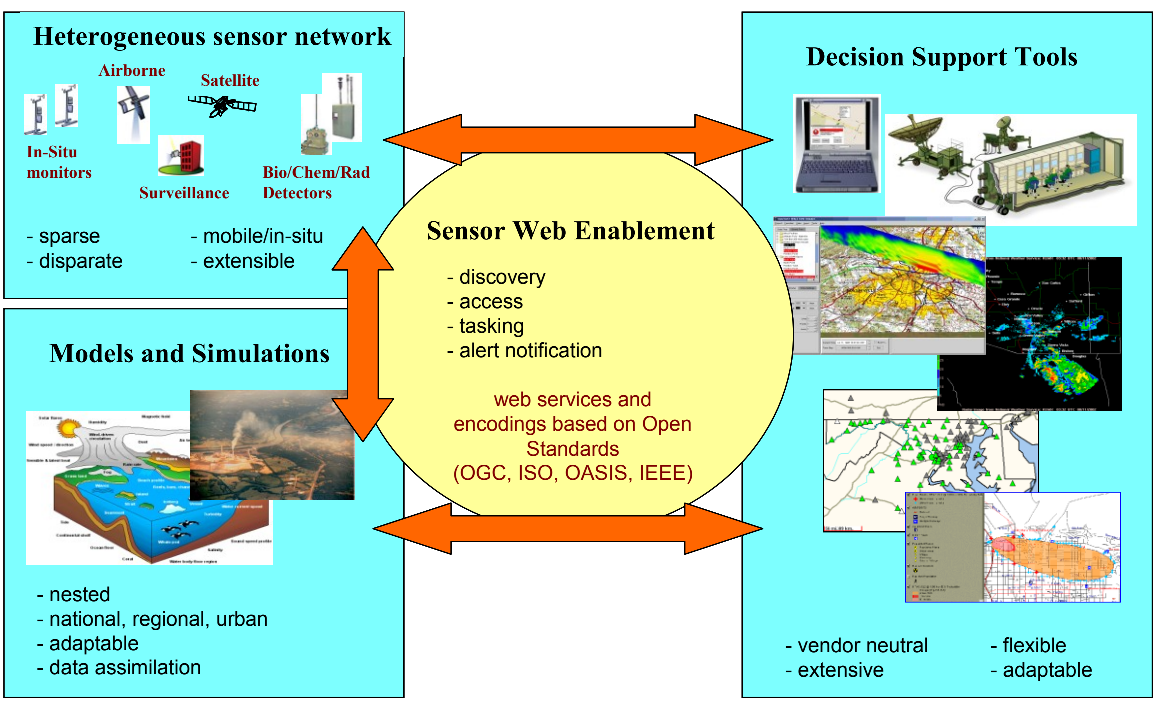


Figure 4-1. The role of SWE

SWE enables the creation of integrated sensor networks where all types of sensors, instruments, imaging devices and repositories of sensor data are discoverable, accessible and, where applicable, controllable via Web technologies and standards. In this vision, connections to sensors are layered with Internet and Web protocols and XML schemas are used to publish formal descriptions of the sensor's capabilities, location and interfaces. Web services for serving, brokering and consuming sensor data can then parse and evaluate sensor characteristics and observations based on their published descriptions. Information provided in XML about a sensor's control interface enables automated communication with the sensor system to determine, for example, its state and location, to issue controlling commands to the sensor platform, and to access its stored or real-time data.

It is expected that the following specifications will be used in this initiative:

- SensorML (OGC 07-000) –v1.0 Implementation Specification
- TransducerML (OGC 06-010r6) –v1.0 Implementation Specification
- Sensor Observation Service (OGC 06-009r5) – v1.0 Implementation Specification
- Sensor Planning Service (OGC 07-014r3) –v1.0 Implementation Specification
- SPS Application Profile for EO Sensors (07-018r2) v0.9.5 – Best Practice Document
- Observations and Measurements - Part 1 - Observation schema (07-022r1) – v1.0 IS
- Observations and Measurements - Part 2 - Sampling Features (07-002r3) – v1.0 IS
- Sensor Alert Service (OGC 06-028r5) – Best Practices Document
- Web Notification Service (OGC 06-095r1) – Best Practices Document
- Sensor Web Enablement Architecture (OGC 06-021r2) - Best Practices Document
- Community Sensor Model (CSM) Technical Requirements Document (TRD) from Community Sensor Model Working Group (CSMWG), http://www.csmwg.seicorp.com/CSM2Doc.htm.

4.1.1 SWE Scope

The OGC Sensor Web Enablement framework has achieved a degree of maturity through previous OWS interoperability initiatives and deployments worldwide. OWS-6 will focus on integrating the SWE interfaces and encodings into cross-thread scenarios and workflows to demonstrate the ability of SWE specifications to support operational needs.

Emphasis for SWE during this phase of the OWS test-bed will be on:

- Apply GeoRM and Trusted Services in SWE environment, including adding SOAP binding to SWE services. Build on OWS-5 SOAP-WSDL Discussion Paper 08-009r1.
- CCSI-Enabled CBRN Sensors into the SWE Environment
- Sensor parameter adjustability and error propagation for georeferenceable imagery. Build on Georeferenceable imagery accomplishments of OWS-5
- Harmonize SWE information models: SensorML, GML, UncertML, MathML
- Events-based architecture including WNS

4.1.2 SWE Requirements

4.1.2.1 Apply GeoRM and Trusted Services in SWE environment

The scope of the SWE thread includes the development of an architecture for GeoRM and Trusted Services in a SWE Environment, taking into account different trust-levels of distinct networks. Building upon previous OGC developments, GeoRM and Trusted Services will be used across several OWS-6 Threads. The discussion of GeoRM and Trusted Services is described once in the RFQ (in the GPW Thread section). Development of an architecture in SWE of these technologies will include SOS, SAS, WNS, SPS and SPS Application Profile for EO Sensors. More specific requirements that the SWE architecture must address include:

- Authentication for establishing secure communication with SWE services and associate access rights to users and clients.
- Secure communication including message layer and network layer security.
- Geo-specific Attribute Based Access Control (ABAC) for SWE services to ensure appropriate functioning and information flow control for requests and responses.
- Trust to SWE services, in particular SPS and SAS to ensure that the alert can be trusted. This feature is very important because severe consequences can be caused by an alert.
- Alerts from SAS/WNS must be delivered to registered (interested) clients. It is important to reach clients in particular behind firewalls! Ensure integrity & confidentiality of messages and insurance of message delivery (reliable messaging).
- Trusted auditing for all communication between secure SWE services.
- Non-repudiation for all communication between secure SWE services.
- Secure SW-Services must advertise the requirements to execute service and secure messages: (i) How must requests be secured to be accepted by SAS? (ii) How will responses be secured?
- Provide SOAP bindings for all SWE services

Topics to be addressed in this architecture study and definition, including consistency with overall OWS security and GeoDRM approaches:

- What are the concrete SWE requirements for “Secure-SWE”?
- Which standards are applicable for securing SWE?
- How to apply existing standards for securing SWE?
- Identification of gaps in existing standards to cover SWE-requirements.
- What is the standardization roadmap for OGC?
- Harmonization with/use of WS-Security standards?

Security and Rights Management are cross-thread topics in OWS-6. Participants should refer to the security requirements, deliverables and architecture described in the GPW sections of this RFQ: Section 4.2.2 GPW Requirements, Section 4.2.3 GPW Deliverables, Section 4.2.4.3 GPW Use Cases, Section 4.2.5.6 Web Services Security and Trust, and Section 4.2.6.2 Security for OGC Web Services, for more details on secure web services for OWS-6.

4.1.2.2 CCSI-Enabled CBRN Sensors into the SWE Environment

In the U.S., the Joint Program Executive Office for Chemical and Biological Defense (JPEO-CBD) has been formed to respond to chemical, biological, radiological and nuclear (CBRN) threats. In January 2008, the JPEO-CBD published the Common CBRN Sensor Interface (CCSI) specification. CCSI is the JPEO-CBD standard for sensor physical and electronic interfaces, including component interconnects, power, external connectors, eXtensible Markup Language (XML) communications, a standard basic command set, and modularity. It will support net-centric communications for all JPEO-CBD sensors and enable host platforms to identify and communicate with any CCSI sensor using common commands and reports.

The CCSI Specification is available from JPEO-CBD:
http://www.jpeocbd.osd.mil/page_manager.asp?pg=4&sub=2

OWS-6 will explore the interoperability of CCSI-enabled CBRN sensors with the OGC SWE standards. The goals for this development include:

- Demonstrate interoperability of CCSI-enabled sensors with OGC SWE standards
- Increase OGC awareness of JPEO-CBD sensor interoperability requirements
- Provide OGC members with opportunities to implement CCSI-oriented services

CCSI-Enabled CBRN Sensors shall be deployed by OWS-6 Participants as a distributed services environment using the SWE standards:

- Deploy a Sensor Observation Service (SOS) that provides access to CCSI-enabled sensor(s)
- Deploy a Sensor Alert Service (SAS) that provides alerts from CCSI-enabled sensor(s)
- Deploy a Sensor Planning Service (SPS) that commands CCSI-enabled sensor(s)
- Register SWE based CCSI-enabled sensors in a catalogue (CSW) using SensorML.
- Deploy a Certification Authority (CA) Service to ensure that any user or application client is authenticated and only authorized to communicate with a specific sensor or set of sensors including via other SWE services.
- The CSW must enable a user / application to obtain a sensor ID based on its type and geospatial data
- The SOS, SAS, and SPS must enable the application to send and receive CCSI commands in addition to SWE command.

The JPEO-CBD will provide the use of CCSI-enabled sensors or sensor emulators to support OWS-6 SWE. The JPEO-CBD Software Support Activity (SSA) will provide technical information and expert advice on the JPEO-CBD CCSI Standard to the industry partners participating in this project.

4.1.2.3 Build on Georeferenceable imagery accomplishments of OWS-5

One objective for OWS-6 SWE is to build upon the work of OWS-5 that focused on enabling a standardized means for users to interactively access 'georeferenceable' imagery data (JPEG 2000 compressed) from a coverage service while supporting the ability to calculate the ground-to-image geolocation relationships using the 'sensor model' parameters associated with georeferenceable imagery. The logical follow-on is to now address a standardized approach for supporting the sensor parameter adjustability and error propagation (e.g. management of standard deviation, variance, and covariance) factors when accessing 'georeferenceable' imagery data interactively in a networked environment. The sponsors require an SOS of this capability for both physical sensor models and replacement (functional fit) sensor models such as those emerging from the Community Sensor Model Working Group (CSMWG) and the renewed ISO 19130 work at TC211.

Participants are to consider the use of UncertML as a candidate solution for communicating error propagation. UncertML has been developed as an XML language for exchanging uncertainty. It has been developed using the GML feature-property model and aspects of SWE Common. Example applications include error characteristics of a sensor observation and interpolation results.

Develop and document, with the intent to ultimately present as an OGC best practice, a fine-grained (sufficient for precise geopositioning, adjustability, and error propagation) profiling solution for imaging remote sensors and processes. UncertML is to be considered for this harmonization. In particular, clarify the roles for use/application of SensorML and TransducerML as they pertain to both physical and replacement sensor models emerging from the CSMWG and TC211 19130 work. Use the knowledge gained through this activity to further the harmonization efforts among SWE Common, Observations and Measurements, SensorML, TransducerML, and other pertinent vocabularies. Evaluate and document recommendations and issues as part of the above Georeferenceable Imagery Engineering Report. consideration/evaluation of geopriv-uncertainty and/or UnCertML for satisfying the above requirements.

It is anticipated that the OWS-5 discussion paper will be available in July 2008: "Supporting Georeferenceable Imagery," OGC Document OGC 08-071.

UncertML References:
http://www.intamap.org/pub/UncertML.pdf
http://wiki.intamap.org/index.php/INTAMAP_Wiki

4.1.2.4 Harmonize SWE information models

OWS-6 will investigate and experiment with candidate solutions for increased synergy between SensorML and GML. The goal is to develop or make use of existing UML models and application schemas to gain a better understanding of interoperability issues between SWE and GML. Consider as part of this work the georeferenceable grid and other grid work in discussion in GMLJP2 SWG, Coverages WG, WCS SWG and GML SWG. Additionally, the contractor shall consider the U.S. National System for Geospatial Intelligence GML application schemas developed under OWS-5 (NAS) and how it might be enhanced to integrate Sensor ML based models with the current GML based models. UncertML should be considered for enhancing the SWE Information Model. The contractor shall document any recommendations and issues in a SWE Information Model Engineering Report.

OWS-6 will investigate and experiment with the possible application of MathML for SensorML method descriptions and execution. SensorML has a vocabulary to describe various sensor-related processes and the method descriptions to perform those processes. For sensor models, these processes and methods often need to include mathematical expressions. The purpose of this investigation is to determine the degree to which other markup languages could (or should) be adopted for use within SensorML rather than defining a SensorML-unique vocabulary for mathematical expressions. Evaluate and document recommendations and issues as part of a SWE Information Model Engineering Report.

In OWS-4, first experiments have been conducted with SWE information models for catalogue entries. OWS-6 will investigate catalogue entry information models that cover both OGC SWE service and sensor instances. The work shall take the important temporal aspect into account, as sensors are often not permanently available.

4.1.2.5 Events-based architecture including WNS

An outcome of the SWE thread of OWS-5 was a recommendation that OWS-6 should develop standards-based approaches Pub-sub, Asynchronous calls, update and Transactional services to be consistent across services. Specific items include:

- More sophisticated handling of events: propagation, filtering
- Further refinement of asynchronous services within SWE
- Application: False alarm reduction by in-network workflow
- Patterns for automated processing based on complex events

This development can be based on Event-based processing. The approach should utilize specifications that have been developed for general information technology. Also the OWS-6 Event Based architecture should be applicable to all OWS activities: OWS, SWE, GPW, DSS, etc. This work should consider past work on OGC Web Notification Service (WNS). This work should consider such specifications as W3C WS-

Addressing, WS-Eventing; OASIS WS-Notification (WS-BaseNotification, WS-BrokeredNotification) and WS-Topics; Atom Publishing Protocol (APP), RSS, OGF INFOD.

4.1.3 SWE Deliverables

The OWS-6 SWE thread requires two types of deliverables:

- **Engineering Reports and Documents**: shall be prepared in accordance with OGC published templates as defined on the OGC portal. Engineering Reports shall be delivered by posting on the OGC Portal Pending Documents list when complete and the document has achieved a satisfactory level of consensus among interested participants, contributors and editors. Engineering Reports are the formal mechanism used to deliver results of the Interoperability Program to sponsors and to the OGC Specification Program Domain and Specification Working Groups for consideration. All documents created in response to this program shall include "OWS-6" in the title, to facilitate later literature searches.
- **Services, Clients and Tools**: shall be provided by methods suitable to its type and stated requirements. For example, services and components (ex. WFS) are delivered by deployment of the service or component for use in the testbed via an accessible URL. A Client software application may be used during the testbed to exercise services and components to test and demonstrate interoperability; however, it is most often not delivered as a license for follow-on usage.

Note that certain draft deliverables will be required by the Interim Milestone at the date shown in the Master Schedule (Main Body, Section 4.6), for use in cross-thread development. These early deliverables will be designated and handled on a thread-by-thread basis.

4.1.3.1 Engineering Reports (ERs) and Documents

The following Engineering Reports (ERs), and Schemas will be developed in the SWE thread and submitted to the OGC Specification Program at the completion of the OWS-6 Testbed.

1) SWE Georeferenceable Imagery ER – Records the evaluation, issues, and recommendations and issues as part of the above georeferenceable imagery activities.
2) SWE Information Model ER - addressing harmonization of the various information models and encoding used in SWE, including SensorML, O&M, OWS Common, GML, etc. New elements for the SWE information model shall also be described including the use of MathML for SensorML process, and proposing how to establish UncertML as a component of SWE. UML models shall be included in the SWE Information Model ER.
3) SWE UML models to support the SWE Information Model ER.
4) SWE CRs – as needed to support new aspects, including SOAP bindings for SWE Services. Needs to reflect OWS-5 SOAP-WSDL Discussion Paper 08-009r1.
5) Secure Sensor Web ER – describing the architecture of GeoRM and Secure Services to the OWS and SWE environments. GeoRM and Secure Services will be addressed as a cross thread topic. Specific application of the technologies to SWE will be addressed as part of the SWE thread.
6) CCSI-SWE Engineering Report - including at least: CCSI-SWE use case; Detailed technical architecture; Technical guidance suitable for procurement; and Lessons learned from OWS-6 developments.
7) Event Architecture ER – This ER will be a cross-thread ER documenting the recommended architecture for event handling in the SWE and more generally OWS environments. ER to describe the

architecture as well as the issues and rationale for recommending the architecture.

4.1.3.2 Services, Clients and Tools

Implementations of the following services, tools and data instances will be developed in this OWS-6 thread, tested in Technology Integration Experiments (TIEs) and invoked for cross-thread scenarios for OWS-6 demonstration events:

1)	SOS for Georeferenceable Imagery - a SOS implementation that ingests and serves JPIP image data, and includes SOAP bindings. This SOS will also be responsible for serving metadata including functional fit parameters as SensorML to facilitate georeferencing and UncertML to assess error propagation.
2)	P*P Service- Authentication Service (PDP), Authorization Service (PIP), Licensing Service (PAP) (cross-thread with GPW)
3)	Gatekeeper(s) for four OWS services (PEP) (cross-thread with GPW)
4)	SOS, SPS, SAS interface to CCSI sensor – components that provide SWE based access to CCSI sensors. The components shall provide SOS, SPS, SAS server interface and will use the JPEO provided CCSI client. Functionally the component provides bridges or translators between SWE and CCSI. The component may need to implement candidate Change Requests to the specifications in order to implement the translator(s) and SOAP bindings. The components will participate in the CCSI use case.
5)	Catalog Service for the Web and associated metadata for SWE – CSW component to participate in the CCSI use case. This deliverable includes developing the catalog information model and instance entries for the CCSI-enabled SWE services. (cross-thread with GPW, DSS)
6)	SWE Client – Provide a SWE client that can interact with the SWE implementations associated with the test-bed. This client will support SOS, SPS, SAS and CSW interfaces at a minimum.
7)	Event service – Event service(s) to support implementation of the Event Architecture
8)	JPIP enabled WCS-T – to access and interactively deliver georeferenceable image data

4.1.4 SWE Enterprise Viewpoint

4.1.4.1 Community and Objectives

The Sensor Web represents a meta-platform that integrates arbitrary sensors and sensor networks; each maintained and operated by individual institutions. This reflects the existing legal, organizational and technical situation. Sensors and sensor systems are operated by various organizations with varying access constraints, security, and data quality and performance requirements. The architectural design of the Sensor Web allows the integration of individual sensors as much as the integration of complete sensor systems without the need of fundamental changes to the constituent systems.

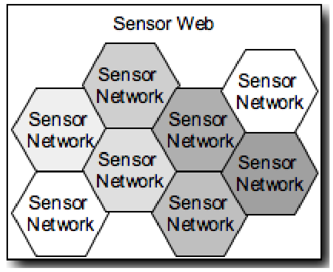


Figure 4-2. Sensor Web: Aggregation of Sensor Networks

Once connected to the Sensor Web, data sets may get used multiple times in applications never intended by the original system setup. Traffic sensors that have been deployed initially to avoid jams by dynamic traffic control might get used to calculate the carbon dioxide ratios of highway sections in another application. Satellites with different sensors on board might get used in a variety of application domains that were not primarily targeted, simply due to interoperable interfaces that allow users to task the satellite based on distinct requirements.

The Sensor Web is a revolutionary concept towards achieving a collaborative, coherent, consistent, and consolidated sensor data collection, fusion and distribution system. It can be viewed as a new breed of Internet for monitoring spatio-temporal phenomena appearing in the physical environment in real time. Any kind of sensor, from a thermometer located at a fixed position to a complex hyper-spectral sensor on board of an earth-orbiting satellite, will be made available on a global level in the near future.

The SWE framework has been designed to enable solutions that meet the following desires:

- Discovery of sensors, observations, and processes
- Determination of a sensor's capabilities and an observation's reliability
- Access to parameters and processes that allow on-demand processing of observations
- Retrieval of real-time or time-series observations in standard encodings –
- Tasking of sensors and simulators to acquire observations of interest
- Subscription to and publishing of alerts based on sensor or simulation observations

4.1.4.2 SWE Use Cases for OWS-6

4.1.4.2.1 Use Case #1: CCSI-enable Sensors in SWE Environment

<table>
<tr><td>Use Case Identifier: SWE #1</td><td colspan="2">Use Case Name: CCSI Sensors in SWE</td></tr>
<tr><td colspan="2">Use Case Domain: OWS-6 SWE</td><td>Status: Draft 2008-7-18</td></tr>
<tr><td colspan="3">Use Case Description: Authenticated and authorized interactions with CCSI-enabled sensors through SWE services.</td></tr>
<tr><td>Actors (Initiators): User of sensor data</td><td colspan="2">Actors (Receivers) Same as initiator</td></tr>
<tr><td>Pre-Conditions:
- User requires transducer data.
- User has authorization to request the collection of the needed data.</td><td colspan="2">Post-Conditions:
Sensor has been collected, processed and is provided to the user for exploitation.</td></tr>
</table>

System Components • SOS, SPS, and SAS that interface to JPEO-CBD-provided CCSI-enabled sensors. • Certification Authority • Catalog Service for the Web • SWE Client with GeoRM and Security enabled
Basic Course of Action: 1. Sensor announces its existence at CSW. Sensor registration contains pointer to metadata registry that contains additional metadata about the sensor 1.1 CSW requests additional metadata about the sensor 1.2 Metadata Registry provides metadata 1.3 CSW requests sensor authentication status from CAS 1.4 CAS provides sensor authentication status 2 User/App requests authorization from CAS to communicate with CSW 2.1 CAS provides credentials 3 User/App requests CSW-records for sensors with identified parameters 3.1 CSW provides sensor metadata and service-URLs that provide access to this sensor 4 User/App requests observations from the sensor at SOS 4.1 SOS connects to sensor and requests readings 4.2 Sensor provides readings 4.3 SOS forwards readings, encoded as observations, back to the User/App in response of the get observations request 5 User/App subscribes to alerts offered by SAS for the sensor

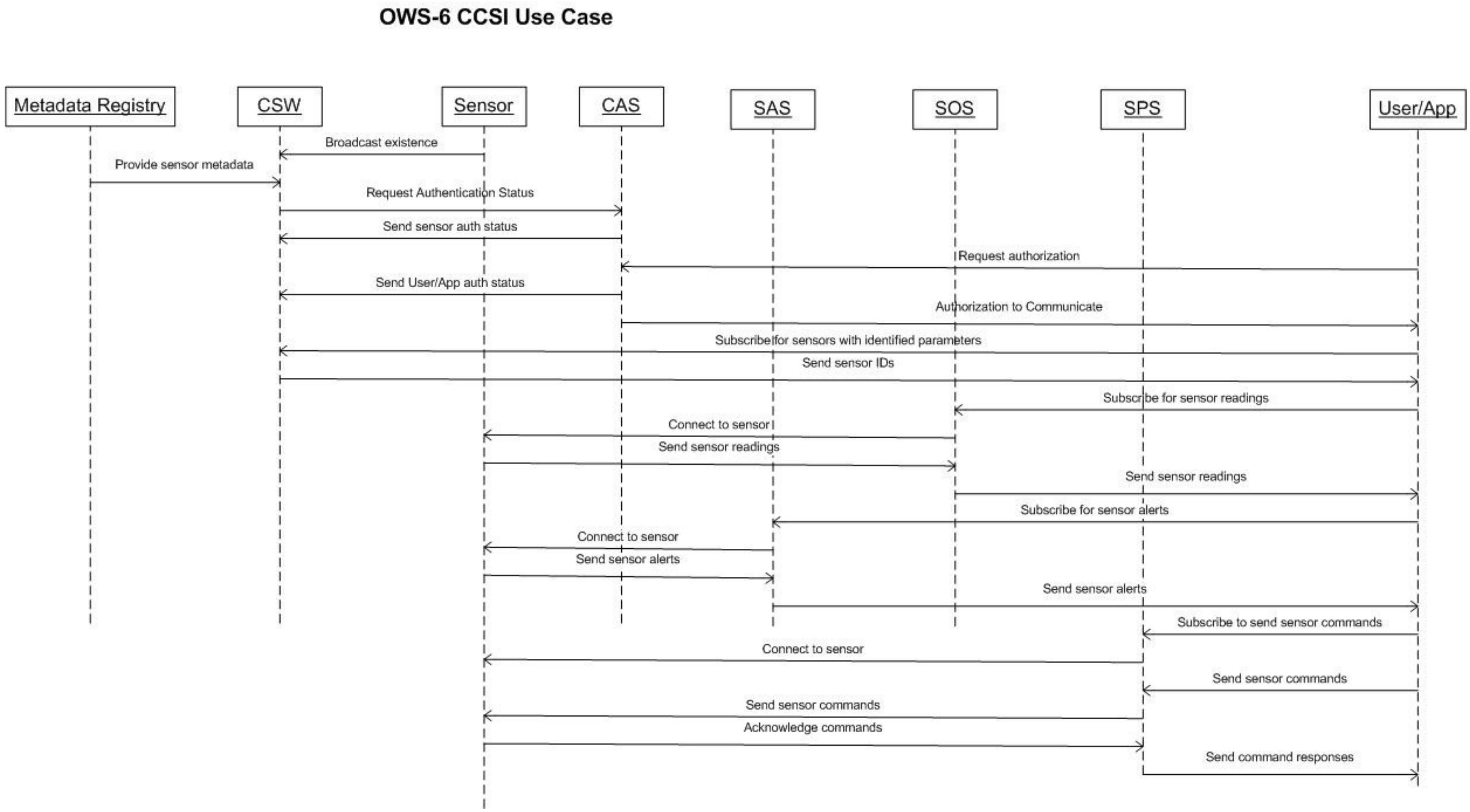


Figure 4-3. CCSI-Enabled Sensors in SWE Use Case (Draft)

4.1.4.2.2 Use Case #2: Georeferenceable Imagery

Catalog search to discover a Sensor Observation Service that can provide JPEG 2000 compressed imagery complete with sensor parameters and/or functional fit parameters over a user defined area falling within a user defined set of temporal parameters. Uncertainty indicators (e.g. management of standard deviation, variance, and covariance) to allow sensor parameter adjustability shall supplement those parameters. Likewise, a catalog search for a JPIP enabled WCS-T that can access and interactively deliver that image data with the sensor/functional fit parameters, adjustable parameters, and error propagation data. Integrated Client then enables the user to interactively select the Area(s) of Interest (at varying resolutions and quality). Once the user has the desired AOI in view, the user saves the compressed data for the AOI and associated parameter information into a self-contained package for exchange that can independently support geopositioning of the selected subset of the source imagery.

<table>
<tr><td>Use Case Identifier: SWE #2</td><td colspan="2">Use Case Name: Georeferenceable Imagery in SWE</td></tr>
<tr><td colspan="2">Use Case Domain: OWS-6 SWE</td><td>Status: Draft 2008-07-18</td></tr>
<tr><td colspan="3">Use Case Description: Access georeferenceable imagery from data streams, extract sensor and/or replacement sensor model parameters along with uncertainty (error propagation) parameters and save area-of-interest data at user-selected resolution.</td></tr>
<tr><td>Actors (Initiators): User of sensor data</td><td colspan="2">Actors (Receivers): Same as initiator</td></tr>
<tr><td>Pre-Conditions:

Data provider offers image data via SOS/JPIP and via WCS-T/JPIP.</td><td colspan="2">Post-Conditions:

Subset of image is stored locally together with sensor and/or replacement sensor model parameters and error support data. Image is stored at user-selected resolution/quality. Geopositioning of the subset image can be performed to the same degree as for the source image for the selected resolution.</td></tr>
<tr><td colspan="3">System Components

- SOS, WCS-T, JPIP server

- SWE Client</td></tr>
<tr><td colspan="3">Basic Course of Action:

1. User requests SOS URLs from catalogue matching identified criteria
a. CSW provides URL of SOS in response
2. User sends get observation request to SOS in order to access georeferenced imagery
a. SOS provides reference to JPIP file as part of the response
3. User connects to the JPIP server to retrieve image stream
a. JPIP server provides image stream in response, containing image data, support data, and sensor and/or sensor replacement model parameters along with uncertainty (error propagation) parameters.
b. SWE Client extracts support data from stream
c. User identifies area of interest
d. User saves the area-of-interest image file at user-selected resolution and stores it locally along with the sensor and/or sensor replacement model, uncertainty parameters and support data.
4. Similar course of action for access and delivery of georeferenceable imagery via WCS-T.</td></tr>
</table>

4.1.4.2.3 Use Case #3: Secure SWE Services

<table>
<tr><td colspan="2">Use Case Identifier: SWE #3</td><td colspan="2">Use Case Name: Secure SWE Services</td></tr>
<tr><td colspan="3">Use Case Domain: OWS-6 SWE</td><td>Status: Draft 2008-07-18</td></tr>
<tr><td colspan="4">Use Case Description: The use case describes the scenario for requesting image acquisition from a space-based sensor using an untrusted network for communication, receiving status notifications and retrieving respective.</td></tr>
<tr><td colspan="2">Actors (Initiators):
- Operator
- Space Operations Centre (SOC)
- Data Centre</td><td colspan="2">Actors (Receivers):

Same as initiator</td></tr>
<tr><td colspan="2">Pre-Conditions:
- The entire communication for execution of the request, receiving messages/notifications and access to observations must be secured.
- Access must be protected to all sensor web services.</td><td colspan="2">Post-Conditions:

Trusted auditing of all communication shall be possible.</td></tr>
<tr><td colspan="4">System Components</td></tr>
<tr><td colspan="4">Basic Course of Action:

This UC starts at when a new acquisition is necessary to provide up-to-date data for a specific area of interest.

Step 1: An operator submits a request to a satellite imaging sensor in a multinational environment via a SPS provided by a SOC.

Step 2: After successful feasibility testing image acquisition is performed.

Step 3: Data is downlinked and stored in the data centre.

Step 4: The operator is notified that data is available.

Step 5: The operator retrieves the data from the repository.</td></tr>
<tr><td colspan="4">Success Guarantee

The use case provides the opportunity to acquire satellite data by trusted procedures in an untrusted environment.</td></tr>
</table>

Additional Information

Different security domains exist secured by Demilitarised Zones. Operator, SOC and Data centres are in different DMZ.

All communication and data transport, e.g. delivery of messages and products, shall be reliable.

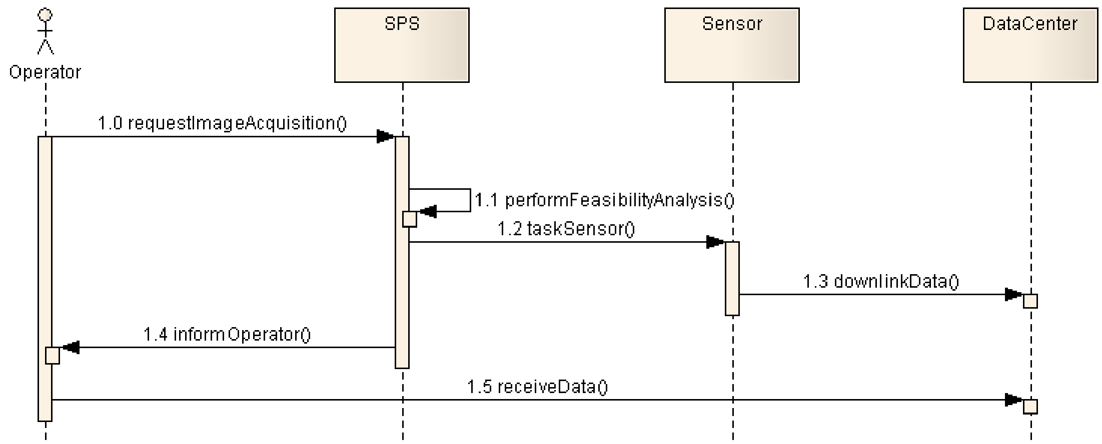


Figure 4-4. Secure SWE use case

4.1.5 SWE Information Viewpoint

The information viewpoint is concerned with the semantics of information and information processing. Thus, it discusses a sensor in regard to the semantics behind a sensor or sensor system abstracted from the physical.

A sensor is a source that produces a value within a well-defined value space of an observed property, which may represent a physical, biological or chemical environmental phenomenon. Sensors and sensor systems as well as simulation models fulfill this definition. If the semantics did not differentiate between produced data based on a physical stimulus or any other data, the limit between model and sensor disappears.

In addition to the observation result, information about the observation procedure, spatial-temporal context, and organizational characteristics have to be provided. Such information is considered to be meta-information for the purpose of interpretation and further processing of the observation results.

4.1.5.1 SWE Common

There are several common core definitions used throughout the SWE framework that have been pulled from other SWE specifications, such as O&M and SensorML, and have been placed within the SWE Common namespace. These are currently not defined within a separate document, but rather are defined within SensorML (mostly) or O&M specification documents. Future releases will separate SWE Common definitions into a separate document. The development of such an SWE Common document isan integral part of the SWE Common Standards Working Group. The first version is expected to be released by the end of 2008.

SWE Common knows a number of fundamental types that derive from the *AbstractDataComponent* (which is derived from *gml:AbstractGMLType*, i.e. it implements abstract-types of the *Geography Markup Language, GML*).

4.1.5.2 Sensor Model Language (SensorML)

SensorML defines models and XML Schema for describing any process, including measurement by a sensor system, as well as post-measurement processing.

Within SensorML, everything including detectors, actuators, filters, and operators are modeled as processes. The type of those processes is either physical or non-physical. The former are called *ProcessModels*, the latter *Components*. Physical processes define hardware assets where information regarding location or interface matters, non-physical processes define merely mathematical operations. The composite pattern allows the composition of complex physical and non-physical processes, called *ProcessChains* and *Systems*. All process types are derived from an *AbstractProcess* that defines the *inputs*, *outputs*, and *parameters* of that process, as well as a collection of metadata useful for discovery and human assistance. The inputs, outputs, and parameters are all defined using SWE Common data types. Process metadata includes identifiers, classifiers, constraints (time, legal, and security), capabilities, characteristics, contacts, and references, in addition to inputs, outputs, parameters, and system location. Further on, it allows modeling the life span of a specific process by defining its history in the form of an event list (e.g. recalibration, adjustments, events, etc.).

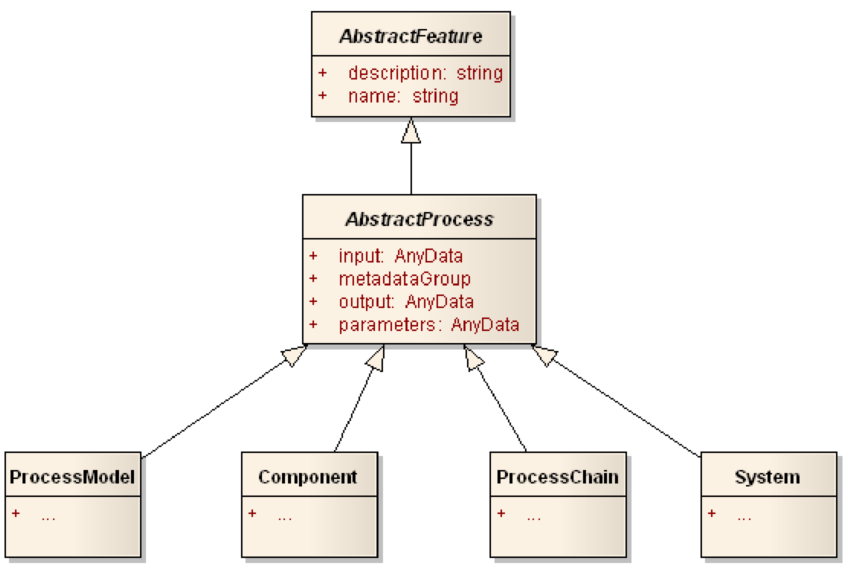


Figure 4-5. SensorML process types

The individual processes, as well as data sources (processes with no inputs), can be linked within a *ProcessChain* such that one can describe either the process by which an observation was derived (i.e. its lineage) or a process by which additional information can be derived from an existing observation. The general idea behind this concept is that one can re-use ProcessChains defined externally as part of the own process chain. Thus, complex chains only have to be defined once and can be re-used when made available online. The definition of links allows the proper lineage of a process chain, i.e. describes the relationships between individual processes of the chain.

System is a physical equivalent of a *ProcessChain.* It allows one to relate one or more processes to the “real world” by allowing one to specify relative locations and data interfaces. A System may include several physical and non-physical processes. In addition to the individual process of a process chain and its relationship to each other in terms of out-input behavior, a System defines the position of each component, i.e. it allows describing one physical asset in relation to another one.

A public forum for SensorML is actively available at
http://lists.opengeospatial.org/mailman/listinfo/sensorml.

4.1.5.3 Observations and Measurements (O&M) [OGC 07-022 & OGC 07-002r3]

To reflect the general difference between the observation itself, i.e. the act of producing a value for a property of a feature, and the sampled feature itself, Observations and Measurements consists of two parts: Part one, *Observation schema* (OGC 07-022), describes a conceptual model and encoding for observations and measurements. This is formalized as an Application Schema, but is applicable across a wide variety of application domains. Part two, *Sampling Features* (OGC 07-002r3), describes a conceptual model and encoding for the feature that has been observed. According to O&M, every observed property belongs to a feature of interest. Though often treated as identical and mostly of little interest to the consumer of the observation data, there is a conceptual difference between the *Sampled Feature* and the *Feature of Interest* of an observation. The difference is described best using some examples:

- In remote sensing campaigns, the sampled feature is a scene or a swath, whereas the feature of interest often defined as a parcel, a region, or any other form of geographically bounded area
- In-situ observation campaigns may obtain the geology of a region at outcrops (sampled features), whereas the feature of interest is the region itself
- Meteorological parameters might get sampled at a station, whereas the feature of interest is - strictly spoken - the world in the vicinity of that station

The term *Measurements* in *Observation & Measurements* reflects the fact that most sensors produce estimates for physical quantities, i.e. for measures. Thus, a measurement is a specialized observation. This is somewhat is contrast to the conventional measurement theory, but inline with discussions in recent publications.

O&M defines an *Observation* as an act of observing a property or phenomenon, with the goal of producing an estimate of the value of the property. The observation is modeled as a Feature within the context of the General Feature Model [ISO 19101, ISO 19109]. An observation feature binds a result to a feature of interest, upon which the observation was made. The observed property is a property of the feature of interest, as illustrated in the following figure.

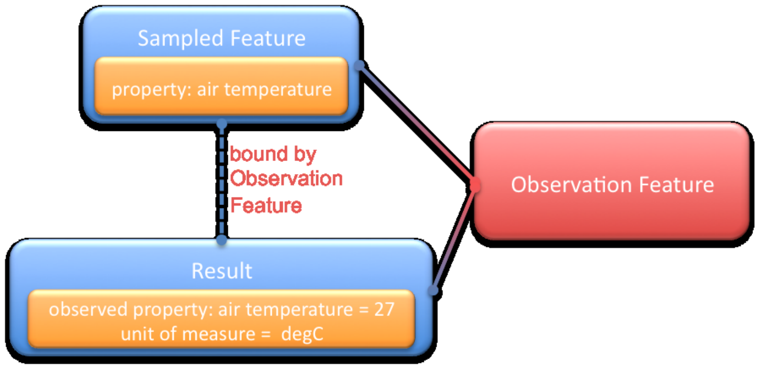


Figure 4-6. Binding of observation results to properties of the sampled feature

An observation uses a procedure to determine the value of the result, which may involve a sensor or observer, analytical procedure, simulation or other numerical process. This procedure would typically be described as a process within SensorML. The observation pattern and feature is primarily useful for capturing metadata associated with the estimation of feature properties, which is important particularly when error in this estimate is of interest.

Through the O&M specification, the SWE framework provides a standard XML-based package for returning observation results. Using a standard package in which to download observations from an SOS alleviates the need to support a wide range of sensor-specific and community-specific data formats. To achieve an even higher level of interoperability, SWE Common shall be used to fill in the result slot of an observation.

4.1.5.4 Transducer Model Language (TML) [OGC 06-010r6]

Transducer Markup Language (TML) is a method and message format for describing information about transducers and transducer systems and capturing, exchanging, and archiving live, historical and future data received and produced by them. A transducer is a superset of sensors and actuators. TML provides a mechanism to efficiently and effectively capture, transport and archive transducer data, in a common form, regardless of the original source. Having a common data language for transducers enables a TML process and control system to exchange command (control data) and status (sensor data) information with a transducer system incorporating TML technology. TML utilizes XML for the capture and exchange of data.

TML was designed with the express goal of facilitating the development of a "Common" Transducer Processing/Control machine while also facilitating interoperable machine-to-machine communications. For the purposes of data fusion and post analysis, it is paramount to preserve raw transducer data in as close a manner to the original form as possible. Although data would be ideally preserved in its raw format, it is impossible in some cases to do so. TML provides facilities to capture data at any stage, from raw production, to partially processed, to final data forms. Greater benefits of TML are realized the closer to the source raw data one gets.

Transducer Markup Language (TML) defines:

- a set of models describing the hardware response characteristics of a transducer.
- an efficient method for transporting sensor data and preparing it for fusion through spatial and temporal associations

4.1.5.5 GeoReferenceable Imagery

The term "image" has many meanings beyond those of geographic imagery. Geographic imagery is imagery whose data is associated with a location relative to the Earth. For example, to view geographic imagery, a presentation process is required. ISO 19101-2 defines Geographic Imagery Scene as imagery associated with a location relative to the Earth whose data consists of measurements or simulated measurements of the natural world produced relative to a specified vantage point and at a specified time. A geographic imagery scene may also be considered as an "observation coverage" which is more general and appropriate for describing the output of a remote sensor. Georeferenceable imagery is a type of geographic imagery scene.

Georeferenceable imagery is typically imagery coming from a remote sensor, which has not previously been georectified, resampled, or regridded. Georeferenceable imagery is expected to be accompanied with information sufficient to allow ground-to-image positioning of selected points, triangulation and georectification of the imagery with error management upon its receipt. This is in contrast to *in situ* sensors where the location of the observed phenomenon is considered to be a point that is coincident with the location of the sensor itself.

In order to determine the precise location of the objects of interest with a given remotely sensed scene one must utilize models (typically referred to as "sensor models"), which provide a mapping of positions in the image to positions within the environment. The design of the collection system and its relationship to the detector and platform on which it is attached, as well as the dynamics of the system, generally dictate a rigorous physical model for this mapping.

Many of these remote sensors generate a spatial-temporal coverage of observation points that can be regularly arranged as a grid and viewed as an image. Within this text, the term *image* will generally be considered synonymous with a regularly gridded collection or coverage of observations coming from a remote sensor.

While georeferenceable imagery typically exists within a regular grid, this grid is not regular within any geodetic coordinate system. That is, the imagery is usually obliquely oriented to geodetic systems (e.g. the latitude-longitude grid) and one cannot expect the spacing of pixels in the imagery to be equal distance; nor can one expect the size of the area covered by each pixel to be equal in area.

All geometric and temporal characteristics of a sensor system must be related to a specified coordinate reference system (CRS). Typically, definitions for sample geometry, look angle, and collection geometry are often described relative to the sensor's CRS. In such cases, it is only through the sensor's relationship to its mount and platform(s), that the sensor and its measurements can be related to an external CRS, such as geographic latitude and longitude.

The relationship between CRSs can be accomplished either by describing a transform process between the coordinate reference systems or by defining the state of the object relative to a CRS.

For a remote sensor, it is necessary to determine the intersection of a pixel's look ray and the surface of the sensor's target (e.g., the Earth's ellipsoid). Typically the look angle and the sensors target are transformed into a common spatial reference frame, such as the Earth-Centered Earth-Fixed (ECEF) or Earth Centered Inertial (ECI) reference system. For in-situ sensors, the process is typically much easier.

4.1.5.6 Sensor Information Models for OWS-6

One objective for the proposed OWS-6 effort is to build upon the work of OWS-5 that focused on enabling a standardized means for users to interactively access 'georeferenceable' imagery data (JPEG 2000 compressed) from a coverage service while supporting the ability to calculate the ground-to-image geolocation relationships using the 'sensor model' parameters associated with georeferenceable imagery. The logical follow-on is to now address a standardized approach for supporting the sensor parameter adjustability and error propagation (e.g. management of standard deviation, variance, and covariance) factors when accessing 'georeferenceable' imagery data interactively in a networked environment. In OWS-6 this capability shall be demonstrated using Sensor Observation Service implementation for both physical sensor models and replacement (functional fit) sensor models such as those emerging from the Community Sensor Model Working Group (CSMWG) and the renewed ISO 19130 work at TC211 is required.

OWS-6 will address a standardized approach for supporting sensor parameters, sensor parameter adjustability, and error propagation (e.g. management of standard deviation, variance, and covariance) factors when accessing 'georeferenceable' imagery data interactively in a networked environment. Towards establishing this capability for both physical sensor models and replacement (functional fit) sensor models, OWS-6 will investigate approaches such as those emerging from the Community Sensor Model Working Group (CSMWG), the renewed ISO 19130 work at TC211, and application/expansion of OGC specifications for SensorML and TransducerML.

OWS-6 will develop and document, with the intent to ultimately present as an OGC best practice, a fine-grained (sufficient for precise geopositioning, adjustability, and error propagation) profiling solution for imaging remote sensors and processes. In particular, clarify the roles for use/application of SensorML and TransducerML as they pertain to both physical and replacement sensor models emerging from the CSMWG and TC211 19130 work. Use the knowledge gained through this activity to further the harmonization efforts among SWE Common, Observations, SensorML, TransducerML, and other pertinent vocabularies.

In addition to the primary SWE Information standards listed above, OWS-6 will include working with these specifications:

- GML
- GMLJP2
- Community Sensor Model Working Group (CSMWG)
- ISO 19130 work at TC211.
- UncertML
- MathML

OWS-6 will investigate and experiment with candidate solutions for increased synergy between SensorML and GML. Develop or make use of existing UML models and application schemas to gain a better understanding of interoperability issues between SWE and GML. GML considerations should in particular take into account the GML application schema work done in OWS5 especially with respect to the U.S. National System for Geospatial Intelligence. Conduct this work in line with the georeferenceable grid and other grid work in discussion in GMLJP2 SWG, Coverages WG and GML SWG. The NSG Application Schema (NAS) shall be considered and how it might be enhanced to begin to integrate Sensor ML based models in addition to the current GML based models.

OWS-6 will consider UncertML as a candidate solution for communicating error propagation. UncertML has been developed as an XML language for exchanging uncertainty. It has been developed using the GML feature-property model and aspects of SWE Common. Example applications include error characteristics of a sensor observation and interpolation results. OWS-6 will investigate and experiment with the possible application of MathML for SensorML method descriptions and execution. SensorML has a vocabulary to describe various sensor-related processes and the method descriptions to perform those processes. For sensor models, these processes and methods often need to include mathematical expressions. The purpose of this investigation is to determine the degree to which MathML could (or should) be adopted for use within SensorML rather than defining a SensorML-unique vocabulary for mathematical expressions.

The contractor shall document any recommendations and issues in a SWE Information Model Engineering Report.

4.1.6 SWE Computational Viewpoint

The computational viewpoint is concerned with the functional decomposition of the system into a set of services that interact at interfaces.

4.1.6.1 Sensor Observation Service (SOS) [OGC 06-009]

An SOS organizes collections of related sensor system observations into Observation Offerings. The concept of an Observation Offering is equivalent to that of a sensor constellation. An Observation Offering is also analogous to a "layer" in Web Map Service because each offering is intended to be a non-overlapping group of related observations.

4.1.6.2 Sensor Alert Service (SAS) [OGC 06-028r6]

The Sensor Alert Service (SAS) can be compared with an event stream processor in combination with an event notification system. An SAS processes incoming sensor data continuously. Pattern-matching algorithms identify satisfied alert conditions and start the alert distribution process.

An SAS might therefore provide a wide variety of alerts related to sensors and sensor observations including, as examples, measured values above a threshold, detected motion or the presence of a recognizable feature, or perhaps sensor status (e.g. low battery, shutdown or startup). SAS is a transactional service in two counts: First sensors can dynamically connect to the service and publish metadata and observation data (using SensorML and O&M). Sensors don't have any knowledge about current subscriptions, as SAS doesn't provide such information. Second, event consumers may define their own event conditions. For every new event type definition, the SAS is opening a new distribution channel and incorporates the alert type in its capabilities documentation, i.e. every new alert type, defined by an alert consumer, is available to other users as well.

An SAS can *advertise* what alerts it can provide. A consumer (interested party) may *subscribe* to alerts disseminated by the SAS. If an event occurs the SAS will *publish* an alert and *notify* all clients subscribed to this event type through a messaging service. Currently, SAS supports XMPP (Extensible Messaging and Presence Protocol) for alert distribution exclusively, although in combination with a Web Notification Service (WNS), it may deliver alerts to other communication endpoints as well. This pattern is likely to change in future versions of the SAS. Currently, the SAS editors are busy doing research on alternative alert and notification mechanisms and protocols.

4.1.6.3 Sensor Planning Service (SPS) [OGC 07-014r3]

The Sensor Planning Service (SPS) is intended to provide a standard interface to collection assets (i.e., sensors, and other information gathering assets) and to the support systems that surround them. Not only must different kinds of assets with differing capabilities be supported, but also different kinds of request processing systems, which may or may not provide access to the different stages of planning, scheduling, tasking, collection, processing, archiving, and distribution of requests and the resulting observation data and information that is the result of the requests. The SPS is designed to be flexible enough to handle such a wide variety of configurations. SPS uses SWECommon to describe planning parameters that have to be set by users. SPS is often used together with WNS and SOS.

4.1.6.4 Web Notification Service (WNS) [OGC 06-095r1]

Web Service environments provide a suitable method to gather requested information in an appropriate way. Synchronous transport protocols such as HTTP provide the necessary functionalities to post requests and to receive the respective responses. HTTP is a reliable protocol in the way it ensures the packet delivery, in order, and with a definitive acknowledgement for each delivery or failure. In case of a simple Web Map Service, a user will receive visualized geographic information after a negligible amount of time, or the user will receive an exception message. HTTP satisfies the needs for this kind of processing, without the need for further functionality.

As services become more complex, basic request-response mechanisms need to contend with delays/failures. For example, mid-term or long-term (trans-) actions demand functions to support asynchronous communications between a user and the corresponding service, or between two services, respectively. A Web Notification Service (WNS) is required to fulfill these needs within the SWE framework and should be considered for more general application in the OWS framework.

The Web Notification Service Model includes two different kinds of notifications. First, the "one-way-communication" provides the user with information without expecting a response. Second, the "two-way-communication" provides the user with information and expects some kind of asynchronous response. This differentiation implies the differences between simple and sophisticated WNS. A simple WNS provides the capability to notify a user and/or service that a specific event occurred. In addition, the latter is able to receive a response from the user.

4.1.6.5 Sensor Web Registry

The Sensor Web registry was implemented successfully using an OGC Catalog Service backed up by an ebXML RIM engine during OWS-4 in 2006. This service provides discovery capability throughout the whole sensor web infrastructure. Typical requests to this service are 'GetRecords' operations containing filtering parameters used to search a database for one or more matching objects of interest. These objects include SWE services (as well as other OGC services), sensor descriptions, process chains and dictionary entries such as phenomena or units, etc.

In order to be able to insert objects to a catalogue, each object type must be defined by a schema and a CSW harvest profile. This profile shall define what information needs to be parsed out of the object XML and advertised as searchable content.

4.1.6.6 SWE Service Interactions

The SWE Services and Encodings interactions are illustrated in the following figures. In the upper right corner, shows sensors that are registered at a SOS and publish observation results to the services. To be discoverable, sensors, using SensorML, and SOS register at a catalog service. The user in the lower right corner requires observation data and sends therefore a *search request* to the catalog. The catalog responses with a list of SOS service instances that fulfill the requirements. Eventually, the user binds the SOS and retrieves the observation data, encoded in O&M.

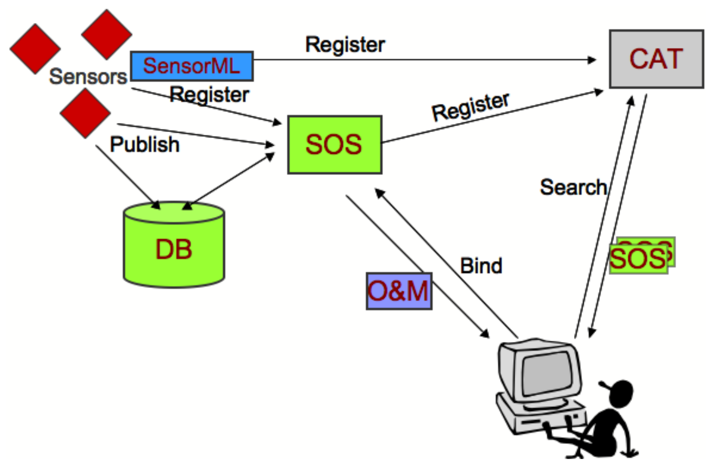


Figure 4-7. SWE Services and Encodings Interactions, part 1

The situation gets a bit more complex in the following. Let's assume that the catalog didn't provide any SOS instances that fulfill the requirements set by the user. In this case, the user may search for Sensor Planning Services that could task sensors to perform appropriate actions in order to produce the observation data our user is looking for. The catalog provides the link to the SPS instance and the user assigns the task. The SPS forwards the command to the sensor. The communication between the SPS and the sensor is opaque for the user. If we imagine a satellite that has to reorient its infrared cameras and has to reach its target position in space, the tasking might take a while. Once observing the requested scene, the sensor dumps its data into a database that is linked to a SOS. The SPS informs the user about data availability using a Web Notification Service. This has the advantage that the SPS can respond to the tasking request right away and has a mechanism to reach the user at a later stage, e.g. if the data is available or if the tasking is delayed or cancelled. The notification message contains all necessary information to access the data from a SOS.

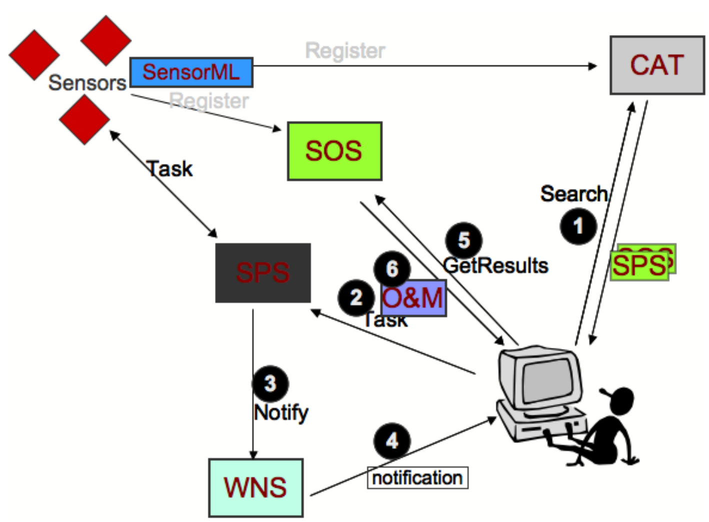


Figure 4-8. SWE Services and Encodings interactions, part 2

Often, we are faced with the situation that a client is not interested in all observation results of a sensor, but wants to get notified immediately if a specific situation is observed. Figure 4-9 illustrates this scenario.

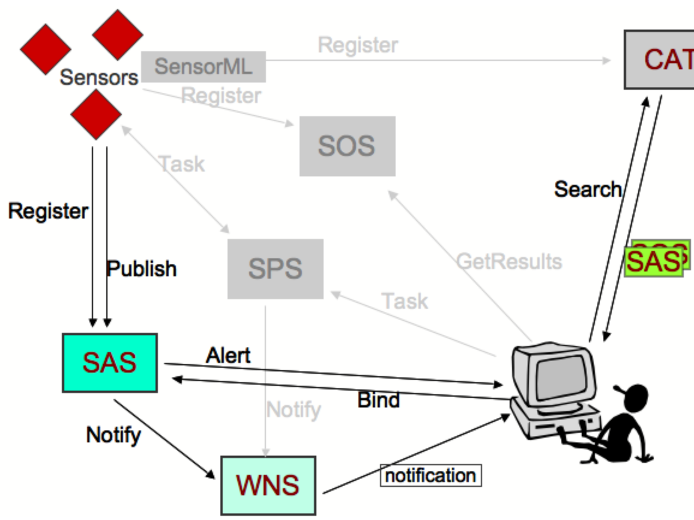


Figure 4-9. SWE Services and Encodings interactions, part 3

Once again, the client receives information about appropriate SAS from a catalog and subscribes to the SAS. Sensors publish observation results continuously to the SAS. The SAS handles all the filtering and alerts the client if the subscription condition is matched. The SAS either sends the alert directly to the client, or makes use of the WNS in order to deliver the alert message.

For detailed UML Sequence Diagrams for several scenarios, see Sensor Web Enablement Architecture (OGC 06-021r2) - Best Practices Document

4.1.6.7 CCSI-Enable Sensor Developments in OWS-6 SWE

During OWS-6, the services of the computational viewpoint will be enhanced with bridges between SWE and CCSI-Enabled Sensors.

The CCSI Specification Volume I provides a Summary and Architecture (http://www.jpeocbd.osd.mil/page_manager.asp?pg=4&sub=2) including an annex that provides the DODAF architecture products. The DODAF Systems Functionality Description (SV-4) is an architecture composite that illustrates “functions” (system functions) and “data” (details of Systems Data Exchanges) primitives of a given architecture at the Designer and Builder levels of detail. The SV-4 Systems Functionality Description functionally decomposes the CCSI system function in the into five discrete system functions:

- P1 – Broadcast sensor existence
- P2 – Send registration data
- P3 – Provide current state
- P4 – Sensor performs operations
- P5 – Sensor terminates operation.

In OWS-6, analysis of the CCSI system functions with respect to the SWE service operations will be a precursor implementing bridges between SWE services and CCSI-enabled sensors.

4.1.6.8 Security and GeoRM for SWE Services

The OGC Architecture for GeoRM and Secure web services is described in the GPW thread section of this document. The architecture for GeoRM and Secure web services will be deployed to secure SWE services in OWS-6.

4.1.6.9 OWS-6 Event Driven Architecture for SWE

Basically, an event-based interaction model relies on two concepts (Muehl/Fiege/Pietzuch, 2006):

- An *event*, that describes any happening of interest that can be observed from within a computer, and
- A *notification*, that reifies an event as a datum, i.e., it contains data describing the event.

The "happening of interest" in the event definition indicates that an event requires an event condition that has to be stated before an event may occur. This is a more restricted definition than provided by (Muehl/Fiege/Pietzuch, 2006), who further define an event as "an arbitrary detectable state change in a computer system". Notification as defined by (Muehl/Fiege/Pietzuch, 2006) is sometimes referred to as “alert” (OMG, 2008). The terms notification and alert are used synonymously.

In the event-based interaction model the initiator is the provider of the data, i.e. the producer of notifications. The addressing scheme is indirect, which means that the notifications are not addressed to any specific set of recipients but mediated by a broker component, which offers a notification service. A consumer may express its interest to notifications by subscribing to the notification service. The essential characteristic of this interaction model is that producers do not know any consumers.

Notifications are conveyed via messages on the network transport level. Figure 4-10 illustrates the distinction between events, notifications, and messages. The model clearly separates the underlying communication technique from the model of interaction.

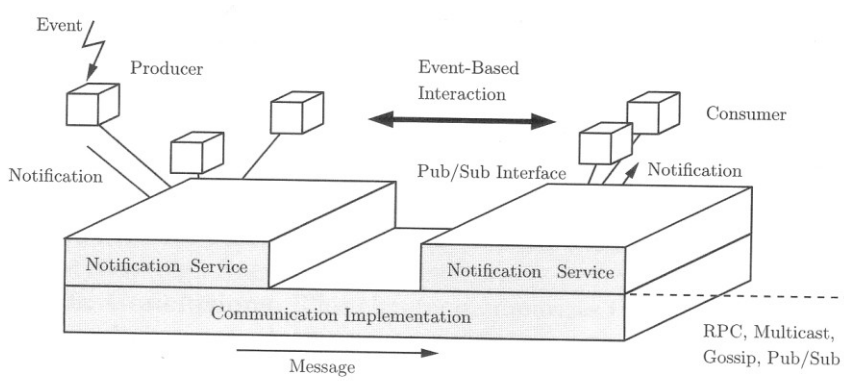


Figure 4-10. Event-based System (Muehl/Fiege/Pietzuch, 2006)

OWS architectures in general require means and mechanisms to define, generate, distribute, receive, and process events. Three types of events are observed most frequently:

1. Events based on observations made by sensors or models,
2. Events based on aggregated observations made by one or multiple sensors or models, and
3. Events related to the operation of the sensor network or the sensor services.

The first type of events occurs if a sensor detects something that matches a previously defined event condition. The occurrence may take place in the environment of the sensor (externally) or internally. Examples of both event types are temperature values that exceed a threshold, the detection of hotspot pixels in a remote sensing image (both externally), or low battery power of a sensor (internally).

The second type of events occurs if non-atomic conditions occur, e.g. temperature and wind speed observation result values exceed thresholds in the case of a multiple sensor event. Additionally, events may be based on conditions that remain for a well-defined number of time intervals, e.g. temperature exceeds a threshold for n time intervals continuously (also known as time series analysis based events).

The third type of events occurs when some state has changed in the sensor or service network configuration (e.g. addition of a new sensor device or sensor service instance) or some unforeseen behaviour has been detected. The latter situation usually results in an exception on programmatic level. If deemed essential by the programmer, such exceptions may be escalated to other components in form of events.

Often all event types are handled equally and the event type is transparent to the receiver of an event notification. It is created by the event observer and published or transferred to notification consumers.

4.1.7 SWE Engineering Viewpoint

4.1.7.1 Deploying SWE as components

The Engineering Viewpoint identifies component types to support distributed interaction between the components. Component is defined using the information and services defined in the previous viewpoints. Assignment of information components and services to component types is optimized for generalized deployment environments and communication types. Communication for services is typically a network like the Internet.

Sensor systems might be either connecting a single sensor or a whole sensor network to the communication network. Further on, a sensor system might even integrate all necessary components to act as one single network node, i.e. the sensor system is addressable and accessible within the communication network.

4.1.7.2 Georeferenceable Deployment Architecture

The OWS-5 Testbed developed approaches for applying SWE and OWS to GeoReferenceable imagery, i.e., imagery that is unprocessed and has not been gridded or geolocated. The primary objective of this activity was to establish a standardized means to allow the user to interactively access a subset pixels from a coverage service stored in the compressed domain (JPEG2000) and preserve the image relationship with the associated 'sensor model' parameters such that precise geopositioning capabilities can be realized in a dynamic, interactive, networked environment.

Figure 4-11 illustrates this effort utilizing a JPIP enabled SOS that provides both SensorML and JPIP encoding streaming responses. A transactional WCS provides an interface that supports the creation of coverages. Users may then easily discover and access this data using the WCS interface. The primary use case involved having a user request subsets of image data. The workflow then uses the functional fit parameters in the metadata to georeference the image segment. The OWS services included in these workflows include: SOS, SPS, WCS-T, WPS, CS/W, SAS. The logical follow-on in OWS-6 is to now address a standardized approach for supporting the sensor parameter adjustability and error propagation (e.g. management of standard deviation, variance, and covariance) factors when accessing 'georeferenceable' imagery data interactively in a networked environment using SOS.

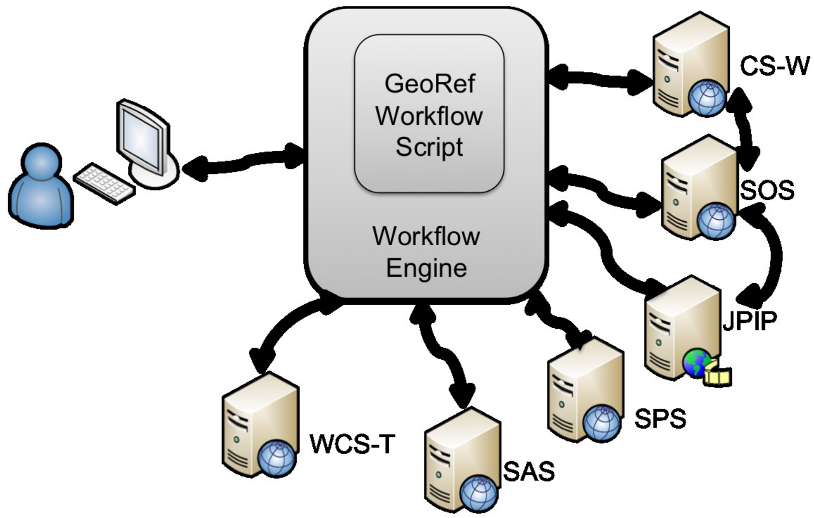


Figure 4-11. Exemplary SWE GeoRef Workflow Wiring Diagram

Figure 4-12 and Figure 4-13 show the variation in types of services which can be handled by thin (browser-based) and thick clients.

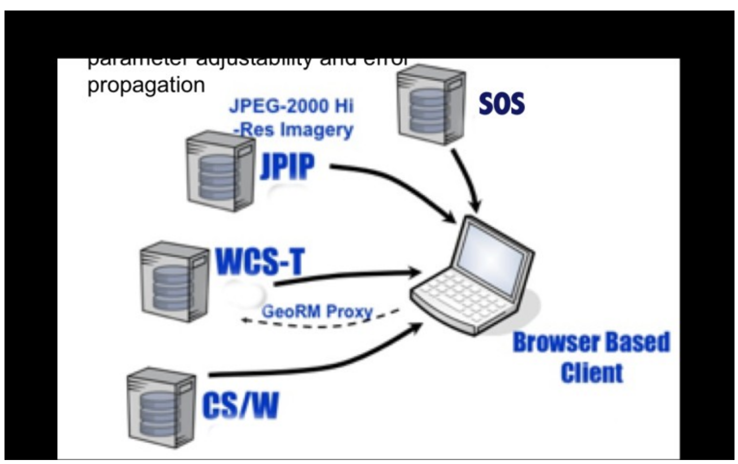


Figure 4-12. Exemplary SWE GeoRef using Browser Based Client

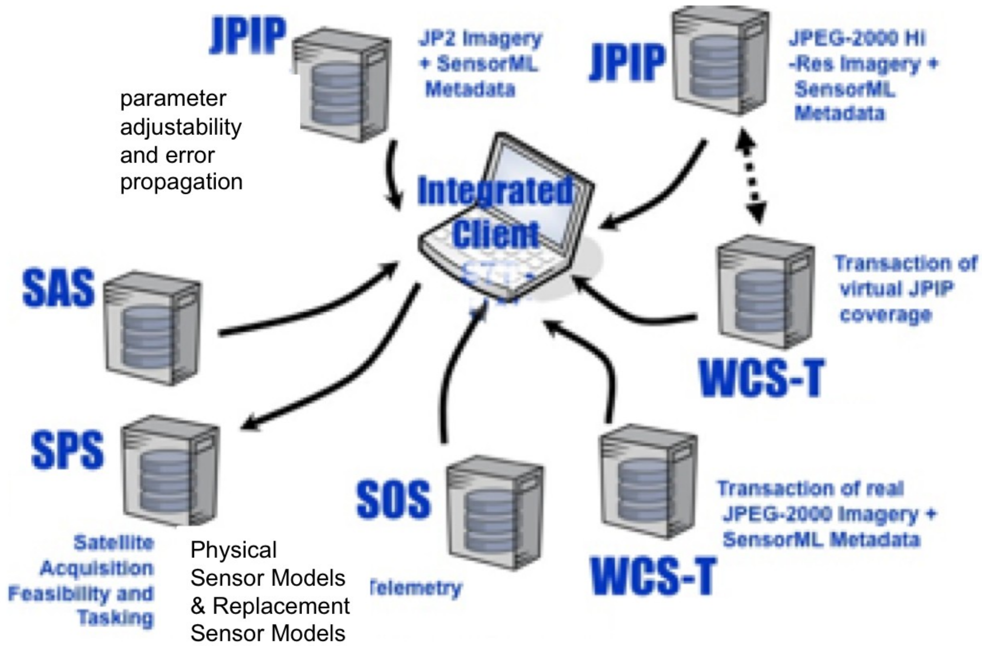


Figure 4-13. Exemplary SWE GeoRef using Rich Client

4.1.7.3 Using SWE services in a "star" network configuration

The following diagrams show the use of SOS, SAS, SPS and WNS within a specific implementation of the SWE framework inspired by Oak Ridge National Labs SensorNet. This configuration involves multiple simple nodes connected through SWE services to a central data center with much greater processing power. Sensors connected to each node are IEEE-1451 sensors that have plug n' play capability. The following diagram illustrates the configuration of each node on the network.

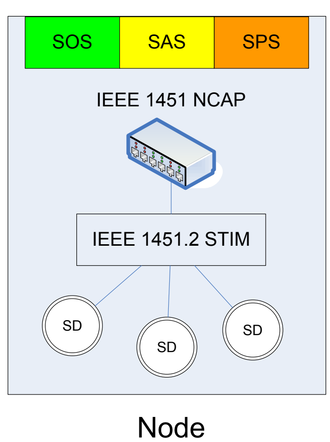


Figure 4-14. Network Nodes

Each node provides connectivity with other network components through optional SWE interfaces. Implementations of these interfaces are expected to be very light weight since the available processing power and storage capacity on each node is very low.

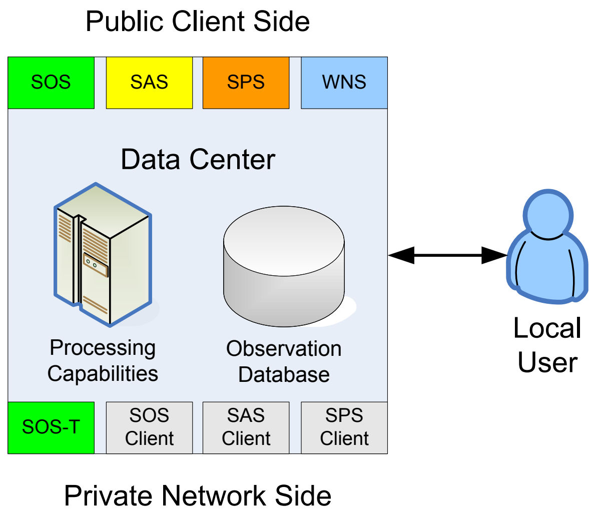


Figure 4-15. Network Interfaces

In this example, each node can either incorporate a small SOS providing access to real time streaming data (if network connection is permanent) or simply inserts its observation in the data center periodically (using if network connection is intermittent, e.g. GSM). The SPS interface is only available on nodes that can achieve simple taskable operations like collecting data at a specific time or changing sensor parameters. The SAS is only available on nodes where a processing of the data is possible and used to automatically detect certain alarming conditions that will be reported to the data center. A simple WNS message generator (not the full service) is also available in order to send alert message or SPS notifications. The following diagram shows the configuration used for the data center, which is also used as a mission center (with SPS).

The data center provides many more features than a single node. It consists of a public interface accessible from the Internet and providing SOS, SAS, SPS and WNS services and a private interface only accessible from inside the sensor network and locally. The following diagram shows a global view of the network.

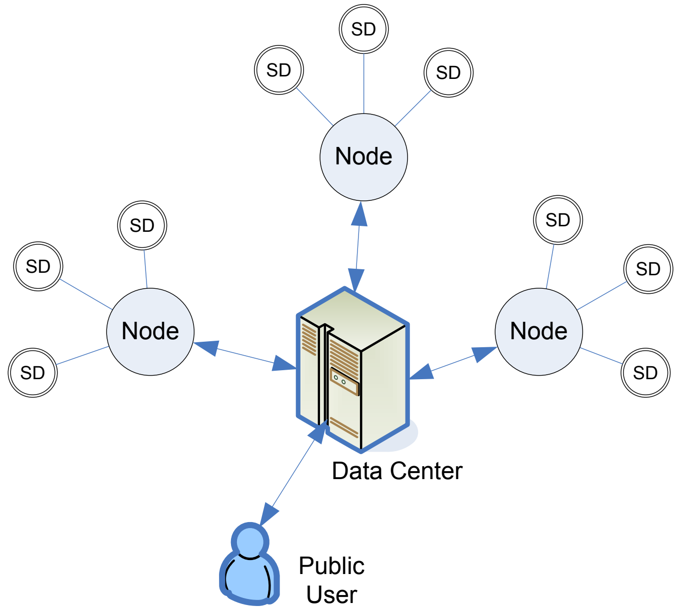


Figure 4-16. Global View of the Network

An unlimited number of nodes can of course be connected to the data center in this fashion, each of which reports data from multiple sensors. A public user doesn't have direct access to the nodes since the datacenter acts as a gateway. The intent with this kind of architecture is that the data center can reflect all individual nodes capabilities in one or more aggregate capabilities documents that are accessible to the public and this for all SWE services. The data center can also filter this information if some of it should not be made available to the public and eventually derive new products from raw sensor data that can be added to the global capabilities of the network.

Further information about Using SWE services in a "star" network configuration can be found in the Sensor Web Enablement Architecture (OGC 06-021r2) - Best Practices Document

4.1.7.4 CCSI-enable Sensors in SWE Components

The deployment architecture for CCSI-Enable Sensors in SWE environment will focus heavily on "bridges" between SWE and CCSI. An aim will be to simplify the bridges by recommending changes to the CCSI and SWE specifications to align the interaction and information elements, thereby reducing the processing required in the bridges.

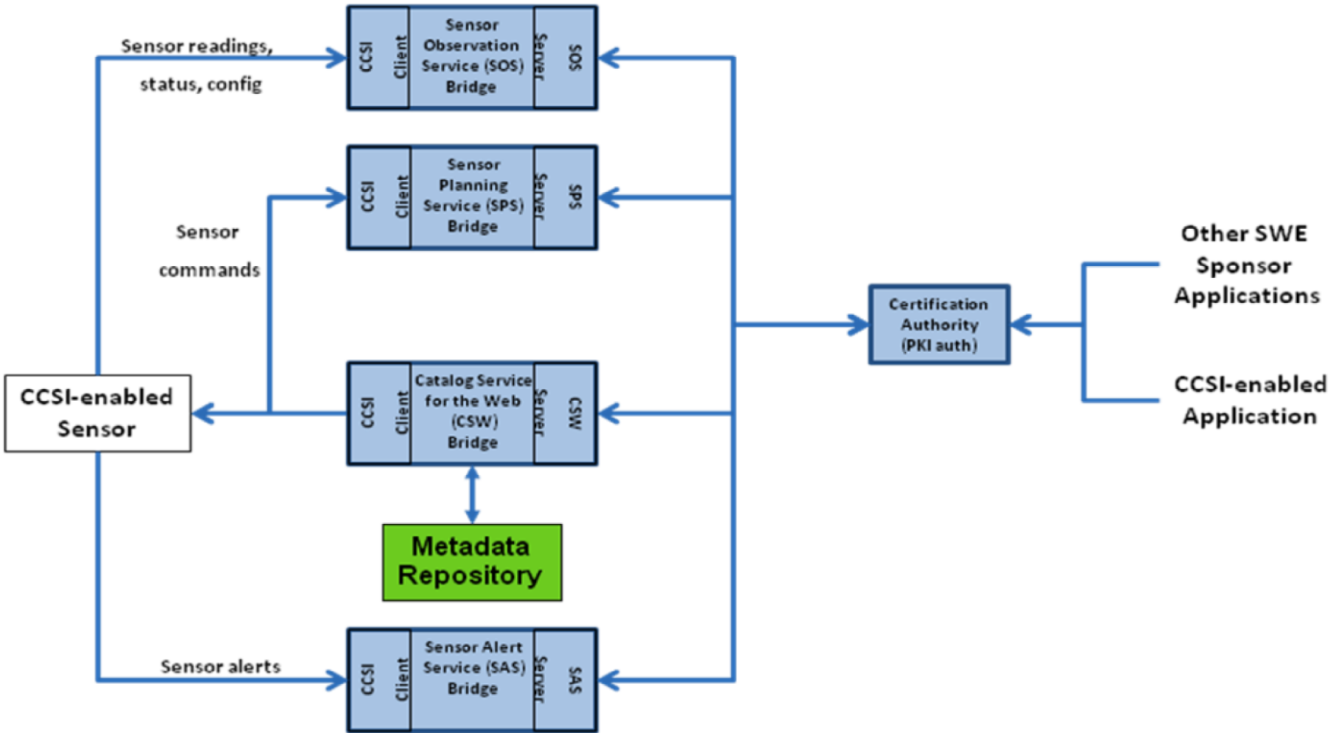


Figure 4-17. Notional CCSI – SWE Architecture

4.2 Geo-Processing Workflow (GPW)

4.2.1 GPW Scope

The Geo-Processing Workflow (GPW) thread aims to develop and demonstrate interoperability among geo-processes through service chaining, workflow and web services, with emphasis on implementing security capabilities for OGC web services, including SWE services . Work in this thread shall build on the results from previous testbeds, which includes authentication/authorization from OWS-4 and SOAP/WSDL recommendations from OWS-5. The workflow and security tasks shall exercise and document results based on three operational security environments: 1) internal to a single trusted domain; 2) between two trusted domains; and 3) between a trusted and non-trusted (or temporarily-trusted) domain. This work shall evaluate and use already approved existing workflow and security standards from OGC and others such as W3C, OASIS, etc when applicable.

The results in this thread will be realized through valued-added workflow scenarios that demonstrate interoperability and service-oriented architectures, and OGC Engineering Reports describing them.

The main topics in the OWS-6 GPW thread are as follows (with the first five bullets of particular relevance for SWE thread):

- Web Services Security and Asynchronous Workflow
- Operational Security Environments
- Web Services Security Environments and Solutions
- Asynchronous Workflows Within and Across Security Domains
- Data Security
- Open Grid Forum (OGF) Enabled Web Processing Services (WPS)
- GML Application Schema Development & ShapeChange Enhancements
- Web Processing Service Profiles
- Data Accessibility for DCS-enabled WFS

4.2.2 GPW Requirements

The following paragraphs describe the tasks and requirements to be satisfied for the GPW thread.

4.2.2.1 Web Services Security and Asynchronous Workflow

Implementation of robust, enterprise spanning applications require security issues and concerns for web services be evaluated, analyzed and demonstrated in a integrated architecture to satisfy mission critical objectives. In this thread, a single demonstrable work environment will address three operational pictures representing three security domains defined below, which shall be implemented as a basis to analyze security requirements, evaluate solutions and make recommendations to address web security issues in these three operational environments.

Participants shall make use of recommendations from OWS-4 (authentication/authorization) and OWS-5 (SOAP/WSDL) as part a solution. Participants shall evaluate and implement web service security recommendations using already-approved security standards, such as those from OGC, W3C, OASIS, etc.

when applicable. Implementations and recommendations shall take into account at a minimum WS-Security, WS-Policy, SAML 1.1/2.0, XACML, XML Encryption, XML Signature, and GeoXACML.

4.2.2.2 Operational Security Environments

Three operational security environments are required for this thread:

- Internal – web services are entirely defined within a single domain (i.e. private network).
- External (trusted) – web services are accessed, exercised, and called between domains which are trusted.
- External (non-trusted) – web services are accessed, exercised, and called between trusted and non-trusted or temporarily trusted domains.

We recognize that even "Internal" can be attacked. One example that usually comes from inside a trusted network is espionage. Further on, security gaps between different networks shall be considered.

The term 'Trust' as used above is defined by OASIS as follows:

> *"The characteristic that one entity is willing to rely upon a second entity to execute a set of actions and/or to make set of assertions about a set of subjects and/or scopes."*

4.2.2.3 Web Services Security Environments and Solutions

Web services security measures shall be evaluated, exercised and solutions recommended addressing the requirements spanning multiple security domains using a variety of OGC web services and security standards. It is assumed that the one work environment that the contractor sets up will be able to handle the 3 operational pictures that have been described above through different soft-controlled configurations rather than different hardware. For example it is suggested that a set of XML appliance capabilities such as encryption, digital signatures and denial of service attack prevention could be the primary means of providing access control and of providing for the appropriate security environment that the operational picture requires.

Participants shall set up a single security environment to effectively test implementations of web services and workflows within and across the three security operational pictures to include, but not be limited to, the use of a Policy Information Point (PIP) also referred to as an Authentication Service, Policy Decision Point (PDP) or Authorization Service, Policy Administration Point (PAP) or License Manager, and a Policy Enforcement Point (PEP) or Gatekeeper.[1] See Section 4.2.5.6.4 GeoXACML, and Figure 4-19 for more about these security components.

Furthermore, with the XACML requirement and the PEP & PDP requirement described above, it is assumed that the XML appliances would be appropriately chosen to satisfy both sets of requirements and be compatible with an external XACML-based PAP and with appropriate PIPs.

To achieve the desired results in this task area, a variety of OGC web services will be required to operate within and across the three security environments; in addition, OGC web services that support different versions of security standards and binding patterns (SOAP vs non-SOAP and REST) shall be deployed to investigate interoperability and possible version-related differences in function.

As a part of the demonstrable work environment described above, consideration should be given to a browser-based front-end solution as well as a thick client implementation, to authenticate/authorize users (OWS-4) and allow for selection of the appropriate target resource. Due to the final operational

---

[1] Para. 7.1, Architecture in OGC 06-184r2, GeoDRM Engineering Viewpoint and supporting Architecture

environment for this system it is expected that Identity Federation will be in use to allow for the appropriately secure transport of credentials from a trusted Identity Provider to the Service Provider. The general principle is to allow for a strong claims-based identity system across different “administrative domains”.

The demonstrable work environment shall exercise non-repudiation mechanisms and shall experiment with both network layer and message layer security to identify the different options provided by the different approaches.

A Web single sign on (SSO) system, a method of access control that enables a user to log in once and gain access to the resources of multiple software systems without being prompted to log in again, is an optional part of the work environment. If included it should be compatible with the use of XACML/GeoXACML described earlier in terms at least of using the same administrative PAP environment as for the SOA infrastructure. As single sign-on provides access to many resources once the user is initially authenticated ("keys to the castle"), it increases the negative impact, in case the credentials are available to other persons and misused. Therefore, single sign-on requires an increased focus on the protection of the user credentials, and should ideally be combined with strong authentication method (e.g., smart cards).

Participants are encouraged to address one or more of these requirements; however, more than one participant is expected to be selected to satisfy the span and scope of these requirements to ensure validity of interoperability results.

Participants are encouraged, but not required, to propose solutions that integrate each of the P*P components and which would interface with more than one OGC web service (OWS) in the security architecture. If no integrated P*P solution is proposed, then the test requires at least a service capability for each of the P*P element of the architecture to accomplish the test requirement for implementation and testing of security and trust relationships using OGC Web Services. Reuse of P*P component services in each of the 3 security domains is desired for economies of resources and effort, if possible.

Evaluate, exercise, and make recommendations for solutions involving the security aspects for service implementations and integration in the following areas in the context of a real-world scenario documented as Use Case #2 (Section 4.4.4.4) in the DSS thread:

a. Asynchronous workflows shall be exercised internal to a single trusted domain and with other trusted and non-trusted domains

b. Security for OGC web services shall be exercised to include REST, SOAP and OGC services that don’t deploy SOAP

c. Interoperability of OGC web services shall be exercised which implement different versions of security standards such as SAML 1.1 and SAML 2.0; and SOAP 1.1 and SOAP 1.2.

d. Data access restrictions shall be exercised using attribute level restrictions based on U.S. Department of Defense IC-ISM security markings and geographic area restrictions using GeoXACML.

e. Audit trail reporting shall be exercised; for example for data - who accessed, when it was accessed, what data was accessed.

f. Non-repudiation mechanisms shall be exercised.

g. Message layer and network layer security aspects shall be analyzed and evaluated.

Participant shall describe which requirements or security measures it proposes to support. Participant shall identify if it proposes to deploy one or more OGC web service(s) to support testing of these requirements.

Due to the diverse nature of this task area, it is anticipated that a number of different participants would be required to meet the range of requirements and to achieve the desired results in this task area.

4.2.2.4 Asynchronous workflows within and across Security Domains

The participant shall investigate, evaluate and deploy asynchronous workflows that require access to services and resources that reside in one or more security domains.

Workflow implementations in this testbed shall focus on workflow engine-independent solutions for workflow management and shall build on the work completed in OWS-5 using standards defined by the Workflow Management Coalition (WfMC) such as WfXML, WfXML-R and XPDL, as well as other workflow tools such as ebXML and BPEL. Workflows developed in this area shall continue work started in OWS-5 to evaluate and exercise these capabilities.

Workflow engines and executable workflow scripts are desired to support the workflow requirements stated in 4.2.2.3 (a. through g.) above and for automating as much of the Conflation process stated in 4.2.2.8.1 below.

Results of work in this testbed to further the experience, architecture, capabilities and practices using workflow standards, methods, engines and workflow languages (such as BPEL, WfXML, BPMN and XPDL) with OGC Web Services shall be documented in an Engineering Report. This Engineering Report shall encompass the results of the work in security, asynchronous processing, RESTful and SOAP-based approaches, and conflation. The ER should be aimed at becoming a Best Practice document.

Participants of this thread shall build and test web service implementations as described in the following sections for three required security domains. The P*P service implementations in these three domains may share executables provided each instance of the service or component is able to behave as if it were an independent entity owned and operating within its own domain.

4.2.2.4.1 Single Trusted Domain

The participant shall analyze the requirements, evaluate potential solutions and implement recommendations to address security issues and concerns when web services are accessed, exercised or called from within a single trusted domain. Consideration and recommendations should take into account OGC SOAP enabled, non SOAP enabled and REST services.

4.2.2.4.2 Trusted to Trusted Domain

The participant shall analyze the requirements, evaluate potential solutions and implement recommendations to address security issues and concerns when web services are accessed, exercised, or called between trusted domains. Consideration and recommendations should take into account OGC SOAP enabled, non SOAP enabled, and REST services.

4.2.2.4.3 Trusted to Non-Trusted Domain

The participant shall analyze the requirements, evaluate potential solutions and implement recommendations to address security issues and concerns when web services are accessed, exercised, or called between trusted and non-trusted or temporarily trusted domains. Consideration and recommendations should take into account OGC SOAP enabled, non SOAP enabled and REST services.

4.2.2.5 Data Security

4.2.2.5.1 Temporal Data Access Restriction

The participant shall evaluate, test and demonstrate methods by which data use may be restricted based on temporality. For example, data files accessed can be useable only until a specific point in time after which the files are locked and cannot be opened.

The restricted data access capability based on temporality shall be exercised as part a demonstration to occur at the end of the testbed. Results of this evaluation should be captured and documented as part of the OGC Web Service Security Engineering Report.

4.2.2.5.2 Preserving Data and Message Integrity

The participant shall evaluate, test and demonstrate methods by which data integrity can be ensured. For example, the data is not changed or altered from the start of a request to the end delivery. As an element of security it is also important to consider the integrity of request and response messages as part of this exercise.

The data integrity capability shall be exercised as part a demonstration to occur at the end of the testbed. Results of this evaluation should be captured and documented as part of the OGC Web Service Security Engineering Report.

4.2.2.6 Open Grid Forum (OGF) Enabled Web Processing Service (WPS)

WPS should be able to benefit from and integrate with distributed computing resources and technologies to enable applications to scale out. Two potential ways to make use of OGF and related specifications, concepts and their implementations can be identified. First, WPS can interact with distributed computing resource in the backend (encapsulating other resources). Secondly, the WPS can become a fully embedded resource accessed by middleware (integration alongside other services). In OWS-6 we seek to produce a proof-of-concept implementation of WPS (as a service encapsulating resources and as a resource for other services) that allows OGF standards and technologies to be leveraged while remaining OGC compliant and without changing existing OGC standards.

In the first case, a WPS could use other services, for example, to send calculation jobs to compute resources or for accessing distributed databases. In this case well defined mechanisms are used to access additional external computation or data storage resource. This approach will be referred to as ‘encapsulation’ as the clients to the WPS are not exposed to any changes.

In the second case, existing OGC Web Services would be integrated within other environments. This approach offers opportunities to realize benefit from existing middleware solutions such as intelligent workflow management, reliability, security and other Quality of Service attributes. This approach is referred to as ‘integration’ as the WPS becomes a service that clients may utilize alongside others in distributed service oriented environment.

OGF has a family of specifications that can be applied to accessing distributed computing and data resources. In addition, the OGF community has much experience, some in the field of geospatial data set analysis, which can be applied to generating use cases to drive the development and use of the test bed. Many of these specifications and the practical experience are embodied in closed and open source software solutions. Relevant specifications might include, but are not limited to, the HPC-Basic Profile, the Simple

API for Grid Applications (SAGA), Grid-RPC, the Data Access and Integration set of specifications (WS-DAI-*), and the Web Services Resource Framework (WSRF).

Some prior work has been done in the OWS-4 Interoperability testbed to investigate use of WPS for processing large datasets using grid techniques based on WSRF. For existing OGC web services, such as WPS, to leverage available web service interfaces to compute and storage resources an approach to 'encapsulating' these services within the WPS must be taken. This activity should be supported by resources to demonstrate the viability of this work.

Clear identification of the motivations for the following is needed prior to participants committing to any specific activity as part of this collaborative test bed:

- Scenarios
- Specifications
- Middleware
- Resources

4.2.2.7 GML Application Schema Development and ShapeChange Enhancements

The sponsor is in the process of evaluating emerging formats for the exchange of geospatial information across the U.S. National System for Geospatial Intelligence (NSG) enterprise, to include transfer of information through Coalition networks. This work item is intended to test the feasibility of GML application schemas based on OGC's Geography Markup Language Version 3.2.1 and OGC's CityGML to encode NSG data and serve as a transfer format among NSG participants. This section describes the requirements for the development of GML application schemas to support the NSG Application Schema (NAS). GML 3.2.1 application schemas shall be created in accordance with ISO 19109 and the DGIWG Profile(s) of ISO 19107 that support two-dimensional topology. GML 3.2.1 application schemas shall include associated metadata XML for discovery of these application schemas through an OGC catalog based on guidance provided by the July 2007 DoD Discovery Metadata Specification (DDMS) (to be provided as GFI). CityGML application schemas shall be based CityGML v.1.0 and the DGIWG Profile(s) of ISO 19107 that support three-dimensional geometry/topology

4.2.2.7.1 Enhancements to the UML-GML Application Schema (UGAS) Tool ShapeChange, and Related Development

Changes shall be developed to enhance the tool to provide the following capabilities:

a. Generate GML 3.2.1 application schemas for the NSG Application Schema (NAS) version 2.6 (Phase 1)
b. Generate UrbanMSD GML application schemas based on CityGML and the DGIWG 3D spatial profile (Phase 1)
c. Create CityGML instance document for prototype Urban MSD (GFI provided as Shapefiles) utilizing the application schema identified as (b) above. This instance document shall include an associated ISO TC/211-19139 encoding of "dataset" level metadata. (Phase 1)
d. Enhance the UML to GML Application Schema software tool Shape Change to address the following requirements:
    1) Enhance ShapeChange to automate the generation of XSLT scripts to aid in portrayal of "data content specification" defined data using WFS, SLD and WMS Component (FPS)
    2) Enhance ShapeChange to handle application schemas that implement 3D, 4D and topology
    3) Enhance ShapeChange to make use of a variety of "dictionaries" such as Unit of Measure dictionary

e. Develop GML Profile Validation tool using ShapeChange utilization of Object Constraint Language (OCL) and or Schematron constraints to create a "data content specification" profile validation.
f. The contractor shall register the schemas developed or used within the scope of this project using the CS/W ebRIM profile. These will be queried as part of the Integrated Client functionality and possibly other client applications.
g. Schemas not fully developed shall be identified and their shortcomings shall be described along with an estimate of the effort necessary to "finish" them.

4.2.2.8 Web Processing Service Profiles

A number of OGC web services have been prototyped in previous testbeds. During this testbed the requirement is to advance these capabilities to mature the implementation to reach the specification stage. To meet this requirement, the participant shall perform analysis and evaluation of the current status of the service, implement and deploy the service to support recommendations on steps to move it forward.

4.2.2.8.1 WPS – Conflation Service Profile

The participant shall implement a WPS conflation profile service with additional enhancements to include capability to ingest and output GML 3.2.1 data as well as the ability to execute processes based on the use of an external "rules" web service. Participant shall be able to provide the external "rules" service for use in testing. The WPS Conflation Service profile shall use workflow enactment (engines and workflow) (ref: 4.2.2.4 above) to automate as much of the conflation process as possible.

The participant shall prepare and submit an Engineering Report aimed at becoming a Best Practice document to record the findings and recommendations resulting from the effort to implement the WPS Conflation Profile service. This Engineering Report shall provide the template for future WPS profile services.

4.2.2.8.2 Web Processing Service – Grid Processing Profile

Investigate, define and implement a WPS-Grid Profile which is integrated with a grid computing infrastructure using relevant specifications which might include, but are not limited to, the HPC-Basic Profile, the Simple API for Grid Applications (SAGA), Grid-RPC, the Data Access and Integration set of specifications (WS-DAI-*), and the Web Services Resource Framework (WSRF). This implementation is expected to be an early proof-of-concept which would establish a foundation for refinement of the profile and leading to more robust implementations in the future.

The participant shall prepare and submit an Engineering Report to record the findings and recommendations resulting from the effort to implement the WPS Grid Processing Profile service.

4.2.2.9 Data Accessibility for DCS-enabled WFS

Consider a situation where data, within a database, is encoded based on the full complete NSG Application Schema. Thus, a full set of attribution and all applicable geometry types are allowable. For example, within the NAS conformant database, a bridge might exist as a point, line or area, which may have 50 attributes and each attribute contains a potential of 10 enumerates. It is not expected that users will require a complete set of data holdings at all times; therefore, a series of Data Content Specifications (DCS), which are a subset of the full application schema, are derived from the NAS. A DCS will identify constraints against the full NAS where, for example, a Local MSD DCS might restrict the bridge to only point and line geometries and further limit the data to only 30 attributes and only 6 of the possible 10 enumerates on each attribute. The intent is to express each DCS without need to define a separate application schema. Rather, the intent is to define a solution to simplify the data retrieval process whereby a user may request data content based on the DCS with limited knowledge of the constraints defined in the DCS against the NAS.

The participant shall investigate and deploy a DCS-enabled Web Feature Service (WFS) with capability to demonstrate techniques and a recommended approach that may be used as a pattern to retrieve data from the NAS. The solution shall provide a recommended approach that will support the use of the DCS to retrieve the desired subset of data from an available Web Feature Service which is hosting a wider range of NAS conformant data. In addition, a client is required to provide access to and exercise the capabilities of the DCS-enabled Web Feature Service. This work shall be coordinated with the geospatial Decision Support Services (DSS) thread of OWS-6, to the extent that the DSS Integrated Client must be capable of providing this access and display capability. See section 4.4.2.5 for more information.

The participant shall prepare and submit an Engineering Report to record the findings and recommendations resulting from the effort to implement the DCS-enabled WFS service. The participant shall coordinate with the DSS thread team, which will prepare an annex to this ER to describe any recommendations and issues with the use of FPS, SLD and XSLT for data content specification portrayal from a WFS output to a WMS.

The participant shall prepare and submit WFS, Filter Encoding or other Change Requests for consideration by the Specification program resulting from the implementation of the DCS-enabled WFS service.

4.2.3 GPW Deliverables

The OWS-6 testbed requires two types of deliverables:

- **Engineering Reports** shall be prepared in accordance with OGC published templates as defined on the OGC portal. Engineering Reports shall be delivered by posting on the OGC Portal Pending Documents list when complete and the document has achieved a satisfactory level of consensus among interested participants, contributors and editors. Engineering Reports are the formal mechanism used to deliver results of the Interoperability Program to sponsors and to the OGC Specification Program Domain and Specification Working Groups for consideration. All documents created in response to this program shall include "OWS-6" in the title, to facilitate later literature searches.
- **Services, Clients and Tools** shall be provided by methods suitable to its type and stated requirements. For example, services and components (ex. Deployed WFS, WMS, etc.) are delivered by deployment of the service or component for use in the testbed via an accessible URL. A Client software application may be used during the testbed to exercise services and components to test and demonstrate interoperability; however, it is most often not delivered as a license for follow-on usage.

Note that certain draft deliverables will be required by the Interim Milestone at the date shown in the Master Schedule (Main Body, Section 4.6), for use in cross-thread development. These early deliverables will be designated and handled on a thread-by-thread basis.

4.2.3.1 Engineering Reports (ERs) and Documents

The following documents shall be prepared and delivered as a result of this testbed. Engineering Reports shall be reviewed by interested participants and contributors within the testbed for content, accuracy and consistency of purpose. When the Engineering Report achieves a satisfactory level of consensus, it shall be posted to the OGC Portal Pending Documents list with an assigned OGC document number.

Document Title
OWS-6 OGC Web Services Security ER
OWS-6 GeoXACML documents
OWS-6 GML 3.2.1 application schemas for NAS
OWS-6 CityGML application schema Urban MSD
OWS-6 GEOINT Profile (GSIP) Schema Processing ER (Update)
OWS-6 Urban MSD Implementation Profile
OWS-6 GML Profile Validation Tool Guidelines ER
OWS-6 WPS – Conflation Profile ER
OWS-6 WPS – Grid Processing Profile ER
OWS-6 GeoProcessing Workflow ER
OWS-6 CityGML CR
OWS-6 WFS and FE CRs (various, as required)
OWS-6 WFS Data Accessibility ER

4.2.3.2 Services, Clients and Tools

Service implementations for use in this testbed shall be deployed during the testbed for integration and interoperability testing. In addition, unless otherwise stated below, these services shall remain available for testing and experimentation for a period of six (6) months following the completion of the testbed.

Service, Client or Tool Title
P*P component (PDP, PIP, PAP) (cross-thread with SWE)
Gatekeeper (PEP) for two secured OWS services (may be supplied as separate proxy or integrated with the Secured OWS) (cross-thread with SWE)
Conflation workflow engine (cross-thread with SWE)
CS/W ebRIM profile server and associated metadata (Datasets, Services, Schemas Register, Conflation Rules, Portrayal Rules)
Enhanced UGAS ShapeChange Tool
GML Profile Validation Tool
WFS – CityGML Urban MSD
WPS – Conflation Service
WPS – Grid Processing
CityGML instance document (dataset) for Urban MSD (delivered preferably on a CD)
WFS – Data for conflation (2 separate servers)

4.2.4 GPW Enterprise Viewpoint

The work of OGC Interoperability and Specification Programs has produced a significant body of knowledge and experience in designing, building and deploying Web Services. The full potential of OGC Web Services as an integration platform will be achieved when applications and business processes can be composed to perform complex interactions using a standard process integration approach.

The Geo-Processing Workflow (GPW) thread in this testbed aims to continue to advance the work accomplished in prior testbeds to develop and demonstrate interoperability among geo-processes through publish-find-bind, service chaining and workflow orchestration. The results will be realized through scenarios that demonstrate composition and interoperability of OGC web services in a service-oriented architecture.

The OWS-6 GPW thread will focus on extension or enhancement of capabilities and integration of OGC web services specifications to address operational security issues for services that may interoperate within a single trusted domain and with other trusted and non-trusted domains.

4.2.4.1 Support of Strategic Objectives

The OWS-5 Interoperability Program (IP) testbed aims to investigate, evaluate and demonstrate certain capabilities and enhancements of OGC web services that are relevant to achieving strategic objectives for modernization goals being pursued by a number of federal agencies and organizations.

4.2.4.2 Asynchronous Workflow

With the growth and maturation of enterprise environments, it becomes necessary to produce more complex functional capabilities which are composed from a variety of existing services using workflow orchestration and choreography. Workflow technologies and standards have continued to advance over time, producing a variety of useful approaches upon which to build complex operations to support the enterprise. These technologies have mostly focused on implementation of workflow processes in the form of a runtime execution language or script for an associated process engine. This approach provides an effective means to deploy and execute processes within a homogeneous environment served by a particular process engine. However, to meet the needs within and across enterprises which may be using different process engines and languages a more abstract approach is needed to facilitate design, integration, execution and management of these processes many of which will be asynchronous by nature. This testbed aims to build on the work accomplished during OWS-5 using Workflow Management Coalition's (WfMC) Wf-XML and XPDL standards to express process models using XPDL to connect asynchronous services, whether they are implemented as process engines or other processing services such as OGC's Web Processing Service (WPS), Workflow Chaining Service (WfCS) or even SensorML for sensor specific needs. Wf-XML (and emerging Wf-XML-R) and XPDL standards provides a way to merge process-oriented tools and technologies into a generic invocation framework.

In addition, approaches to deploy asynchronous workflow should investigate and test approaches to integrate technologies which use OASIS's Business Process Execution Language (BPEL).

4.2.4.3 GPW Use Cases

The use cases provided in this section provide a framework to clarify and address the requirements described in the preceding GPW requirements sections (para. 4.2.2 et.al.)

4.2.4.3.1 Use Case #1: Single Trusted Domain Security

<table>
<tr><td>Use Case Id:</td><td>GPW #1</td><td>Use Case Name:</td><td>Processing IC-ISM classified data within a single trusted domain</td></tr>
<tr><td>Use Case Domain:</td><td>OWS-6 GPW for Security Domains</td><td>Status:</td><td>Draft 2008-07-18</td></tr>
<tr><td>Use Case Description:</td><td colspan="3">A source in a trusted domain maintains and serves IC-ISM classified data to authorized users within the same trusted domain</td></tr>
<tr><td>Actors (Initiators):</td><td>User1: owner/custodian of IC-ISM classified data
User2: trusted user requesting data</td><td>Actors (Receivers)</td><td>User2: trusted user receiving data</td></tr>
<tr><td colspan="2">Pre-Conditions:
- User1 manages secured data for publication and access.
- IC-ISM classified data is available for query and retrieval by authorized users
- User2 has appropriate credentials to request Desired Data.</td><td colspan="2">Post-Conditions:
- User2 retrieves and displays Requested Data</td></tr>
<tr><td colspan="4">System Components
- Authentication Service (PIP): authenticate users who wish to retrieve data from a trusted source
- Authorization Service (PDP): Grant permission for requesting user to access resources based upon access rights
- License Manager (PAP): maintains policies to determine who may retrieve data based on identity and IC-ISM classification of data.
- Gatekeeper (PEP): provides interface between the client and the OGC web service to control access to data based on authentication and authorization results
- WFS: serves feature data with controlled access
- CS-W: repository of service offerings, data and metadata for IC-ISM classified data</td></tr>
</table>

Use Case Id:	GPW #1	Use Case Name:	Processing IC-ISM classified data within a single trusted domain

Basic Course of Action

1. User2 enters his credentials to the PIP, which verifies his identity and returns an Identity Token to User2 to certify his identity
2. User2 queries CS/W for Desired Data using the Identity Token provided by the PIP.
3. CS/W returns Query Result describing data available to User2.
4. User2 examines the Query Result to determine that the Desired Data exists on WFS. The Query Result also indicates that Requested Data is IC-ISM coded which requires that User2 have appropriate authorization to retrieve the Requested Data.
5. User2 prepares and issues request to retrieve the data from WFS via Gatekeeper (PEP) using his Identity Token.
6. PEP contacts PDP providing User2 Identity Token and metadata about the Requested Data to obtain authorization for User2 to retrieve Requested Data.
7. PDP contacts PAP to determine policies for retrieval of the Requested Data by User2
8. PAP returns result to PDP to determine is User2 is authorized to retrieve Requested Data.
9. PDP notifies PEP that User2 is authorized to retrieve the Requested Data.
10. PEP submits request to WFS for the Requested Data.
11. WFS returns Requested Data to User2 via the PEP.

4.2.4.3.2 Use Case #2: Trusted-to-Trusted Domain Security

Use Case Id:	GPW #2	Use Case Name:	Processing IC-ISM classified data across two trusted domains
Use Case Domain:	OWS-6 GPW for Security Domains	Status:	Draft 2008-07-18
Use Case Description:	A source in a trusted domain maintains and serves IC-ISM classified data to authorized users in a separate trusted domain		
Actors (Initiators):	User1(domain1): owner/custodian of IC-ISM classified data User2 (domain2): trusted user requesting data	Actors (Receivers)	User2 (domain2): trusted user receiving data
Pre-Conditions: - User1 manages secured data for publication and access. - IC-ISM classified data is available for query and retrieval by authorized users - User2 has appropriate credentials to request needed data.		Post-Conditions: - User2 retrieves and displays requested data from Domain1	

Use Case Id:	GPW #2	Use Case Name:	Processing IC-ISM classified data across two trusted domains
System Components - Authentication Service (Domain1) (PIP1): authenticate users for Domain1 trusted sources - Authentication Service (Domain2) (PIP2): authenticate users who wish to retrieve data from a Domain2 trusted source - Authorization Service (Domain1) (PDP1): Grant permission for requesting user to access resources based upon access rights - Authorization Service (Domain2) (PDP2): Grant permission for requesting user to access resources based upon access rights - License Manager (Domain1) (PAP1): maintains policies to determine who may retrieve data based on identity and IC-ISM classification of data. - Gatekeeper (Domain1) (PEP1): provides interface between the client and the OGC web service to control access to data based on authentication and authorization results for Domain1 - WFS: serves feature data with controlled access (Domain1) - CS-W: repository of service offerings, data and metadata for IC-ISM classified data (Domain1)			
1. User2 enters his credentials to the PIP2, which verifies his identity and returns an Identity Token to User2 to certify his identity 2. User2 queries CS/W for Desired Data using the Identity Token provided by the PIP2. 3. CS/W returns Query Result describing data available to User2. 4. User2 examines the Query Result to determine that the Desired Data exists on WFS. The Query Result also indicates that Requested Data is IC-ISM coded which requires that User2 have appropriate authorization to retrieve the Requested Data. 5. User2 prepares and issues request to retrieve the data from WFS via Gatekeeper (PEP1) using his Identity Token. 6. PEP1 contacts PDP1 providing User2 Identity Token and metadata about the Requested Data to obtain authorization for User2 to retrieve Requested Data. 7. PDP1 contacts PAP1 to determine policies for retrieval of the Requested Data by User2 8. PAP1 returns result to PDP1 to determine is User2 is authorized to retrieve Requested Data. 9. PDP1 notifies PEP1 that User2 is authorized to retrieve the Requested Data. 10. PEP1 submits request to WFS for the Requested Data. 11. WFS returns Requested Data to User2 via the PEP1.			

4.2.4.3.3 Use Case #3: Trusted to Temporarily-Trusted Security Domain

Use Case Id:	GPW #3	Use Case Name:	Processing IC-ISM classified data between a trusted domain and a temporarily trusted domain
Use Case Domain:	OWS-6 GPW for Secure Domains	Status:	Draft 2008-07-18
Use Case Description:	A source in a trusted domain (domain1) maintains and serves IC-ISM classified data to authorized users in a temporarily trusted domain (domain3)		
Actors (Initiators):	User1(domain1): owner/custodian of IC-ISM classified data User3 (domain3): temporarily-trusted user requesting data	Actors (Receivers)	User3 retrieves and displays requested data from Domain1
Pre-Conditions:		Post-Conditions:	

Use Case Id:	GPW #3	Use Case Name:	Processing IC-ISM classified data between a trusted domain and a temporarily trusted domain
- User1 manages secured data for publication and access. - IC-ISM classified data is available for query and retrieval by authorized users - User3 has appropriate credentials to request needed data.		- User3 retrieves and displays requested data	
System Components - Authentication Service (Domain1) (PIP1): authenticate users for Domain1 trusted sources - Authentication Service (Domain2) (PIP3): authenticate users who wish to retrieve data from a Domain3 trusted source - Authorization Service (Domain1) (PDP1): Grant permission for requesting user to access resources based upon access rights - Authorization Service (Domain2) (PDP3): Grant permission for requesting user to access resources from Domain3 based upon access rights - License Manager (Domain1) (PAP1): maintains policies to determine who may retrieve data based on identity and IC-ISM classification of data. - Gatekeeper (Domain1) (PEP1): provides interface between the client and the OGC web service to control access to data based on authentication and authorization results for Domain1 - WFS: serves feature data with controlled access (Domain1) CS-W: repository of service offerings, data and metadata for IC-ISM classified data (Domain1)			
Basic Course of Action 1. User3 enters his credentials to the PIP3, which verifies his identity and returns an Identity Token to User3 to certify his identity 2. User3 queries CS/W for Desired Data using the Identity Token provided by the PIP3. 3. CS/W returns Query Result describing data available to User3. 4. User3 examines the Query Result to determine that the Desired Data exists on WFS. The Query Result also indicates that Requested Data is IC-ISM coded which requires that User3 have appropriate authorization to retrieve the Requested Data. 5. User3 prepares and issues request to retrieve the data from WFS via Gatekeeper (PEP1) using his Identity Token. 6. PEP1 contacts PDP1 providing User2 Identity Token and metadata about the Requested Data to obtain authorization for User3 to retrieve Requested Data. 7. PDP1 contacts PAP1 to determine policies for retrieval of the Requested Data by User3 8. PAP1 returns result to PDP1 to determine is User3 is authorized to retrieve Requested Data. 9. PDP1 notifies PEP1 that User3 is authorized to retrieve the Requested Data. 10. PEP1 submits request to WFS for the Requested Data. 11. WFS returns Requested Data to User3 via the PEP1.			

4.2.4.4 Use Case #4: Publish GML Application Schemas

Use Case Id:	GPW #4	Use Case Name:	Geographic Information Publication
Use Case Domain:	OWS-6 Publish GML Applications Schema	Status:	Draft 2008-07-18
Use Case Description:	This use case describes the process for translating geographic models from UML into GML, creating GML instance documents based on that model, and publishing both the model and instance documents in a catalog.		

<table>
<tr><td>Use Case Id:</td><td>GPW #4</td><td>Use Case Name:</td><td colspan="2">Geographic Information Publication</td></tr>
<tr><td>Actors (Initiators):</td><td colspan="2">Data custodian</td><td>Actors (Receivers)</td><td>Catalog service</td></tr>
<tr><td colspan="3">Pre-Conditions:
- Initiator has data
- Initiator has GML competency
- Initiator is proficient with UGAS tool
- Initiator has write access to catalog service</td><td colspan="2">Post-Conditions:
- GML application schemas created
- GML instance documents created
- Both published to catalog</td></tr>
<tr><td colspan="5">System Components
- UGAS ShapeChange tool
- CS-W: Catalog Service Web Profile
- XML editing software</td></tr>
<tr><td colspan="5">Basic Course of Action (numbered list of actions) (include optional/alternative paths, as appropriate)
1. User develops UML model
2. User exports information model in XMI format
3. User exercises UGAS tool on the XMI export of the UML model
4. User processes data from native format into GML
5. Users publishes UML, GML application schema and GML instance documents to catalog</td></tr>
</table>

4.2.5 GPW Information Viewpoint

The Information Viewpoint considers the information models and encodings that will make up the content of the services and exchanges to be extended or developed to support OWS-6. This section will consider these according to the areas discussed in the Enterprise Viewpoint.

4.2.5.1 Workflow Process Models and Encodings

Relevant Documents and Specifications:

- OWS-5 GeoProcessing Workflow Architecture Engineering Report (OGC 07-138r1)
- “The Workflow Reference Model”, The Workflow Management Coalition, WfMC-TC- 1003, 1.1, 19 Jan 1995. http://www.wfmc.org/standards/docs/tc003v11-16.pdf
- Workflow Standard - Interoperability Wf-XML Binding (Wf-XML) (v1.1 and v2.0) (ref: www.wfmc.org)
- Workflow Standard - Interoperability Wf-XML-R Binding (RESTful) (WfXML-R) (Draft 0.4)
- XML Process Definition Language (XPDL) (ref:: www.wfmc.org)
- Business Process Modeling Notation (BPMN) v1.0 (Adopted Standard) (ref: www.bpmn.org)
- Web Services Business Process Execution Language (WS-BPEL 2.0) (ref: www.oasis-open.org)

Workflows in OWS-6 aim to continue work from previous testbeds to evaluate and test standards and techniques that separate design and modeling of workflow processes from its executable form as a script to be run on a process engine. This approach also promotes workflow engine interoperability and interoperability with a wider variety of workflow implementation languages.

Wf-XML (WfMC)

Wf-XML utilizes a loosely coupled, message-based approach to facilitate rapid implementation using existing technologies. It will describe the syntax of these messages in an open, standards-based fashion that allows for the definition of a structured, robust and customizable communications format.

Wf-XML can be used to implement three models of interoperability; specifically, chained workflows, nested workflows and parallel-synchronized workflows. Wf-XML supports these three types of interchanges both synchronously and asynchronously, and allows messages to be exchanged individually or in batch operations. Furthermore, this specification describes a language that is independent of any particular implementation mechanism, such as programming language, data transport mechanism, OS/hardware platform, etc.

XPDL (WfMC)

XPDL provides an XML file format that can be used to interchange process models between tools. This specification forms part of the documentation relating to Interface 1 - supporting Process Definition Import and Export shown in the diagram above. This interface includes a common meta-model for describing the process definition (this specification) and also a companion XML schema for the interchange of process definitions.

An XPDL package corresponds to a Business Process Diagram (BPD) in Business Process Markup Notation (BPMN), and consists of a set of Process Definitions. The WfMC defines a process as:

> *The representation of a business process in a form that supports automated manipulation, such as modeling, or enactment by a workflow [or business] management system. The process definition consists of a network of activities and their relationships, criteria to indicate the start and termination of the process, and information about the individual activities, such as participants, associated IT applications and data, etc. (WfMC Glossary - WfMC-TC-1011)*

The process definition provides an environment for a rich description of a process that can be used for the following:

- Act as a template for the creation and control of instances of that process during process enactment.
- For simulation and forecasting.
- As a basis to monitor and analyse enacted processes.
- For documentation, visualization, and knowledge management.

The process definition may contain references to subflows, separately defined, which make up part of the overall process definition.

BPMN (OMG)

Business Process Modeling Notation (BPMN) is a standardized graphical notation for drawing business processes in a workflow. BPMN was developed by Business Process Management Initiative (BPMI), and is now being maintained by the Object Management Group since the two organizations merged in 2005. Its current adopted version is 1.1 and the proposed one is 2.0.

The primary goal of BPMN is to provide a standard notation that is readily understandable by all business stakeholders. These business stakeholders include the business analysts who create and refine the processes, the technical developers responsible for implementing the processes, and the business managers who monitor and manage the processes. Consequently BPMN is intended to serve as common language to bridge the communication gap that frequently occurs between business process design and implementation.

WS-BPEL (OASIS)

Web Services Business Process Execution Language (WS-BPEL), sometimes referred to in this RFQ as BPEL, defines a model and a grammar for describing the behavior of a business process based on interactions between the process and its partners. The interaction with each partner occurs through Web Service interfaces, and the structure of the relationship at the interface level is encapsulated in what is called a partnerLink. The WS-BPEL process defines how multiple service interactions with these partners are coordinated to achieve a business goal, as well as the state and the logic necessary for this coordination. WS-BPEL also introduces systematic mechanisms for dealing with business exceptions and processing faults. Moreover, WS-BPEL introduces a mechanism to define how individual or composite activities within a unit of work are to be compensated in cases where exceptions occur or a partner requests reversal.

WS-BPEL utilizes several XML specifications: WSDL 1.1, XML Schema 1.0, XPath 1.0 and XSLT 1.0. WSDL messages and XML Schema type definitions provide the data model used by WS-BPEL processes. XPath and XSLT provide support for data manipulation. All external resources and partners are represented as WSDL services. WS-BPEL provides extensibility to accommodate future versions of these standards, specifically the XPath and related standards used in XML computation.

BPEL vs XPDL

BPEL and XPDL are entirely different yet complimentary standards. BPEL is an "execution language" designed to provide a definition of web services orchestration, specifically the underlying sequence of interactions, the flow of data from point-to-point. For this reason, it is best suited for straight-through processing or data-flows vis-a-vis application integration. The goal of XPDL is to store and exchange the process diagram, to allow one tool to model a process diagram, and another to read the diagram and edit, another to "run" the process model on an XPDL-compliant BPM engine, and so on. For this reason, XPDL is not an executable programming language like BPEL, but specifically a process design format that literally represents the "drawing" of the process definition.

4.2.5.2 GML 3.2.1

Relevant Specifications:

- OpenGIS® Geography Markup Language (GML) Encoding Specification 3.2.1 (OGC 07-036) (http://portal.opengeospatial.org/files/?artifact_id=20509)

The Geography Markup Language (GML) is an XML encoding for the transport and storage of geographic information, including both the geometry and properties of geographic features. Work to be performed in this thread shall use of GML version 3.2.1 unless otherwise indicated.

4.2.5.3 GML Application Schema for Urban Environments

Relevant Specifications:

- GEOINT Structure Implementation Profile (GSIP) Schema Processing IPR (07-028)

- OpenGIS® City Geography Markup Language (CityGML) Implementation Specification (candidate) (OGC 08-007r1)

OWS-6 will investigate and test the feasibility of GML application schemas based on OGC's Geography Markup Language Version 3.2.1 and OGC's CityGML to encode NSG data and serve as a transfer format among NSG participants. This section describes the requirements for the development of GML application schemas to support the NSG Application Schema (NAS). GML 3.2.1 application schemas shall be created in accordance with ISO 19109 and the DGIWG Profile(s) of ISO 19107 that support two-dimensional topology. GML 3.2.1 application schemas shall include associated metadata XML for discovery of these application schemas through an OGC catalog based on guidance provided by the July 2007 DoD Discovery Metadata Specification (DDMS) (to be provided as GFI). CityGML application schemas shall be based CityGML v.1.0 and the DGIWG Profile(s) of ISO 19107 that support three-dimensional geometry/topology

4.2.5.4 CityGML

Relevant Specifications:

- OpenGIS® City Geography Markup Language (CityGML) Implementation Specification (candidate) (OGC 08-007r1)

CityGML is an open data model and XML-based format for the storage and exchange of virtual 3D city models. It is an application schema for the Geography Markup Language version 3.1.1 (GML3), the extendible international standard for spatial data exchange issued by the Open Geospatial Consortium (OGC) and the ISO TC211.

The aim of the development of CityGML is to reach a common definition of the basic entities, attributes, and relations of a 3D city model. This is especially important with respect to the cost-effective sustainable maintenance of 3D city models, allowing the reuse of the same data in different application fields.

CityGML not only represents the graphical appearance of city models but specifically addresses the representation of the semantic and thematic properties, taxonomies and aggregations. CityGML includes a geometry model and a thematic model. The geometry model allows for the consistent and homogeneous definition of geometrical and topological properties of spatial objects within 3D city models (chapter 8). The base class of all objects is CityObject which is a subclass of the GML class Feature. All objects inherit the properties from CityObject.

The thematic model of CityGML employs the geometry model for different thematic fields like Digital Terrain Models, sites (i.e. buildings; future extensions of CityGML will also include explicit models for bridges and tunnels), vegetation (solitary objects and also areal and volumetric biotopes), water bodies, transportation facilities, and city furniture. Further objects, which are not explicitly modeled yet, can be represented using the concept of generic objects and attributes. In addition, extensions to the CityGML data model applying to specific application fields can be realized using the Application Domain Extensions (ADE). Spatial objects of equal shape which appear many times at different positions like e.g. trees, can also be modeled as prototypes and used multiple times in the city model. A grouping concept allows the combination of single 3D objects, e.g. buildings to a building complex. Objects which are not geometrically modeled by closed solids can be virtually sealed in order to compute their volume (e.g. pedestrian underpasses, tunnels, or airplane hangars). They can be closed using ClosureSurfaces. The concept of the TerrainIntersectionCurve is introduced to integrate 3D objects with the Digital Terrain Model at their correct positions in order to prevent e.g. buildings from floating over or sinking into the terrain.

CityGML differentiates five consecutive Levels of Detail (LOD), where objects become more detailed with increasing LOD regarding both their geometry and thematic differentiation. CityGML files can - but do not have to - contain multiple representations (and geometries) for each object in different LOD simultaneously. Generalization relations allow the explicit representation of aggregated objects over different scales.

In addition to spatial properties, CityGML features can be assigned appearances. Appearances are not limited to visual data but represent arbitrary observable properties of the feature's surface such as infrared radiation, noise pollution, or earthquake-induced structural stress.

Furthermore, objects can have external references to corresponding objects in external datasets. Enumerative object attributes are restricted to external code lists and values defined in external, re-definable dictionaries.

4.2.5.5 Security Markings

Relevant Document:

- IC ISM Schema Version 2.0

The IC ISM XML Schema is described in the IC Information Security Marking (IC ISM) Data Element Dictionary and the IC ISM Implementation Guide. It is one of the Intelligence Community (IC) Metadata Standards for Information Assurance. The goal of the IC ISM XML Schema is to provide a common set of XML attributes for implementing security-based metadata throughout the IC. The IC ISM XML Schema provides markup for the tokens that are used to format the CAPCO markings.

The IC ISM XML Schema may be incorporated into organizational XML-based schemas by (a) declaring the IC ISM v2.0 namespace and (b) inserting an "import" statement.

Refer to the Data Element Dictionary and Implementation Guide for an explanation of the relationships of the IC ISM attributes and the associated controlled vocabularies. The CAPCO Register and CAPCO Implementation Manual provide business rules (that may be classified) not provided in the DED or Implementation Guide.

4.2.5.6 Web Services Security and Trust

Relevant Standards and Documents:

- WS-Security v1.1
- WS-Trust v 1.3
- WS-Policy
- WS-ReliableMessaging
- WS-Federation
- XML Digital Signature
- XML Encryption
- Security Access Markup Language (SAML) v 1.1, 2.0
- XML Access Control Markup Language (XACML)
- Geospatial eXtensible Access Control Markup Language (GeoXACML) (OGC 07-026r2)
- OWS-4 Trusted GeoServices (OGC 06-107r1)
- OWS-4 GeoDRM Engineering Viewpoint and supporting Architecture (OGC 06-184r2)
- Geospatial Digital Rights Management Reference Model (GeoDRM RM) (OGC 06-004r3)
- ISO 15000, 15408, 15443, 10181

See Table 3-8 for summaries and links to the non-OGC standards just listed. Web services security is founded on the following concepts[2]:

- **Authentication**: Who is accessing the resource? Verify that principals (humans or application components) are who they claim to be through appropriate proof of identity. Determine the identity or role of a party attempting to perform some action, such as accessing a resource or participating in a transaction.
- **Authorization**: What can they do? Grant permission for principals to access resources based upon access rights. Determine whether some party is allowed to perform a requested action or access particular resources.
- **Integrity**: Ensure that information is intact. Ensure that information is not changed in transit, either due to malicious intent or by accident. This may be information transmitted over a network, information stored in a database or file system, or information passed in a Web services message and processed by intermediaries.
- **Non-repudiation**: Verify the identity of authors using electronic signatures. Produce or verify an electronic signature for purposes such as approval, confirmation of receipt, acceptance or agreement.
- **Confidentiality**: Make content unreadable by unauthorized parties. Ensure that only legitimate parties may view content, even if other access control mechanisms are bypassed, and guarantee that exchanged information is protected against eavesdroppers. Confidentiality is generally associated with encryption technologies.
- **Privacy**: Limit access and use of individually identifiable information. Personally identifiable information is required by individuals and organizations to perform services for an individual.

Ensuring the security of Web services involves implementation of security frameworks based on use of authentication, authorization, confidentiality, and integrity mechanisms which include the following security techniques.

- Confidentiality - Using XML Encryption as a mechanism to encrypt XML documents
- Integrity - Using XML Signature to provide a means to selectively sign XML data
- Authentication and Authorization - Using SAML and GeoXACML/XACML
- Public Key Infrastructure (PKI) - using XKMS
- WS-Security - SOAP header extensions for end-to-end SOAP messaging security which supports message integrity and confidentiality.

4.2.5.6.1 WS-Security (OASIS)

The WS-Security standard proposes a standard set of SOAP extensions that can be used when building secure Web services to implement message content integrity and confidentiality. It refers to a set of extensions and modules known as the “Web Services Security: SOAP Message Security” or “WSS: SOAP Message Security”.

WS-Security is flexible and designed to be used as the basis for securing Web services within a wide variety of security models including PKI, Kerberos, and SSL. Specifically, it provides support for multiple security token formats, multiple trust domains, multiple signature formats, and multiple encryption technologies. The token formats and semantics for using these are defined in associated profile documents.

---

[2] Guide to Web Services Security (DRAFT), NIST Special Pub 800-96, September 2006

WS-Security provides three main mechanisms: ability to send security tokens as part of a message, message integrity, and message confidentiality. These mechanisms by themselves do not provide a complete security solution for Web services. Instead, WS-Security is a building block that can be used in conjunction with other Web service extensions and higher-level application-specific protocols to accommodate a wide variety of security models and security technologies.

These mechanisms can be used independently (e.g., to pass a security token) or in a tightly coupled manner (e.g., signing and encrypting a message or part of a message and providing a security token or token path associated with the keys used for signing and encryption).

4.2.5.6.2 WS-Trust (OASIS)

WS-Security defines the basic mechanisms for providing secure messaging. It uses these base mechanisms and defines additional primitives and extensions for security token exchange to enable the issuance and dissemination of credentials within different trust domains.

In order to secure a communication between two parties, the two parties must exchange security credentials (either directly or indirectly). However, each party needs to determine if they can "trust" the asserted credentials of the other party.

WS-Trust defines extensions to WS-Security that provide:

- Methods for issuing, renewing, and validating security tokens.
- Ways to establish assess the presence of, and broker trust relationships.

Using these extensions, applications can engage in secure communication designed to work with the general Web services framework, including WSDL service descriptions, UDDI businessServices and bindingTemplates, and SOAP messages.

4.2.5.6.3 SAML (OASIS)

A particular method for encoding identity tokens in federated environments is the Security Assertion Markup Language Standard (SAML). SAML makes use of statements which assert certain characteristics of a subject (claims), e.g. a subject's authentication, name and role. SAML can be used to encode qualified identity tokens and may be combined with XML-Signature. Security Assertions Markup Language (SAML) is an XML standard from OASIS designed for exchanging security information. SAML is a very flexible specification and can be used in a multitude of scenarios. One of the problems that it tries to solve is the Single Sign On problem and in this way it can be used for authentication in scenarios. Because it addresses Single Sign On, SAML is ideal for federation scenarios where users come from a different security domain than the one where the service provider is. SAML denotes a various types of statements for expressing identity information, which all are referring to an included subject. The structure of an assertion object is shown in Figure 4-18. For additional details on Authentication Statement, Attribute Statement, and Subject Statement refer to OWS-4 GeoDRM Engineering Viewpoint and supporting Architecture (OGC 06-184).

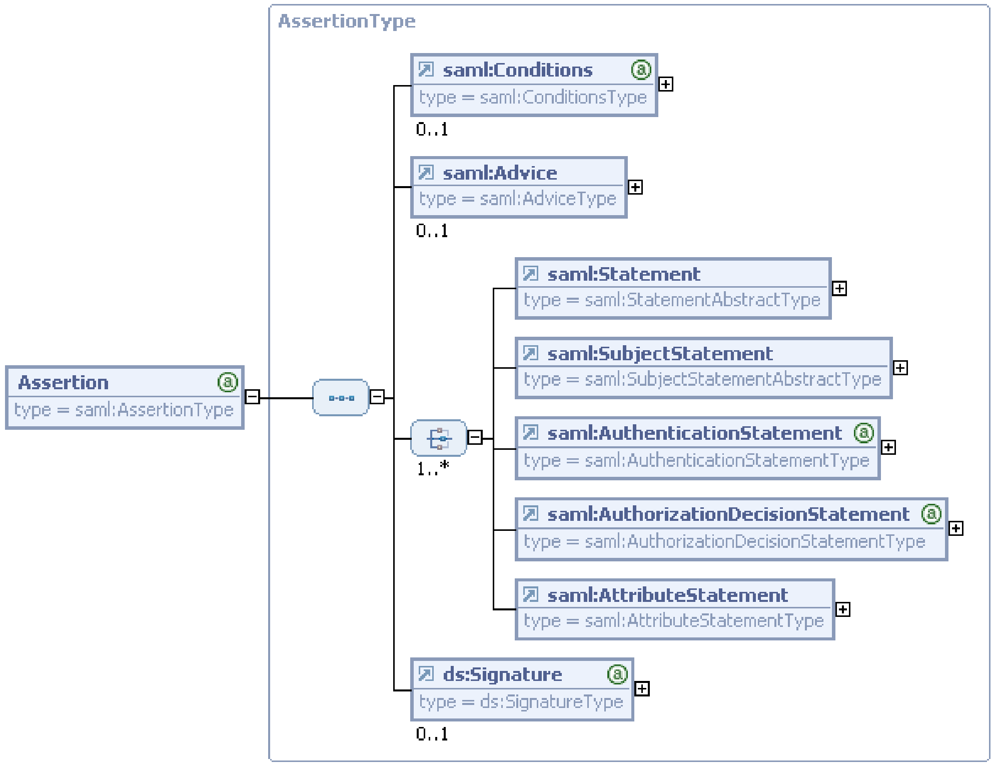


Figure 4-18. SAML Assertion

The integrity of identity tokens handled by various parties can be assured by XML digital signature, which is applied to an Assertion artifact containing one or more Statements. If the identity provider is known to the service provider, esp. PEP, PDP, the integrity of the identity information contained in an identity token can be verified.

4.2.5.6.4 GeoXACML

Geospatial eXtensible Access Control Markup Language (GeoXACML) defines an extension to XACML for spatial data types and spatial authorization decision functions. Those data types and functions can be used to define additional spatial constraints for XACML based policies. Beside a schema to encode policies, XACML includes a context schema that includes a specification of the generic data level interface to the XACML processor (PDP). Since GeoXACML defines extensions for the policy encoding schema only, it does not affect the XACML context schema and therefore does not include an interface specification of any kind. In addition, XACML includes a model for an access control system. This incorporates stereotype definitions of a Policy Information Point (PIP) also referred to as an Authentication Service, Policy Decision Point (PDP) or Authorization Service, Policy Administration Point (PAP) or License Manager, and a Policy Enforcement Point (PEP) or Gatekeeper, as well as their relations to each other in the context of an access control system.

GeoXACML defines an extension to the XACML Policy Language that supports the declaration and enforcement of access restrictions on geographic information. The geospatial extension to XACML is based on the extensibility points as allowed in the XACML standard. In short, GeoXACML defines:

- geometry model on which the geometric data types in access rules have to be based on,
- different encoding languages for geometric data types (which are provided in the extensions to this core specification),
- testing functions for topological relationships between geometries, and
- geometric functions.

The XACML Information Flow Model shown in Figure 4-19 defines the architecture of a modular and distributed access control system. In addition, it defines the exchange of messages between the components and the structure of the messages. The following figure illustrates the informative architecture and the sequence of messages, sent between the components of the access control system.

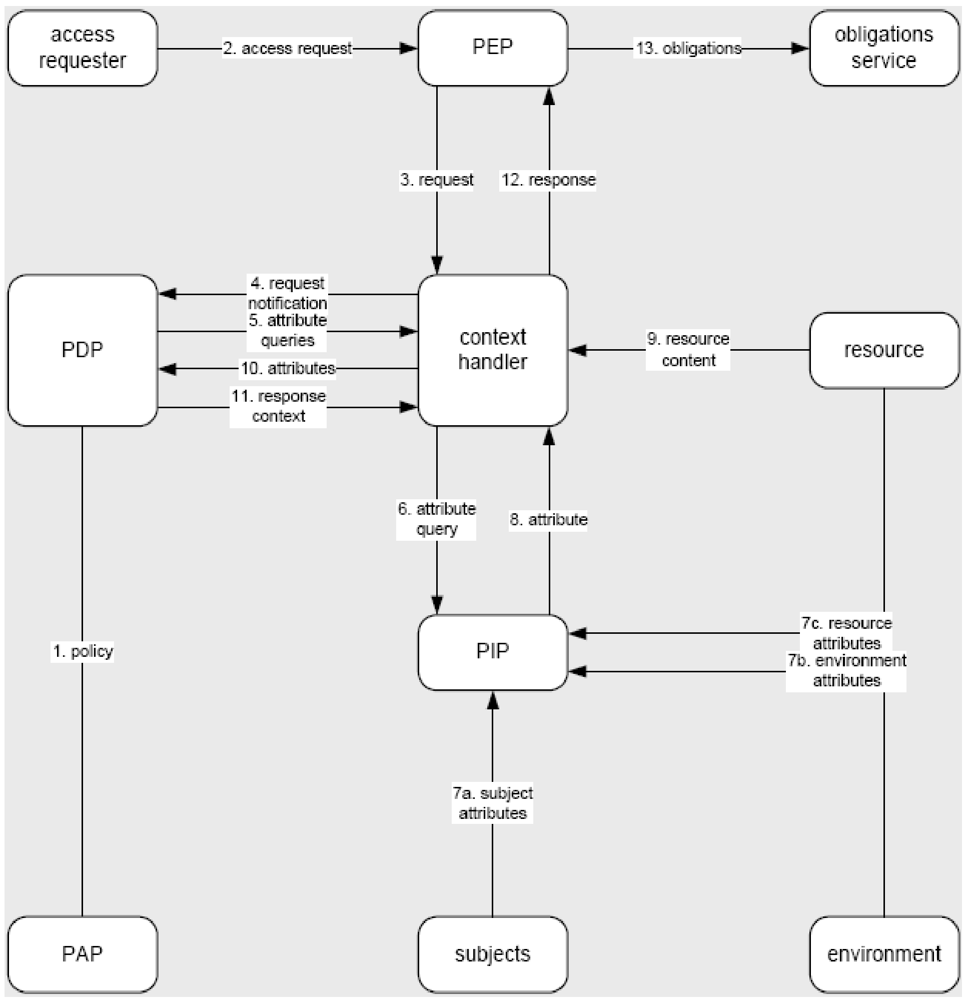


Figure 4-19. XACML Information Flow Model

4.2.6 GPW Computational Viewpoint

The Computational Viewpoint of the OGC Reference Model reflects the Components, Interfaces, Interactions and Constraints of the Service Architecture.

4.2.6.1 GPW Workflow Architecture

Relevant Documents:

- OWS-5 GPW Architecture Engineering Report. (OGC 07-138)
- OWS-4 Workflow IPR (OGC doc 06-187r1)

Workflow efforts in this thread will focus on design, implementation and demonstration of security capabilities in workflows that using WfMC's process model for workflow integration as well as OASIS's Web Services for Business Process Execution (WS-BPEL) technologies. The GPW thread aims to continue work started in previous testbeds to develop and demonstrate interoperability among geo-processes through publish-find-bind, service chaining and workflow orchestration. Workflow implemented through loosely-coupled integration of OGC web services in a service-oriented architecture is an objective of GPW. The results will be realized through scenarios that demonstrate interoperability and service-oriented architectures.

4.2.6.2 Security for OGC Web Services

Relevant Documents:

- OWS-4/5 GeoRM Architecture Security Analysis (presentation at Security DWG at TC-Potsdam
- NIST Special Pub 800-95, Guide to Secure Web Services, September 2006

The GeoRM architecture, as shown in Figure 4-20, has focused on addressing authentication, authorization and license management for OGC web services. To achieve a complete security solution within the architecture for OGC web services, additional security measures must be added as shown in the Information Viewpoint, section 4.2.5 above. The measures include capabilities for encryption and digital signatures.

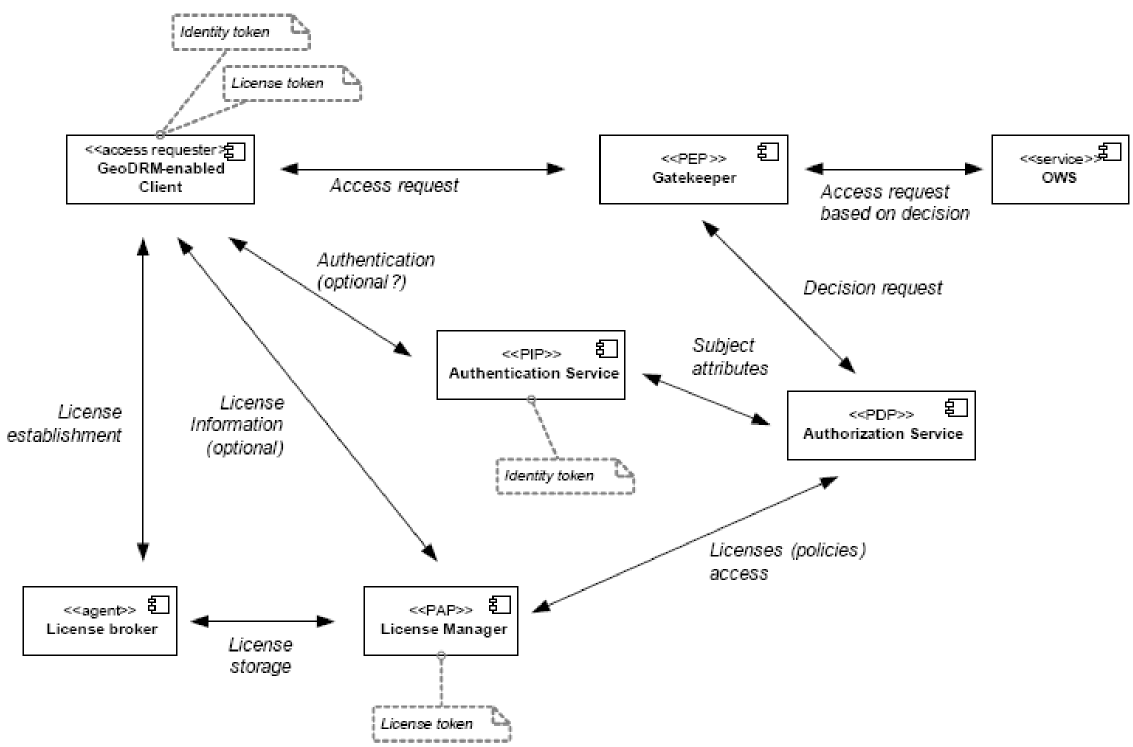


Figure 4-20. GeoRM Architecture

4.2.6.3 Wf-XML

The Workflow Management Coalition (WfMC)'s Workflow Standard – Interoperability Wf-XML Binding defines a XML language used to achieve a level of abstraction and independence from specific workflow engine implementations. The following diagram shows a representation of this integration approach which is based on the WfMC's Workflow Reference Model as shown in Figure 4-21.

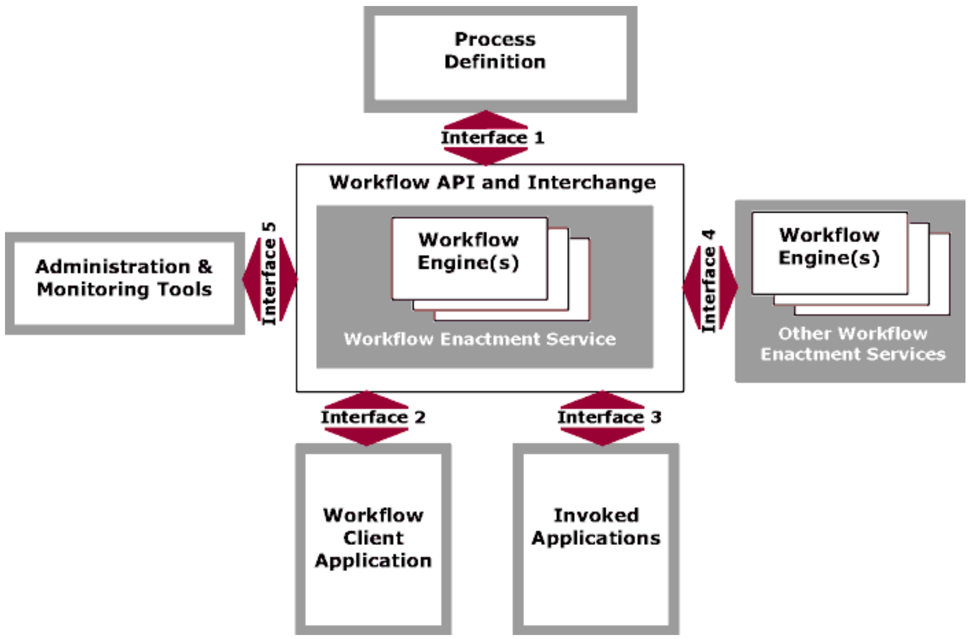


Figure 4-21. WfMC's Workflow Reference Model

In this diagram, Wf-XML supports the Interface 4. The goals of this standard are:

- Support chained, nested and parallel-synchronized models of interoperability
- Provide for both synchronous and asynchronous interactions
- Support individual and batch operations
- Remain implementation independent
- Define a light, easy-to-implement protocol

4.2.6.4 Web Processing Service (WPS)

Relevant Specifications:

- OpenGIS® Web Processing Service (WPS), OGC document 05-007r5

The Web Processing Service (WPS) is a standardized interface that facilitates the publishing of geospatial processes, and the discovery of and binding to those processes by clients. "Processes" include any algorithm, calculation or model that operates on spatially referenced data. "Publishing" means making available machine-readable binding information as well as human-readable metadata that allows service discovery and use. This interface specification provides mechanisms to identify the spatially referenced data required by the calculation, initiate the calculation, and manage the output from the calculation so that the client can access it.

WPS is a general purpose web service in that it does not identify any specific processes that are supported. WPS can be thought of as an abstract model of a web service, for which profiles need to be developed to support use, and standardized to support interoperability. As with other OGC specifications, it is the development, publication, and adoption of profiles which define the specific uses of this specification.

The WPS interface specifies three operations that can be requested by a client and performed by a WPS server, all mandatory implementation by all servers. Those operations are:

- GetCapabilities – This operation allows a client to request and receive back service metadata (or Capabilities) documents that describe the abilities of the specific server implementation. The GetCapabilities operation provides the names and general descriptions of each of the processes offered by a WPS instance. This operation also supports negotiation of the specification version being used for client-server interactions.
- DescribeProcess – This operation allows a client to request and receive back detailed information about the processes that can be run on the service instance, including the inputs required, their allowable formats, and the outputs that can be produced..
- Execute – This operation allows a client to run a specified process implemented by the WPS, using provided input parameter values and returning the outputs produced.

The WPS specification by itself allows service developers to reuse significant amounts of code in the development of web interfaces, while at the same time facilitating ease of understanding among web application developers. However, fully-automated interoperability can be achieved only through using standardized profiles. While it is possible to write a generic client for WPS, the use of a profile enables optimization of interoperable client user interface behaviour, as well as the publish/find/bind paradigm. To achieve high interoperability, each process shall be specified in an Application Profile of this specification.

A WPS Application Profile describes how WPS shall be configured to serve a process that is recognized by OGC. An Application Profile consists of

- An OGC URN that uniquely identifies the process (mandatory)
- A reference response to a DescribeProcess request for that process (mandatory).
- A human-readable document that describes the process and its implementation (optional, but recommended).
- A WSDL description (optional in the WPS specification, required in OWS-5).

WPS Application Profiles are intended for consumption by web service registries which maintain searchable metadata for multiple service instances.

4.2.6.5 Workflow Chaining Service (WfCS)

Relevant Documents:

- OWS-5 GPW Architecture Engineering Report. (OGC 07-138)
- OWS 2 Service Chaining with BPEL Discussion Paper (OGC 04-078)
- Web Services – Business Process Execution Language (WS-BPEL) 2.0

Previous testbeds have implemented Workflow Chaining Service (WfCS) using SOAP-BPEL and RESTful bindings for selected service integration patterns as described in the first two referenced documents listed above.

The Workflow Chaining Service (WfCS) 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.

As described in ISO 19119, there are many possible approaches to composing chains of processing services into aggregate or compound service components. General patterns can be used to describe these approaches based on, for example, the visibility of the services to the user (or client application) as well as the difference in how control of the services is managed. Using these criteria, the service chaining patterns are shown in Figure 4-22.

a. User defined (transparent) chaining: the client application manages the workflow and control of the chain is exclusively with the user of the client application.

b. Workflow-managed (translucent) chaining: in which the client application invokes a Workflow Management service that controls the chain and the user is aware of the individual services; a workflow service controls the chain execution, perhaps with oversight by the human user of the client application.

c. Aggregate service (opaque): in which the client application invokes a service that carries out the chain, with the user having no awareness of the individual services; the aggregate service exclusively performs the control function with no visibility by the client application.

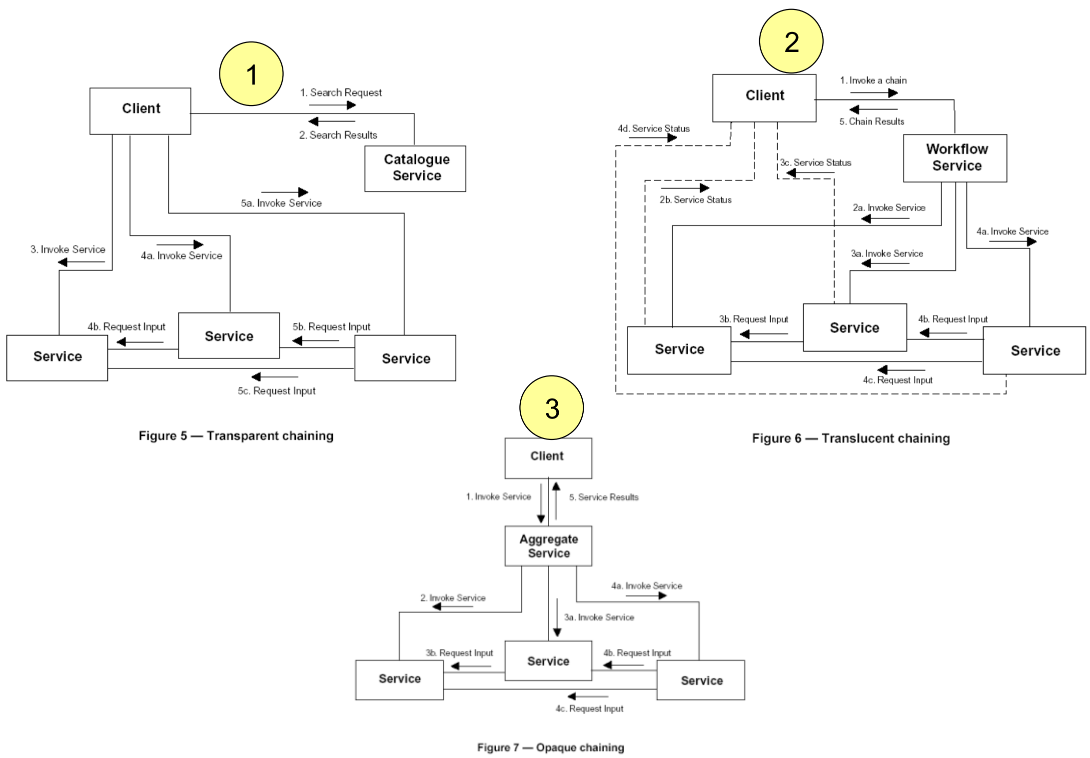


Figure 4-22. Service Chaining Patterns

4.2.6.6 Web Feature Service

Relevant Specifications:

- OpenGIS® Web Feature Service version 1.1

The purpose of the Web Feature Server Interface Specification (WFS) is to describe data manipulation operations on OpenGIS® Simple Features (feature instances) such that servers and clients can "communicate" at the feature level. In the GPW thread WFS will provide access to a variety of feature data including sponsor-provided Urban MSD data.

4.2.6.7 UML to GML Application Schema Processing and UGAS Tool

Relevant Specifications:

- OWS-5 GEOINT Schema Implementation Profile (GSIP) Schema Processing (OGC 08-078r1)
- OWS-5 Local MSD Implementation Profile for GML 3.2.1 (OGC 08-077)
- OWS-5 Data View Architecture (OGC 07-163)
- OWS-4 GEOINT Structure Implementation Profile (GSIP) Schema Processing (OGC 07-028r1)
- OWS-3 Schema Tailoring and Maintenance (OGC 05-117)
- OWS-2 Application Schema Development (OGC 04-100)

The UML to GML Application Schema (UGAS) tool, originally developed as part of the GOS-TP initiative, is used to facilitate creation of GML Application Schemas from information models expressed in UML. UGAS has been updated and enhanced during subsequent testbeds including OWS-2, OWS-3, OWS-4, and OWS-5. During OWS-6, UGAS will be used to develop GML application schemas based on OGC's Geography Markup Language Version 3.2.1 and OGC's CityGML to encode NSG data and serve as a transfer format among NSG participants.

In addition, enhancements will be developed for the UGAS Tool to:

- Use Object Constraint Language (OCL) and or Schematron constraints to perform "data content specification" profile validation;
- Use a variety of Dictionaries to be provided by the Sponsoring organizations.

4.2.6.8 Catalog ebRIM Profile

Relevant Documents:

- OGC Catalogue Service v2.0.2 (07-006r1)
- OGC CSW-ebRIM Registry Service - Part 1: ebRIM profile of CSW (07-110r2)
- OGC CSW-ebRIM Registry Service – Part 2: Basic extension package (07-144r2)
- ISO Metadata using ebRIM Profile Discussion Paper (07-038)
- ISO 19115 Geographic information — Metadata
- ISO 19139 Geographic Information – Metadata Schema Implementation

The OGC Catalogue Services v2.0.2 specification (OGC 07-006r1) establishes a framework for implementing catalogue services that can meet the needs of stakeholders in a wide variety of application domains. This application profile is based on ebXML RIM ISO/TS 15000-3.

4.2.7 GPW Engineering Viewpoint

This Engineering Viewpoint contains information on how services from the computational viewpoint are implemented. The component types interact based upon the services identified in the Computational Viewpoint. Figure 4-23 provides a summary of the component types organized consistent with a three-tier model.

- User Interfaces - The top tier is the only one with which clients (people or systems) deal directly. It provides the interfaces to describe and use the services offered;
- Business Processes - The middle tier embodies all the business processes required to respond to requests issued by clients. The services in general embody everything from authentication to complex geoprocessing on sets of data from various repositories and from generation of map views to statistical charts that the client gets back at the end of the process;
- Data Access - The lower tier provides read and/or write access to data, whether its geospatial data, accounting records, or catalogue entries stored in any of a dozen different types of registries.

To limit the complexity of the diagram, interactions between components are not made explicit in Figure 4-23.

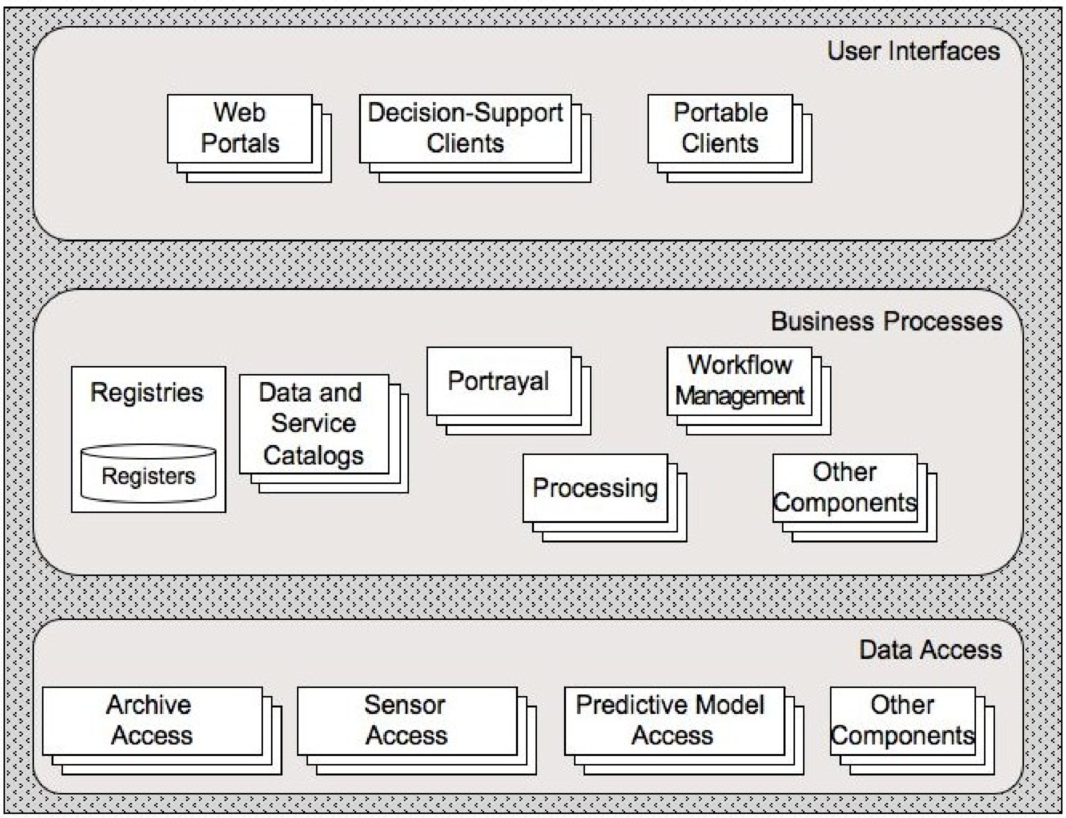


Figure 4-23. Engineering Viewpoint Components

An example workflow using the components is shown in Figure 4-24.

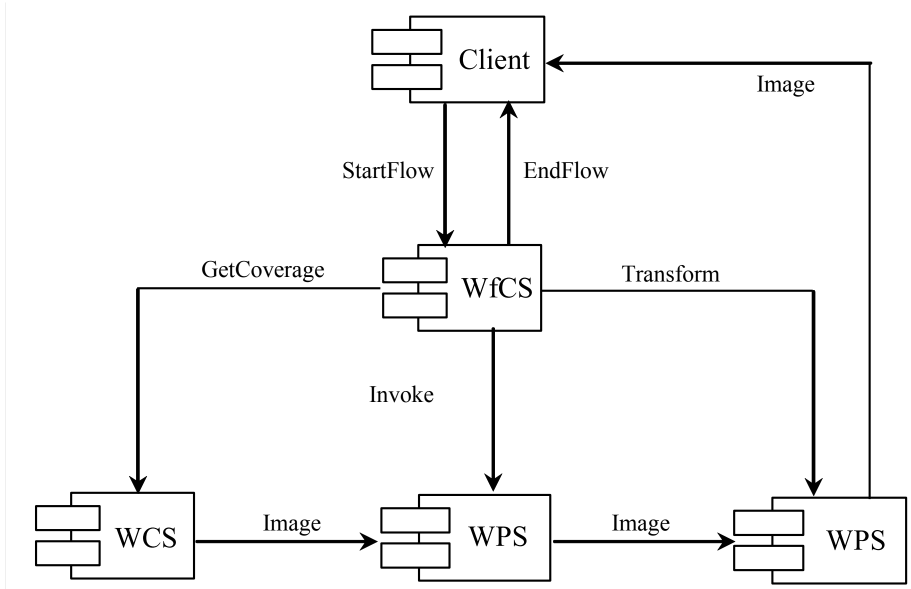


Figure 4-24. Example Geo-Processing Workflow

4.3 Aviation Information Management (AIM)

The Aviation Information Management (AIM) subtask is a new thread within OWS to develop and demonstrate the use of the Aeronautical Information Exchange Model (AIXM) in an OGC Web Services environment. The AIM subtask shall focus on evaluating and advancing various AIXM features in a realistic trans-Atlantic Aviation scenario setting by devising and prototyping a Web Services Architecture for providing valuable aeronautical and weather information directly to flight decks, Electronic Flight Bags (EFB) and hand-held devices (such as PDAs and Blackberries) while the airplane is at the gate or en-route to its destination (for the purposes of OWS-6, the aeronautical information in the latter case does not depend on the knowledge of the airplane's location). .

AIXM was developed by the Federal Aviation Administration (FAA) and Eurocontrol as a global standard for the representation and exchange of aeronautical information. It was designed as a basis for digital aeronautical information exchange and for enabling the transition to a net-centric, global aeronautical management capability. Both agencies seek to evaluate the potential of OGC Web Services and other information sharing technologies in conjunction with the net-centric System Wide Information Management (SWIM) concept by demonstrating and enhancing the use of models such as AIXM in a Web Services Environment. It is envisioned that the core principles of OGC Web Services and standards may be included in the overall logic of future interoperable ATM information sharing on the local, regional and global levels.

4.3.1 AIM Scope

The AIM thread focuses on providing up-to-date aeronautical and weather information to pilots and aircraft while at the airport gate or en-route to its destination by

- Providing access to integrated, rich and distributed aeronautical and weather information in a standard way, both on demand and automatically,
- Maximizing amount of useful information delivered to aircraft avionics, EFB or hand-held electronic display devices while at the gate or en-route to destination,
- Filtering information based on spatial and temporal criteria based on intended flight plan and current conditions.

To support these goals, the scope of the OWS-6 AIM thread is defined by the following areas of work:

- Use and enhancement of Web Feature Service and Filter Encoding specifications in support of AIXM features and 4-dimensional flight trajectory queries,
- Architecture of standards-based mechanism to alert/notify users of changes to user-selected aeronautical information,
- Prototype of Aviation client for retrieval, integration and visualization of AIXM, Weather and other aviation-related data, emphasizing time and spatial filtering in order to present just the right information into any given user context anytime, anywhere,

The AIM thread shall ensure that the integrity of aeronautical data (full or partial) is preserved throughout all data exchange operations by applying the work performed in the OWS-6 GPW thread (see Trusted to Trusted Domains Section 4.2.2.4.2).

4.3.2 AIM Requirements

4.3.2.1 AIXM 5.0 through Web Feature Service

The FAA and Eurocontrol have developed AIXM as a global standard for the representation and exchange of aeronautical information. AIXM was developed using the OGC Geography Markup Language (GML 3.2) tailored to the specific requirements for the representation of aeronautical objects, especially the temporality concept that allows for time dependent changes affecting AIXM features. In AIXM, every feature may have several "TimeSlices" that describe the temporal evolution of the feature properties. The temporality concept of AIXM 5.0 is detailed in the following document on the www.aixm.aero website: http://www.aixm.aero/gallery/content/public/release_candidate_3/AIXM%20Temporality%200%205.pdf.

A major objective of the AIM thread is to demonstrate the use of Web Feature Service (WFS) 1.1 and the Filter Encoding Specification 1.1 and to identify the necessary enhancements needed for these two specifications in order to distribute native AIXM 5.0 data and to provide 4D flight trajectory information filtering using government furnished geospatial data and location content including one or more of the following: airports, airspace, navigational aids and fixes, obstacles, and procedures (approaches)

- Demonstrate the provision through WFS of AIXM 5.0 baseline information (airports, airspace, navigational aids and fixes, obstacles and/procedures), temporary and permanent updates to an end-user on the airport surface,
- Exercise the temporal aspects of AIXM 5.0 and provide temporally-based spatial queries using WFS/Filter Encoding,
- Document required changes to WFS, Filter Encoding, AIXM 5.0 and/or GML Profiles as needed.

4.3.2.2 Integration of Weather Data/Evaluation of Weather Data Standards

The OWS-6 AIM thread shall investigate how weather data affecting a designated area within a given interval of time can be accessed via OGC Web Services and consumed by an Aviation client.

Towards that purpose, the thread shall first evaluate existing and evolving standards for modeling, encoding and transmitting weather data in order to identify the optimum source(s) from which useful weather data can be obtained and consumed in the testbed. The evaluation must take into account the maturity and planned evolution of surveyed weather standards, the availability of data encoded to each standard and the relationship between the standards (present and planned).

At a minimum, the evaluation shall address the following standards or sources of weather data:

- Weather Exchange Conceptual Model (WXCM)/Weather Information Exchange Model (WXXM) 1.0, which were developed according to the OGC standard "Observations and Measurements – Part 1 – Observation Schema". This task includes investigating the possibility of feeding back extensions and adaptations for aeronautical meteorological requirements and listing them for possible further adoption in subsequent releases of the relevant ISO/OGC specifications,
- Joint METOC Conceptual Data Model (JMCDM)/Joint METOC Broker Language (JMBL): JMBL is a specification for a standard XML that will broker the exchange of information between Meteorological and Oceanographic (METOC) data providers and user applications. It is well aligned with the FAA NextGen weather concept and is considered for development of the emerging WXCM/WXXM standard,
- National Weather Service (NWS) Digital Weather GML (dwGML): used to provide access to the NWS Digital Forecast Database (NDFD) via WFS,
- World Meteorological Organization (WMO) Information System: coordinated global infrastructure for the collection, distribution, retrieval and access to data and information of all WMO and

related programmes to ensure interoperability of Information Systems between WMO programmes as well as with outsides the WMO community,

- Network Common Data Form (NetCDF): machine-independent, self-describing, binary data format standard for exchanging scientific data, commonly used in climatology and meteorology (e.g. weather forecasting and climate change) and GIS applications.

More information on these weather formats and access methods can be found in the Information Viewpoint of the AIM thread. Depending on the outcomes of the evaluation, this thread may require the development of a capability to convert existing data from various sources to WXXM as applicable in the Aviation scenario.

4.3.2.3 Architecture of Change Alert/Notification Mechanism

The AIM thread shall develop and test a mechanism for notifying users of changes to user-selected aeronautical information using standards such as GeoRSS, and to provide those users with access to the information changes:

- Notify users of the most recent information for user-specified locations, or in relation to a planned flight, and then provide them with access to that information automatically or on user demand,
- Notify users when changes to selected data sets have been made to aeronautical data; the notification should provide a retrieval key,
- Notify users when aeronautical database modifications are applied by authoritative sources to selected sets of aeronautical information,
- Enable clients to use WFS 1.1 and any enhancements needed to retrieve AIXM 5.0 aeronautical information for which a change notification has been received, using the retrieval key provided in the notification, and apply the changes to the local copy of the data.

4.3.2.4 Retrieval and Visualization of AIXM and Weather Data in Aviation Client

In order to support the Aviation scenario and to encourage the adoption and use of AIXM 5.0 by the vendor community, the AIM thread also includes the development of one or more Aviation clients and/or client surrogates representing on-board avionics, Electronic Flight Bags (EFB) and hand-held electronic display devices.

Priority is given to the investigation of the filtered provision of relevant aeronautical and weather information subsets in relation to a planned flight and its gate-to-gate trajectory buffer. Such filtering uses time criteria and spatial constraints to extract the necessary information from weather, airspace/route and airport information sources. To support this goal, the Aviation Client shall

- Demonstrate the use of WFS 1.1 and Filter Encoding 1.1 (in addition to any enhancements identified in the testbed to support AIXM 5.0, including temporality and GML 3.2) to retrieve and visualize 3D government furnished geospatial data and location content including one or more of the following: airports, airspace, navigational aids and fixes, obstacles, and procedures (approaches),
- Connect to OGC Web Services (WFS, WCS, WMS and/or SOS) to obtain weather data affecting a designated area within a given interval of time, and display the results on the AIXM data visualization,
- Support change alert/notification mechanism (subscription to alerts or GeoRSS feeds) to be notified of and gain access to most recent information for user-specified locations, or in relation to a planned flight, automatically or on user demand,
- Support a trans-Atlantic flight scenario in order to demonstrate Global applicability.

4.3.3 AIM Deliverables

The OWS-6 AIM thread requires two types of deliverables

- **Engineering Reports and Documents:** shall be prepared in accordance with OGC published templates as defined on the OGC portal. Engineering Reports shall be delivered by posting on the OGC Portal Pending Documents list when complete and the document has achieved a satisfactory level of consensus among interested participants, contributors and editors. Engineering Reports are the formal mechanism used to deliver results of the Interoperability Program to sponsors and to the OGC Specification Program Domain and Specification Working Groups for consideration. All documents created in response to this program shall include "OWS-6" in the title, to facilitate later literature searches.

- **Services, Clients and Tools:** shall be provided by methods suitable to its type and stated requirements. For example, services and components (ex. WFS) are delivered by deployment of the service or component for use in the testbed via an accessible URL. A Client software application may be used during the testbed to exercise services and components to test and demonstrate interoperability; however, it is most often not delivered as a license for follow-on usage.

Note that certain draft deliverables will be required by the Interim Milestone at the date shown in the Master Schedule (Main Body, Section 4.6), for use in cross-thread development. These early deliverables will be designated and handled on a thread-by-thread basis.

4.3.3.1 Engineering Reports (ERs) and Documents

The following Engineering Reports (ERs) shall be developed in the Aviation thread and may be submitted to the OGC Specification Program at the completion of the OWS-6 Testbed.

1) OWS-6 Aviation Information Management ER covering a) Detailed technical architecture of the OGC Web Services for the Aviation scenario b) Refined Aviation scenario and use cases c) Documentation of the rationale for the selection of the weather source(s) and the weather encoding standard(s) used in the AIM thread d) Lessons learned and recommendations for the use of AIXM/WFS/Filter Encoding in support of AIM objectives
2) OWS-6 AIM Change requests for extensions and adaptations, as needed, to a) AIXM 5.0 b) WXCM/WXXM 1.0 c) GML 3.2 Profiles d) WFS (1.1 and 1.2) /Filter Encoding, e) Web Notification Service f) Weather standards
3) OWS-6 AIXM Schemas

4.3.3.2 Services, Clients and Tools

Implementations of the following services, clients and data instances shall be developed in the AIM thread for integration and interoperability testing and may be invoked for cross-thread scenarios for OWS-6 demonstration events:

1)	Alert/Notification/GeoRSS service to notify the user of changes made to aeronautical data.
2)	WFS with Filter Encoding including enhancements needed to support AIXM 5.0 (based on AIXM 5.0 GML 3.2.1 aspects, WFS 1.2 draft/change requests and support for temporal queries) serving AIXM schemas and instance documents relevant to the Aviation Scenario and to the client components.
3)	AIXM instance documents containing data relevant to the Aviation Scenario, including airports, airspace, navigational aids and fixes, obstacles, and/or procedures (approaches).
4)	Electronic Flight Bag (EFB) Class 2 and/or Class 3 client and/or hand-held electronic display devices component (or their surrogates); Class 2 EFBs are Portable Electronic Devices that can be mounted on board the aircraft while Class 3 EFBs are fixed, installed equipment that are integrated with the Avionics system.
5)	OGC Web Service providing access to Aviation weather data (specific OGC Web Service to be determined based on weather data standard/source used in the Aviation scenario).

4.3.4 AIM Enterprise Viewpoint

According to the FAA web site (http://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=8807), the air transportation system is stretched thin with forecasts indicating increases in passenger demand ranging from a factor of two to three by 2025. The current system is already straining with ever-increasing levels of congestion, declining on-time arrivals, increasing delays (and customer frustration) as well as increasing costs and environmental impacts.

To address these global problems, both the US's Federal Aviation Administration (FAA) and Europe's Eurocontrol are working on ATM modernization expanding capabilities to maintain the safety and efficiency of global aviation. This involves working towards a vision of an evolving global Aeronautical Information Management paradigm (Figure 4-25) built extensively on standards, digital data exchange and process automation to provide amongst others

- End to end management of information,
- Support for a common operating picture of system collecting and feeding the right information at the right time at the right place,
- Improved predictability through better information integration,
- Value to customers for using information in flight planning, navigation, rerouting and setting adaptation to various tools.

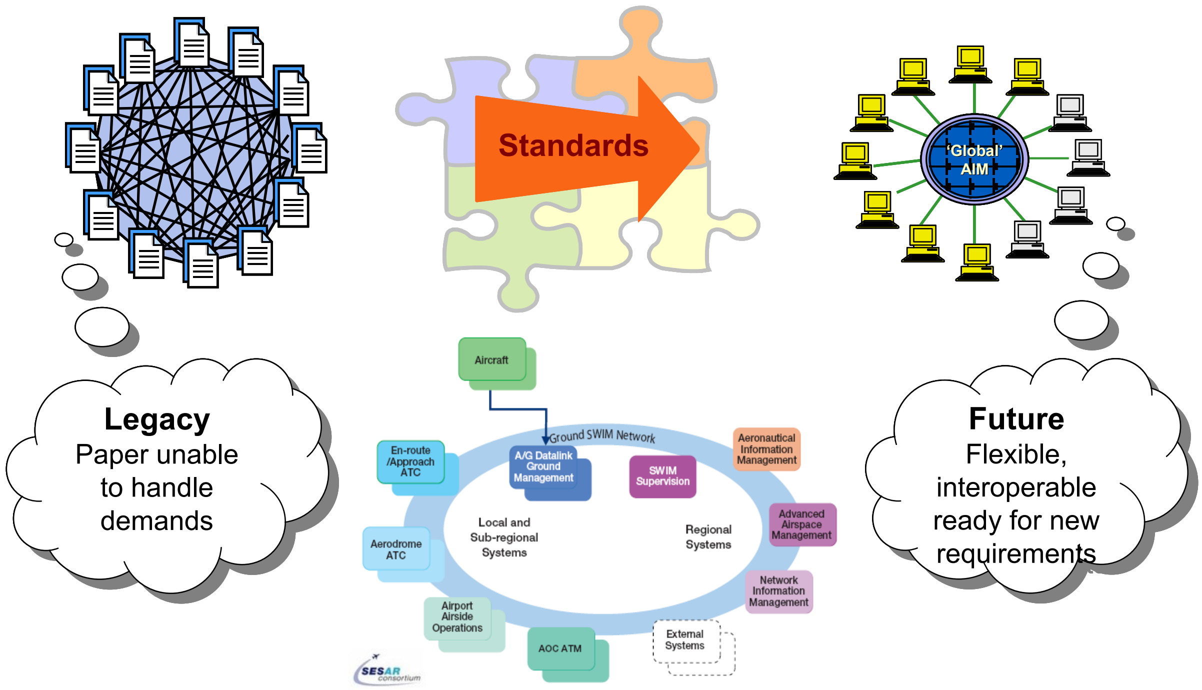


Figure 4-25. Towards a New Aeronautical Information Management Paradigm

At the heart of this new AIM paradigm is the Aeronautical Information Exchange Model (AIXM) (Figure 4-26). AIXM is a comprehensive aeronautical information content and exchange model developed by Eurocontrol in the 1990s and then considerably expanded and modernized through the collaboration of the FAA and NGA and others (Figure 4-27).

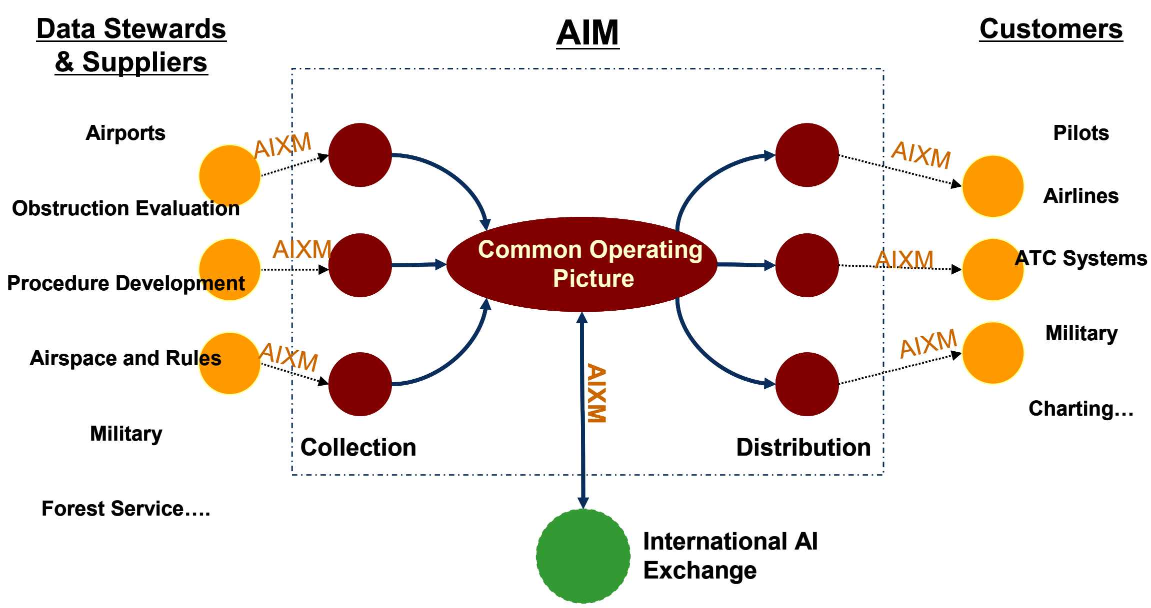


Figure 4-26. AIXM in Support of New AIM Paradigm

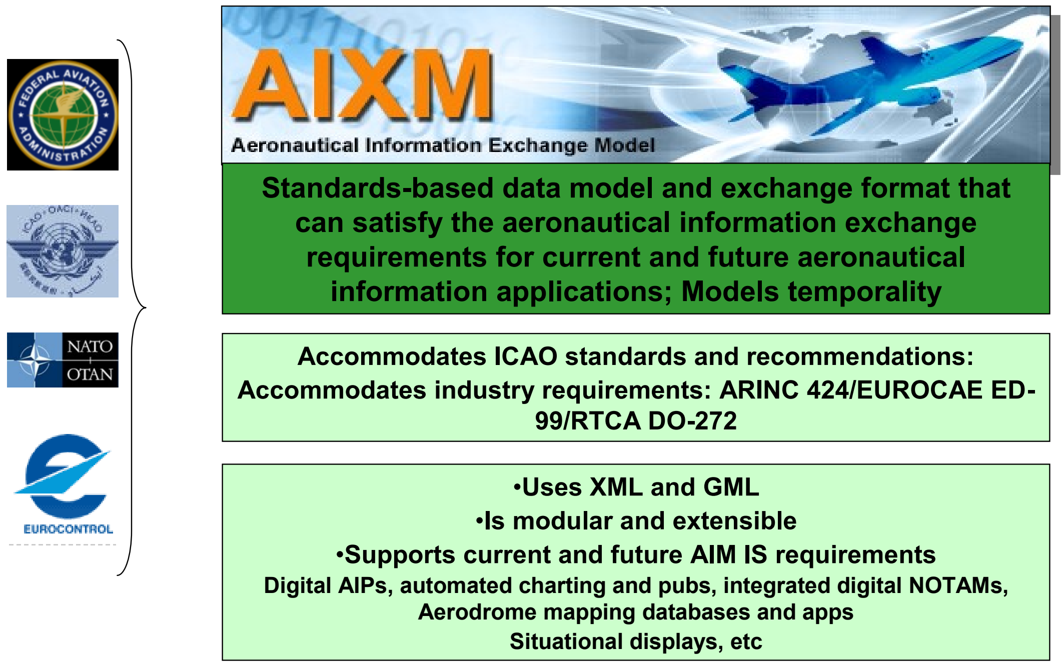


Figure 4-27. AIXM Overview

The FAA plans to use AIXM in conjunction with the FAA Next Generation Air Transport System (NextGen) and the net-centric System Wide Information Management (SWIM), described in the next sections. Eurocontrol is also seeking to advance the use of geo-spatial technologies for AIM SWIM concepts in the OWS-6 testbed to serve their objective of developing a uniform pan-European Air Traffic Management (ATM), embodied in the concept of a Single European Sky (as described in section 4.3.4.3). Previous versions of AIXM (3.3 and 4.5) are already used operationally in Europe. The new version AIXM 5.0 is required to support the concept of “digital NOTAM”. The AIM enterprise viewpoint also introduces the Data Link Operational Service and Environment Definition (OSED; section 4.3.4.4) which is based on the NextGen and SESAR initiatives.

4.3.4.1 FAA Next Generation Air Transport System (NextGen)

NextGen encompasses the operational and technological changes needed to increase the US National Airspace System (NAS) capacity, to meet future demands and avoid gridlock in the sky and in the airports (http://www.faa.gov/regulations_policies/reauthorization/). NextGen requires improved common situational awareness, integration of air traffic management and control, consistent use of weather data and forecasts for flight planning and better coordination of responses to adverse conditions. The FAA is a key participant in the US Joint Program Development Office (JPDO) which is a multi-agency initiative overseeing the evolution of NextGen concepts.

AIXM provides the foundation for NextGen (Figure 4-28). Upon that foundation rests many of the next generation operational improvements including on-demand NAS information, continuous flight day evaluations, trajectory-based management, constrained flight planning, collaborative air traffic management and reduced weather impacts amongst others.

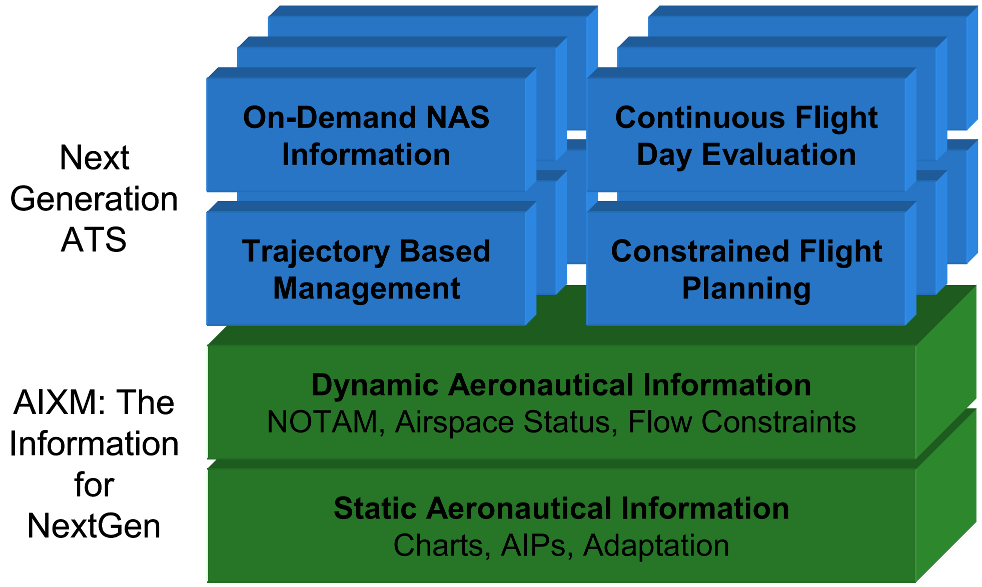


Figure 4-28. AIXM as the Foundation for NextGen

A key element of NextGen is the NextGen Network Enabled Weather (NNEW) which will serve as the core of the NextGen weather support services and provide a common weather picture across the NAS. These services will, in turn, be integrated into other key components of NextGen required to enable better air-transportation decision-making. It is anticipated that tens of thousands of global weather observations and sensor reports from ground-, airborne-, and space-based sources would fuse into a single national weather information system, updated as needed in real-time. The goal of NNEW (in conjunction with other NextGen technologies) is to cut weather-related delays at least in half since currently seventy percent of NAS delays are attributed to weather every year.

4.3.4.2 System Wide Information Management (SWIM)

System Wide Information management (SWIM) provides the infrastructure and services to deliver network-enabled information access across the NextGen air transportation operations. SWIM will provide high quality, timely data to many users and applications – extending beyond the previous focus on unique, point-to-point interfaces for application-to-application data exchange (Figure 4-29). By reducing the number and types of interfaces and systems, SWIM is expected to reduce redundancy of information and better facilitate multi-agency information-sharing. SWIM will also enable new modes of decision-making, as information is more easily accessed by all stakeholders affected by operational decisions.

SWIM will be built on an open, flexible, and secure information management architecture for sharing NAS advisory data and enabling increased common situational awareness and improved NAS agility. SWIM will use commercial off-the-shelf hardware and software to support a loosely coupled service-oriented architecture that allows for easier addition of new systems and connections. More information about SWIM can be found at http://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/swim/.

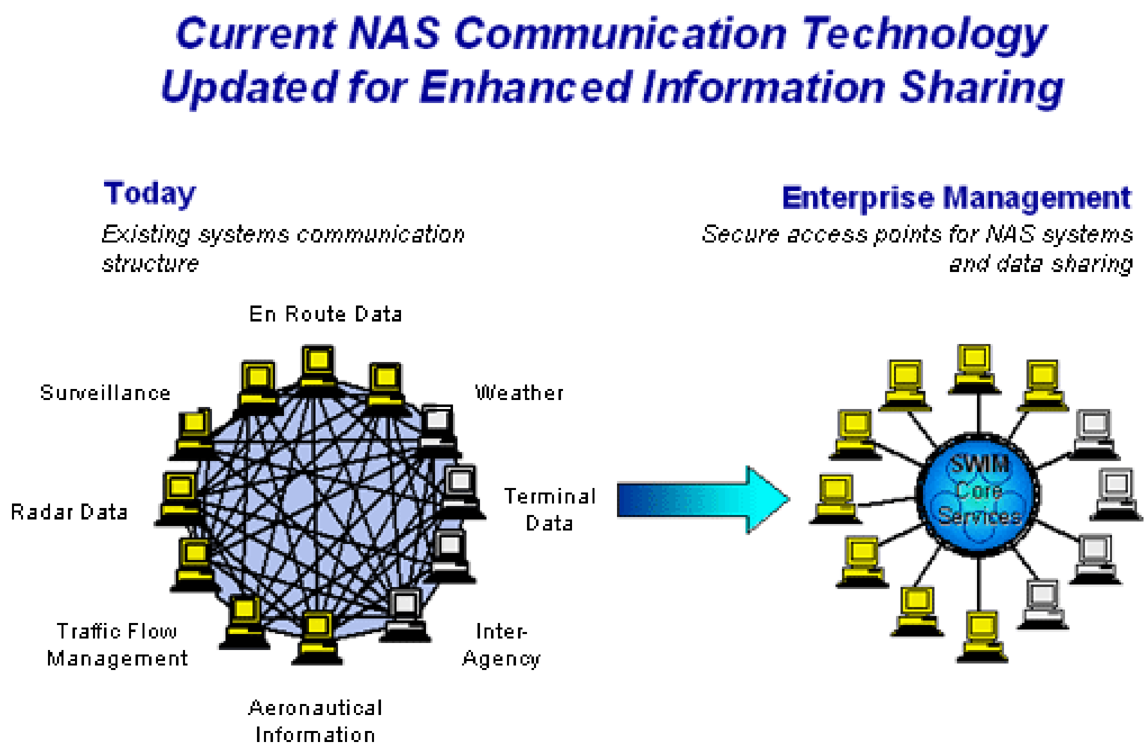


Figure 4-29. SWIM for Enhanced Information Sharing

4.3.4.3 Single European Sky ATM Research (SESAR)

The European Airspace is fragmented and will become more and more congested, as traffic is forecast to grow steadily over the next 15 years. The Air navigation services and their supporting systems are not fully integrated and are based on technologies which are already running at maximum. In order to accommodate future Air Traffic needs, a “paradigm shift”, supported by state-of-the-art and innovative technologies, is required. That paradigm shift is realized via the Single European Sky ATM Research (SESAR) initiative.

SESAR is a Joint European Commission/Eurocontrol initiative (currently in the Development phase according to Figure 4-30) that targets the elimination of the fragmented approach to ATM, the transformation of the European ATM system and the synchronization of plans and actions of the different partners and federated resources:

http://www.eurocontrol.int/sesar/public/subsite_homepage/homepage.html

For the first time in European ATM history, an ATM improvement programme is involving the Aviation Players (civil and military, legislators, industry, operators, users, ground and airborne) for defining, committing to and implementing a pan-European programme, and to support the Single European Sky legislation.

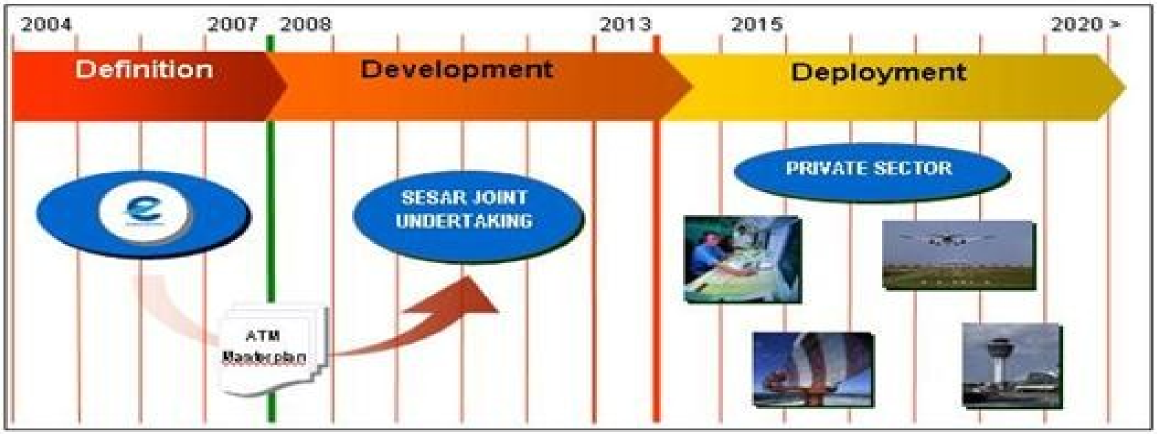


Figure 4-30. SESAR Phases

Aeronautical Information Management (AIM) is one component of the future Air Traffic Management/Communications, Navigation, Surveillance (ATM/CNS) Overall Target Architecture that includes an overall information architecture based on the concept of SWIM (Figure 4-31). The SWIM concept is required for building the net-centric environment and enterprise architecture (a light-weight, massively distributed, horizontally applied architecture that distributes components and/or services across an enterprise's information value chain using internet technologies and other network protocols as the principal mechanism for supporting the distribution and processing of information services.

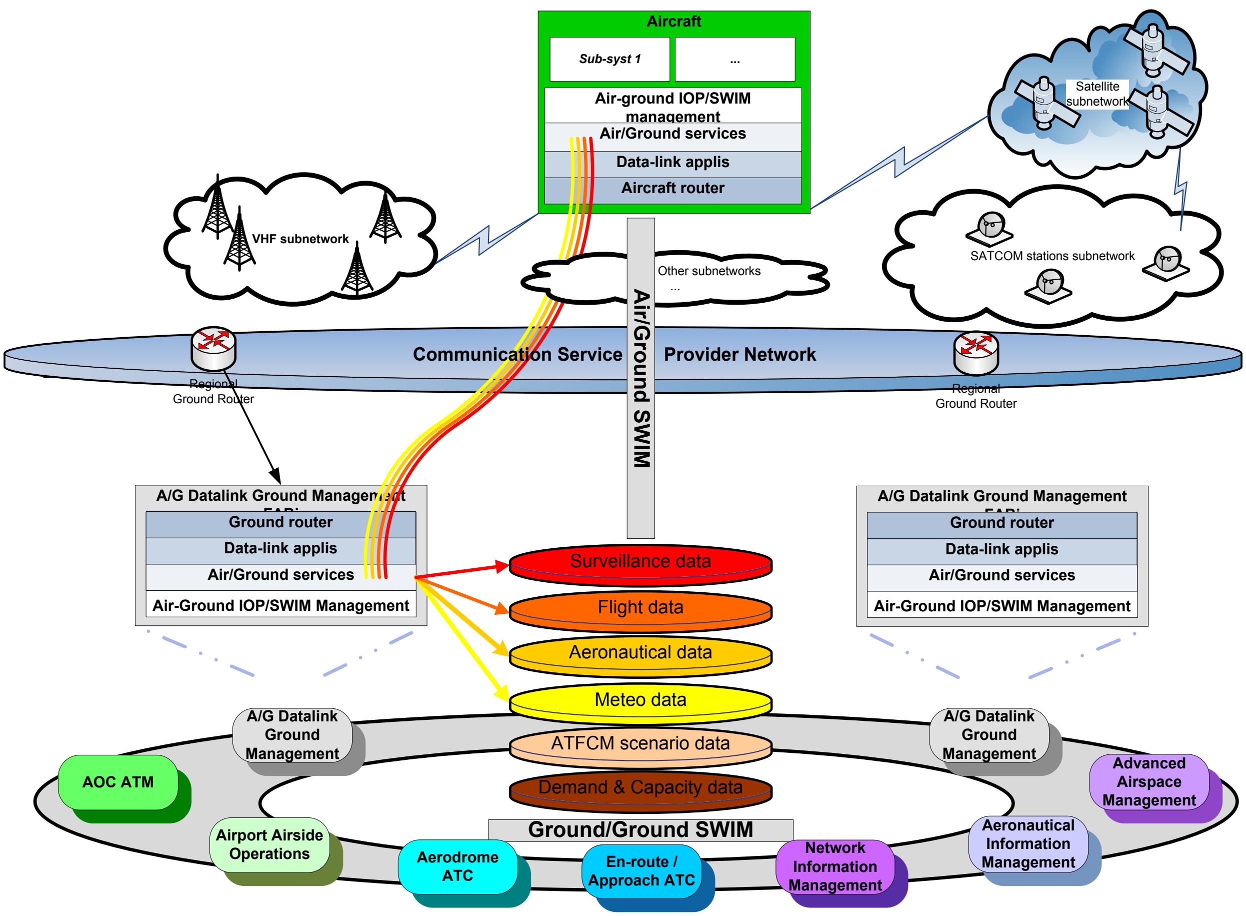
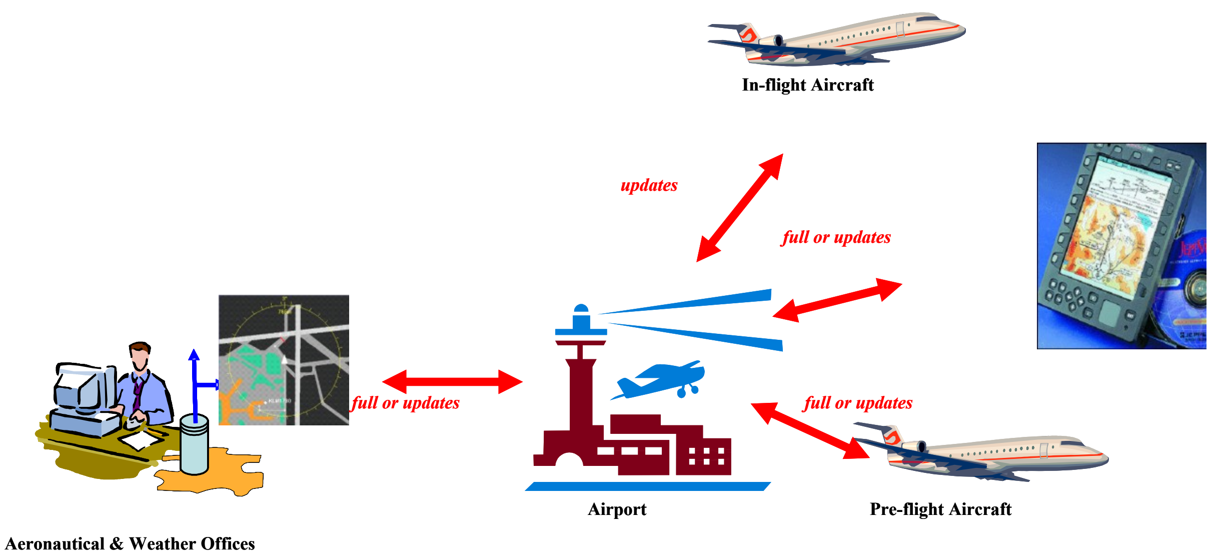


Figure 4-31. SWIM Concept in SESAR

4.3.4.4 Data Link Operational Service and Environment Definition (OSED)

The operational concepts in the Data Link Operational Service and Environment Definition (OSED) are based on the NextGen and SESAR initiatives, and are defined by the Joint RCTA/EuroCAE committee on Aeronautical Information Services (AIS) and Meteorological (MET) Datalink. Those concepts (Figure 4-32) include

- AIS and MET information recognized as being important components of the ATM system,
- Providing cockpit access to current MET and AIS info for more active aircrew participation in collaborative decision making,
- Moving from a product-centric methodology to a data-centric and Service Oriented Architecture (SOA) approach.



Derived from the RTCA/EUROCAE joint committee on Aeronautical Information and Weather Datalink.

Figure 4-32. Data Link Operational Service and Environment Definition

4.3.4.5 AIM Scenario and Use Cases

This scenario provides a fictitious, but realistic context for a demonstration of the functionality that will be developed in the AIM thread of the OGC OWS-6 and for the interaction with other OWS components. It is intended to prompt the exercising of interfaces and the use of encodings that will be developed or enhanced within OWS-6. One major objective will be to demonstrate the ability of Web Feature Services (WFS) and the Filter Encoding (FE) Specification to distribute aeronautical data in AIXM 5.0 format in response to direct user queries or in response to alerts to a user when specific aeronautical information – as defined by that user – is updated. An operational context for demonstrating the retrieval of pertinent weather information and delivering it to the aircraft is also provided. The participants in this scenario are the flight crew and a provider of aeronautical information updates.

Note: The airports denoted by ***TBD1-TBD5*** *will have to be determined by various factors, notably availability of AIXM data and weather data. Flight times may also be determined by available weather data.*

Scenario

ZZ Flight uvw is scheduled to depart from Airport **TBD1** for Airport **TBD2**. After the pilot receives the flight briefing from Airline Operations Control (AOC) flight operations, a ground delay due to adverse weather or other event (e.g. approaching wildfire) on the first segment of the route is put into effect for **ZZ Flight uvw** which remains at the gate. The pilot enters the aircraft three hours after his briefing and requests an automatic feed to his EFB or hand-held electronic display device of any updates to aeronautical information for **TBD2,** for **TBD3** - an alternate airport close to the filed route of flight to be used in an emergency occurring during the initial segment of the flight, and for **TBD4 –** an alternate airport to be used in the event that a landing at **TBD2** cannot be made. It is assumed that, while on the ground, the aircraft is connected to a high-speed TCP/IP network. For the purpose of the test bed, the public Internet will be used. However, it is likely that such applications will use a non-public communication network. Shortly after requesting the automatic feed, updated aeronautical information for **TBD3** is displayed on the EFB or hand-held electronic display device indicating that **TBD3** is closed due to bad weather or other event (e.g. approaching wildfire). The update information is super-imposed on the display of the route as a highlight at the affected location. The highlight is expandable to show the update in either text or graphic format.

The pilot enters a request on the EFB or hand-held electronic display device for convective weather, turbulence and lightning in an area that will affect the planned route and altitude of flight in the next three hours. The requested information is received and displayed on the EFB or hand-held electronic display device with the forecast weather events superimposed on the map. The display shows a strong weather system that will affect the filed route within the times specified. The pilot requests AOC flight operations and Air Traffic Control (ATC) (by voice) for a new route and is given an amended route to **TBD2** that will avoid the weather system.

The pilot enters a request on the EFB or hand-held electronic display device to retrieve and display information on airports within 100 nautical miles of the amended route of flight at which his aircraft will be able to land in the event of an emergency occurring while the aircraft is within 500 nautical miles from **TBD1**. The location of Airport **TBD5**, is returned and is displayed on the EFB or hand-held electronic display device, superimposed on a visualization of the amended route of flight. The pilot enters a request for detailed aeronautical information, including approaches, on **TBD5**. The information is super-imposed on the display of the route as a highlight at the affected location. The highlight is expandable to show the update in either text or graphic format. . The pilot estimates the time of his closest approach to **TBD5** and requests forecast weather information for **TBD5** for a possible landing at that time. The requested information is displayed on the EFB or hand-held electronic display device with the forecast weather events superimposed on the map. The display shows no significant weather forecast for **TBD5** so the pilot requests an automatic feed to the EFB or hand-held electronic display device of any updates to aeronautical information for **TBD5** and cancels the automatic feed for **TBD3**. Shortly after this **TBD3** is re-opened and no alert is received by the EFB or hand-held electronic display device.

The ground controller at **TBD1** airport clears **Flight uvw** to taxi for departure. As the aircraft taxis the EFB or hand-held electronic display device is disconnected from the internet (to alleviate this disconnect, a WFS could possibly be stood up to provide the Taxi path to the pilot based on the D-Taxi Graph standard which is currently <u>under development</u> by EUROCAE WG78 RCTA SC214 – Standards for air traffic data communication services). After take-off the pilot requests an automatic feed to the EFB or hand-held electronic display device of any updates to aeronautical information for **TBD2, TBD4** and **TBD5.** During the flight, whenever update information is available for these airports, an audible alert and a visual indication is given on the display of the EFB or hand-held electronic display device. An alert is received for the destination airport, TBD2, and the pilot requests display of the update information for that airport, which is then displayed on the EFB or hand-held electronic display device. The update information is super-imposed on the display of the route as a highlight at the affected location. The highlight is expandable to show the update in either text or graphic format.

The updated information displayed for **TBD2** shows that all the runways at **TBD2** are closed. (Fires in close proximity to the airport are producing extensive smoke over the airport.) The pilot obtains permission from ATC to divert to **TBD4.** During this time, no alerts had been received for TBD4. The pilot requests an automatic feed to the EFB or hand-held electronic display device of any weather advisories or METARS in effect for **TBD4.** The responses to this request are displayed on the EFB or hand-held electronic display device as the flight continues and indicates that there is no significant weather at **TBD4**. **ZZ uvw** then lands without incident**.**

4.3.4.5.1 Use Case #1: Access to Aeronautical and Weather Data and Updates by Aviation Client at the Gate

Use Case Id:	AIM #1	Use Case Name:	Access to AIM and weather data at the gate
Use Case Domain:	OWS-6 AIM	Status:	Draft 2008-07-18
Use Case Description:	This use case describes the process of the Aviation client requesting and retrieving aeronautical and weather data and/or updates during the pre-flight stage of operation		
Actors (Initiators):	Flight crew, AOC flight operations, ATC, Ground controllers, Aeronautical Information custodians/providers, Weather information custodians/providers	Actors (Receivers)	Flight crew
Pre-Conditions: - Airplane connected to the Internet - Security and integrity mechanisms in place - Aeronautical and weather information available via trusted OGC Web Services		Post-Conditions: - Requested aeronautical and weather information have been successfully retrieved, integrated and displayed in the Aviation client for use by the flight crew in preparation for take-off	
System Components - One or more WFS serving aeronautical data - One or more OWS serving weather data - Aviation client			

Use Case Id:	AIM #1	Use Case Name:	Access to AIM and weather data at the gate
Basic Course of Action			

1. Pilot downloads information about origin (TBD1) and destination airport (TBD2) from Aeronautical information custodians/providers.
2. After 3 hours (due to a delay), pilot subscribes to an automatic feed to the EFB of aeronautical information updates for 3 airports: destination airport (TBD2) , alternate to destination airport (TBD4), and airport close to filed route of flight in case of an emergency during the initial flight segment (TBD3).
    a. Pilot receives updates on TBD3 that it is closed due to bad weather or other event. The update information is superimposed on the display of the route as a highlight at the affected location. The highlight is expandable to show the update in either text or graphic format.
3. Pilot requests convective weather, turbulence and lightning information from weather data custodians/providers for the area affecting the planned route and altitude of flight for the next 3 hours.
    a. Pilot receives requested information displayed on EFB along with forecast weather events superimposed on the map.
4. Pilot requests new route from ATC based on weather information.
    a. Pilot receives by voice amended route to TBD2 that will avoid the weather system.
5. Pilot requests information on airports within 100 nautical miles of amended route of flight at which the aircraft will be able to land in the event of an emergency occurring while the aircraft is within 500 nautical miles from TBD1
    a. Pilot receives location of an alternate airport TBD5 superimposed on the visualization of the amended route of flight
    b. Pilot requests and receives detailed aeronautical information (including approaches) on TBD5 superimposed on the display of the route as a highlight of the affected location. The highlight is expandable to show the update in either text or graphic format.
    c. Pilot requests forecast weather information for TBD5 for possible landing at a given time. No significant weather events are predicted.
6. Pilot subscribes to automatic feed to the Aviation client of updates to aeronautical information for TBD5 and cancels the subscription related to TBD3 (When TBD3 re-opens, no alert is received by the pilot because the subscription to that event was canceled).

4.3.4.5.2 Use Case #2: Access to Aeronautical and Weather Data and Updates by Aviation Client En-Route

<table>
<tr><td>Use Case Id:</td><td>AIM #2</td><td>Use Case Name:</td><td colspan="2">Access to AIM and weather data en-route to destination</td></tr>
<tr><td>Use Case Domain:</td><td colspan="2">OWS-6 AIM</td><td>Status:</td><td>Draft 2008-07-18</td></tr>
<tr><td>Use Case Description:</td><td colspan="4">This use case describes the process of the Aviation client requesting and retrieving aeronautical and weather data and/or updates during the flight to the destination airport.</td></tr>
<tr><td>Actors (Initiators):</td><td>Flight crew, AOC flight operations, ATC, Ground controllers, Aeronautical Information custodians/providers, Weather information custodians/providers</td><td>Actors (Receivers)</td><td colspan="2">Flight crew</td></tr>
<tr><td colspan="2">Pre-Conditions:
- Airplane connected to the Internet
- Security and integrity mechanisms in place
- Aeronautical and weather information available via trusted OGC Web Services</td><td colspan="3">Post-Conditions:
- Requested aeronautical and weather information have been successfully retrieved, integrated and displayed in the Aviation client for use by the flight crew during the flight.</td></tr>
<tr><td colspan="5">System Components
- One or more WFS serving aeronautical data
- One or more OWS serving weather data
- Aviation client</td></tr>
<tr><td colspan="5">Basic Course of Action
1. After take-off, pilot subscribes to automatic feed to aeronautical information updates for airports TBD2, TBD4 and TBD5.
a. During the flight, whenever updates are available, an audible alert and a visual indication are generated.
b. An alert for TBD2 (destination airport) is received.
2. Pilot requests display of the update information.
a. Update superimposed on the map shows that all runways at TBD2 are closed.
b. Pilot obtains permission from ATC to divert to TBD4.
3. Pilot subscribes to automatic feed of weather advisories or METARS in effect for TBD4.
a. Data returned shows that there is no significant weather at TBD4.</td></tr>
</table>

4.3.5 AIM Information Viewpoint

The Information Viewpoint describes the information models and encodings that will make up the content of the services and exchanges to be extended or developed to support the AIM thread activities in the following areas

- Aeronautical and Weather models and encodings (AIXM 5.0, GML 3.2, WXCM/WXXM, JMBL, NNEW, NetCDF, dwGML).
- Information filtering, integration and sharing (Filter Encoding 1.1, OWS Context, OGC KML, GeoRSS).

4.3.5.1 AIXM 5.0

***Relevant Specifications**:*

– AIXM 5.0 (http://www.aixm.aero)

AIXM is designed to enable the management and distribution of Aeronautical Information Services (AIS) data in digital format. It leverages existing and emerging information engineering standards and supports current and future aeronautical information system requirement. The major tenets are

- An exhaustive temporality model, including support for the temporary information contain in NOTAM (Notice to Airmen), Aeronautical features have a start of life and end of life, and can change over time as well. The AIXM temporality model is used to describe when features are valid and when feature properties change over time,
- Alignment with ISO standards for geospatial information (Figure 4-33), including the use of the Geography Markup Language (GML 3.2),
- Support for the latest ICAO and user requirements for aeronautical data including obstacles, terminal procedures and airport mapping databases,
- Modularity and extensibility to support current and future aeronautical information messaging requirements and additional data attributing requirements.

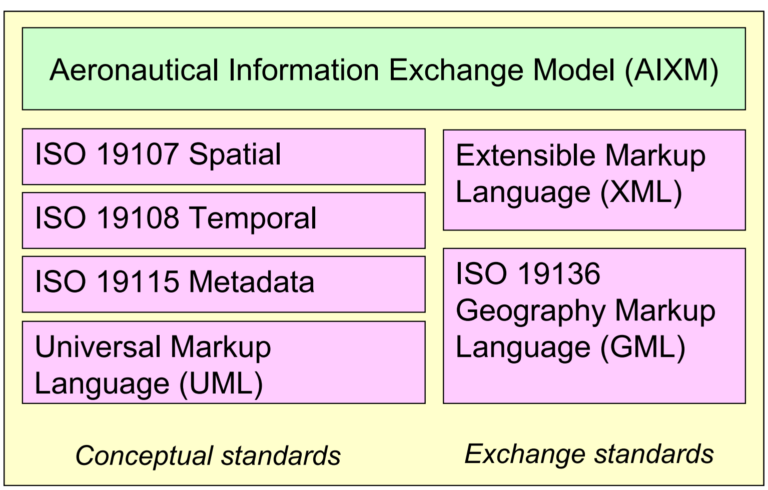


Figure 4-33. AIXM Based on International Standards

AIXM 5.0 is divided into 11 conceptual areas (Aerodrome/Heliport, Airspace, Holding, Navaids and Points, Obstacles, Organizations, Procedures, Refueling Aerial, Routes, Services, and Surveillance) and 4 shared components (Geometry, Notes, Time Management, Aircraft). Those areas are not independent from each other and are related via a high number of associations that cross the boundaries of each area.

4.3.5.2 GML 3.2

***Relevant Specifications**:*

– OpenGIS® Geography Markup Language (GML) Encoding Specification 3.2.1 (OGC pending document #07-036) (http://portal.opengeospatial.org/files/?artifact_id=20509&version=1)

The Geography Markup Language (GML) is an XML encoding for the transport and storage of geographic information, including both the geometry and properties of geographic features.

4.3.5.3 WXCM/WXXM

***Relevant Documents**:*

– Weather Exchange Information Model WXXM (http://www.eurocontrol.int/aim/public/standard_page/met_wie.html)

The WXXM is part of a family of platform (technology) independent, harmonized and interoperable information exchange models designed to cover the information needs of ATM. The first public release of WXXM occurred in May 2007. Later that year, the first draft release of the overarching WXCM was built. Both models were developed according to the OGC standard "Observations and Measurements – Part 1 – Observation Schema".

4.3.5.4 JMCDM/JMBL

***Relevant Documents**:*

– Joint METOC Public Data Administration Website (http://www.cffc.navy.mil/metoc/)

JMCDM is a logical data model which integrates the geophysical data requirements of all DoD components. The JMCDM and its supporting encyclopedia are a subset of the DoD Enterprise Data Model. JMBL is a specification for a standard XML that will broker the exchange of information between Meteorological and Oceanographic (METOC) data providers and user applications. It is well aligned with the NextGen weather concept and is considered for development of the emerging WXCM/WXXM standard.

4.3.5.5 NextGen NNEW

***Relevant Documents**:*

– Operational Evolution Partnership Plan Reference Sheet NNEW Website (http://www.faa.gov/about/office_org/headquarters_offices/ato/publications/oep/version1/reference/nnew/)

The NextGen Network Enabled Weather (NNEW) serves as the core of the NextGen weather support services and provides network access to weather information from distributed weather information sources (e.g. General Weather Processor) by all users; and fusion and integration of continually updated weather information into applicable NextGen decision support systems.

4.3.5.6 NetCDF

***Relevant Specifications**:*

– NetCDF 4.0 (http://www.unidata.ucar.edu/software/netcdf/)

The Network Common Data Form (NetCDF) is a machine-independent, self-describing, binary data format standard for exchanging scientific data, commonly used in climatology and meteorology (e.g. weather forecasting and climate change) and GIS applications.

4.3.5.7 dwGML

Relevant Documents:

– Digital Weather Geography Markup Language (dwGML) GML 2.1.2 schema (http://www.weather.gov/forecasts/xml/OGC_services/ndfdOWSserver.php?SERVICE=WFS&Request=DescribeFeatureType&VERSION=1.0.0&TYPENAME=Forecast_Gml2Point,Forecast_Gml2AllWx)
– dwGML GML 3.1.1 schema (http://www.weather.gov/forecasts/xml/OGC_services/ndfdOWSserver.php?SERVICE=WFS&Request=DescribeFeatureType&VERSION=1.1.0&TYPENAME=Forecast_GmlsfPoint,Forecast_GmlObs,NdfdMultiPointCoverage)

The National Digital Forecast Database (NDFD) Web Feature Service (WFS) provides access to the National Weather Service's (NWS) digital forecast database Responses from the WFS are returned in dwGML format (GML 2.1.2-based application schema as well as GML 3.1.1-based application schema).

4.3.5.8 Filter Encoding

Relevant Specifications:

– OGC Filter Encoding Implementation Specification 1.1 (http://portal.opengeospatial.org/files/?artifact_id=8340&version=4)

The OGC Filter Encoding Implementation Specification describes an XML encoding of the OGC Common Catalog Query Language (CQL) as a system neutral representation of a query predicate. The filter encoding is a common component used by a number of OGC Web Services (e.g. WFS) requiring the ability to query objects from a web-accessible repository.

Filter Encoding 1.1 may need to be enhanced in the AIM thread to better support AIXM 5.0 queries and 4D trajectory information filtering (support for GML 3.2 and queries involving TimeSlices).

Relevant Documents:

– Temporal Recommendations v 0.0.9 Discussion Paper (http://portal.opengeospatial.org/files/?artifact_id=14898&version=1)
– OWS-4 WFS Temporal Investigation v 0.1.0 Interoperability Program Report (http://portal.opengeospatial.org/files/?artifact_id=19580&version=2)

Both documents include recommended changes to Filter Encoding 1.1 (and WFS 1.1) to better support temporal requirements.

4.3.5.9 OWS Context

Relevant Documents:

- OGC Web Services Context Documents (OWS Context 0.2) Interoperability Experiment: FINAL REPORT (http://portal.opengeospatial.org/files/?artifact_id=12015&version=1)

OWS Context document is an XML encoding that references remote and/or local OGC Web Services. OWS Context documents are related to, but more powerful than, Web Map Context Documents (05-005). The latter are limited to referencing OGC Web Map Services (WMS), whereas the former can reference other OGC Web Services such as Web Feature Services (WFS), and Web Coverage Services (WCS).

The Aviation client shall support the OWS context as the basis for integrating, visualizing and sharing AIM's aeronautical and weather information.

4.3.5.10 OGC KML

Relevant Specifications:

- OGC KML 2.2 (http://www.opengeospatial.org/standards/kml/)

OGC KML is an XML-based encoding schema for expressing geographic annotation and visualization on existing or future web-based online maps and Earth browsers (that includes non only the presentation of graphical data on the globe, but also the control of the user's navigation in the sense of where to go and where to look).

4.3.5.11 GeoRSS

Relevant Documents:

- Geographically Encoded Objects for RSS Feeds GeoRSS Webpage (http://georss.org)

GeoRSS describes a number of ways to encode location in RSS feeds. GeoRSS-Simple supports basic geometries (point, line, box, polygon) and covers the typical use cases when encoding locations. GeoRSS-GML is a formal GML Application Profile and supports a greater range of features, notably coordinate reference systems other than WGS-84. Both formats are designed for use with Atom 1.0, RSS 2.0 and RSS 1.0.

The AIM thread shall leverage GeoRSS in the process of communicating information about aeronautical data changes or availability to AIM users.

4.3.6 AIM Computational Viewpoint

The computational viewpoint is concerned with the functional decomposition of the AIM architecture into a set of services that interact at interfaces. It reflects the components, interfaces, interactions and constraints of the Service Architecture without regard to their distribution. For the AIM thread, those services are

- Web Feature Service (WFS) for access to aeronautical and weather information,
- Sensor Observation Service (SOS) and Web Coverage Service (WCS) for access to weather information (depending on standard encoding and data source used in the AIM scenario),
- Notification Services (OASIS Web Services Notification) or OGC Web Notification Service (WNS) as alternatives for supporting the change alert mechanism described in section 4.3.2.3,
- GeoRSS Service for access to GeoRSS feeds published by data custodians in the event of information change or update.

4.3.6.1 Web Feature Service

Relevant Specifications:

- OpenGIS® Web Feature Service version 1.1 (http://portal.opengeospatial.org/files/?artifact_id=8339)

The WFS Implementation Specification allows a client to retrieve and update geospatial data encoded in Geography Markup Language (GML) from multiple Web Feature Services. It defines interfaces for data access and manipulation operations on geographic features, using http as the distributed computing platform. A Transactional WFS allows creation, deletion and updating of features in addition to querying and retrieval of features.

WFS 1.1 shall be used in the AIM thread to serve and query AIXM 5.0 aeronautical features including airports, airspace, navigational aids and fixes, obstacles, and/or procedures (approaches). WFS 1.1 shall be enhanced in this thread to better support AIXM 5.0 access and 4D trajectory information filtering (support for GML 3.2 and temporal queries involving TimeSlices). Such enhancements shall take into account the recommendations captured in the documents below as well as the draft WFS 1.2 specification.

Relevant Documents:
– Temporal Recommendations v 0.0.9 Discussion Paper
(http://portal.opengeospatial.org/files/?artifact_id=14898&version=1)
– OWS-4 WFS Temporal Investigation v 0.1.0 Interoperability Program Report
(http://portal.opengeospatial.org/files/?artifact_id=19580&version=2)

Both documents include recommended changes to WFS 1.1 (and Filter Encoding 1.1) to better support temporal requirements.

4.3.6.2 Sensor Observation Service

Relevant Specifications:

- OpenGIS® Sensor Observation Service specification v 1.0.0
(https://portal.opengeospatial.org/files/?artifact_id=26667)

The Sensor Observation Service (SOS) provides access to observations from sensors and sensor systems in a standard way that is consistent for all sensor systems including remote, in-situ, fixed and mobile sensors. SOS provides information about the sensor itself encoded in SensorML and observation data encoded in Observation and Measurement (O&M).

The SOS may be exercised in the AIM thread to provide access to weather data (depending on the weather data standard and source recommended and available for use in the AIM scenario).

4.3.6.3 Web Coverage Service

Relevant Specifications:

- OpenGIS® Web Coverage Service specification v 1.1.2
(http://portal.opengeospatial.org/files/?artifact_id=27297)

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.

The WCS may be exercised in the AIM thread to provide access to gridded weather data (depending on the weather data standard and source recommended for use in the AIM scenario).

4.3.6.4 OASIS Web Services Notification

Relevant Specifications:

- OASIS WNS Base Notification 1.3
(http://www.oasis-open.org/committees/download.php/20625/wsn-ws_base_notification-1.3-spec-os.pdf)
- OASIS WNS Brokered Notification 1.3
(http://www.oasis-open.org/committees/download.php/20626/wsn-ws_brokered_notification-1.3-spec-os.pdf)

The purpose of the OASIS Web Services Notification is to define a set of specifications that standardize the way Web services interact using "Notifications" or "Events". They form the foundation for Event Driven Architectures built using Web services. These specifications provide a standardized way for a Web service, or other entity, to disseminate information to a set of other Web services, without having to have prior knowledge of these other Web Services. They can be thought of as defining "Publish/Subscribe for Web services".

The WS-Notification family of specifications defines a standard Web services approach to notification. This document is the base specification on which all the other specifications in the family depend. It defines the normative Web services interfaces for two of the important roles in the notification pattern, namely the NotificationProducer and NotificationConsumer roles. This specification includes standard message exchanges to be implemented by service providers that wish to act in these roles, along with operational requirements expected of them.

The OASIS WS-Notification family of specification may be used in the AIM thread to support the mechanism of notifying clients of data change or availability.

4.3.6.5 Web Notification Service

Relevant Documents:

- Web Notification Service Best Practices Paper 0.0.9
(http://portal.opengeospatial.org/files/?artifact_id=18776)

The Web Notification Service (WNS) supports asynchronous service handling. WNS instances forward incoming messages on various transport protocols (e.g. email, HTTP, Instant Messaging protocols, phone, etc.) to clients. The service interface allows clients to register a target address and protocol which will be used to deliver messages from calling services that need to inform clients about specific events. The WNS may be used in the AIM thread to support the mechanism of notifying clients of data change or availability.

4.3.7 AIM Engineering Viewpoint

The Enterprise, Information, and Computation viewpoints describe a system in terms of its purposes, its content, and its functions. The Engineering viewpoint relates these to specific components linked by a communications network.

The Engineering viewpoint identifies component types in order to support distributed interaction between the components of the system. Those components interact based upon the services identified and described in the Computational viewpoint. Figure 4-34 provides an overview of the components of the AIM thread and their interactions.

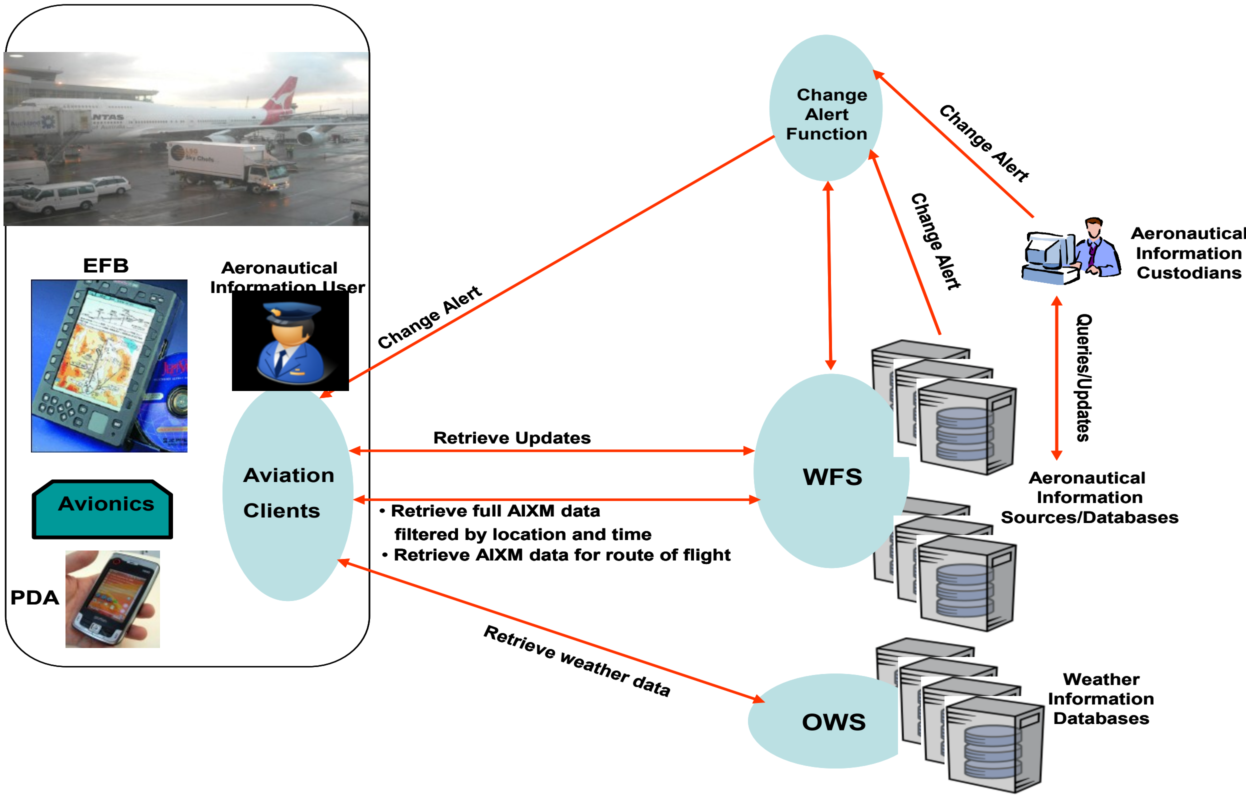


Figure 4-34. AIM Engineering Viewpoint

Figure 4-34introduces three types of Aviation clients (or their surrogates as needed for the OWS-6 AIM thread) namely:

- Avionics: electronic systems for use on aircraft comprising communications, navigation, monitoring, flight-control, collision-avoidance, weather, display and management of multiple systems.
- Electronic Flight Bags (EFB): electronic information management devices intended primarily for cockpit/flightdeck or cabin use. They help flight crews perform flight management tasks more easily and efficiently with less paper. EFB devices (which include COTS or modified COTS laptops and handheld devices) can display a variety of aviation and weather data or perform basic calculations (including performance data and fuel calculations). There are three classes of EFBs:
    - Class 1 EFBs are Portable Electronic Devices (PED) which are stowed and not normally used during take-off and landing operations, and do not require an administrative process to remove them from the aircraft.
    - Class 2 EFBs are PEDs that are normally mounted in a position where they're utilized during all phases of flight and require an administrative process to remove/replace from the aircraft.
    - Class 3 EFBs are fixed, installed equipment, are integrated with the aircraft avionics systems and hence require installation design approval.
- Electronic hand held display devices (such as PDAs or Blackberries) that can access the Internet and issue requests for Web Services.

The Aviation clients shall interact with the aeronautical and weather data sources/services via the service interfaces described in the Computational viewpoint.

4.4 Geospatial Decision Support Services (DSS)

Decision Support Services having an emphasis on applications of geospatial and temporal information has been a recurring thread in previous OWS testbeds. This thread focuses on presenting and interacting with data obtained from the sensor web and geoprocessing workflows in the most effective ways to support analysis and decision making.

It is expected that the following specifications and reports will be used in this initiative:

- OGC 04-094 – Web Feature Service
- OGC 06-042 – Web Map Service
- OGC 05-016 – Open Location Services (OpenLS)
- OGC 05-005, 08-050 – Web Map Context Implementation Specification, and Corrigendum
- OGC 05-062 – OWS Context IE Final Report
- OGC 05-116 – OWS Integrated Client GeoDSS Client
- OGC 02-070, Annex F – SLD (first version of spec), Annex on harmonizing SLD and ISO 19117
- OGC 05-078r4 – SLD Profile of WMS; with 05-077r2, supersedes 02-070
- OGC 07-123r1 – Corrigendum for SLD Profile of WMS 05-078r4
- OGC 05-077r2 – Symbology Encoding (SE); with 05-078r4, supersedes 02-070
- ISO 19117:2005 – Portrayal
- IHO S-52 Colour & Symbol Specifications for ECDIS, Edition 4.3, January 2008
- OWS-3 IPR 05-112r1 – Symbology Management
- OWS-3 IPR 05-110 – Feature Portrayal Service
- OWS-4 IPR 06-140 – Feature Styling
- OWS-4 IPR 06-187r1 – Workflow descriptions and lessons learned
- OWS-5 ER 07-124r2 – KML Engineering Report
- OWS-5 CR 08-064 – SLD Profile of WMS Change Request
- OGC 08-007r1 – CityGML Candidate Implementation Specification

4.4.1 DSS Scope

The focus for DSS in OWS-6 builds on portrayal, WMS Tiling, and integrated client work from OWS-3, OWS-4 and OWS-5, with additional work on 3D visualization and integration of the built environment and landscape.

This thread will encompass these capabilities and task areas:

- ISO 19117 and OGC SLD Portrayal
- 3D Portrayal of GML with Fly-through
- Outdoor and indoor 3D route services
- WMS performance (tiling)
- Integrated Client for multiple OWS services

4.4.2 DSS Requirements for OWS-6

4.4.2.1 ISO 19117 and OGC SLD Portrayal

ISO TC211 has opened a New Work Item Proposal (NWIP) to revise ISO 19117:2005 Portrayal, primarily to enhance the symbol specification information, and to improve the potential for interoperability of external functions. Related to this NWIP, the ISO 19117 Project Team anticipates that OGC will consider edits to the existing OGC portrayal specifications to make the consistent with the revised ISO 19117. This will establish a clear separation between the abstract and implementation specifications for portrayal as separate standards. The revised ISO 19117 Committee Draft is expected January 2009, and the Draft International Standard version by May 2009. The work in this thread will be coordinated with, and inform, these efforts to the extent possible.

Data portrayal requirements are often complex and largely based on feature attribution as opposed to simply feature types. In evaluation of OGC Styled Layered Descriptors for symbology encoding a question of the adequacy for SLD as it is written today to satisfy complex portrayal requirements has arisen. OWS-6 has a requirement for the analysis of SLD to address complex symbology requirements. A possible merging or integration of ISO 19117 v.2.0 and SLD is required in order to satisfy complex symbology requirements. Evaluate and document recommendations and issues as part of a ISO 19117 / SLD Harmonization Engineering Report.

Specific issues with ISO 19117, Styled Layer Descriptor (SLD) and Symbol Encoding (SE) include:

- Portrayal rules in ISO 19117 are currently defined only at the level of feature types; it is not possible to make symbolization conditional on feature attribute values.
- ISO 19117 does not support map composition by layers, as in the OGC Styled Layer Descriptor (SLD) specification. This means that in following ISO 19117, the client application cannot control z-order of layers relative to each other, or any feature selection filters.
- ISO 19117 does allow the use of external functions to perform related operations that affect symbolization, while SLD does not.
- Neither ISO 19117 nor SLD/SE have an approach for handling portrayal subtypes, that is, rules for handling subtypes of features differently from each other. This is important for supporting the concept of Data Content Specifications (DCS) which are profiles of a NGA Application Schema.
- SLD and SE do not specify storing portrayal specifications and portrayal rules in a manner consistent with ISO 19117. As described in OGC 02-070, Annex F, the first version of SLD specification, portrayal specifications (SE) should be referenced by URL from within portrayal rules stored in a portrayal catalogue. This separation of portrayal rules and symbolization from the feature data is the essential requirement.
- The current version of SLD/SE does not fully meet the portrayal requirements of the International Hydrographic Office (IHO) and the Digital Geographic Information Working Group (DGIWG) in supporting some of the S52 symbology for maritime features. Part of this has to do with supporting complex feature construction rules, and part has to do with the high resolution requirements of S52 symbology. A report[3] developed by Tenet Technology Ltd of UK presents some of the issues between OGC SE, ISO 19117, and IHO S-52 in detail. This report is available here: http://portal.opengeospatial.org/files/?artifact_id=29160.

---

[3] Neil Kirk (2008), TENET Report: Some Unresolved Issues with the OGC Symbology Encoding (SE), UTP1146-01, 18 July 2008 Draft, Tenet Technology Ltd,, West Sussex, UK.

The portrayal work item will continue to extend the portrayal work started in OWS-3 and OWS-4. The OWS-3 work is described in IPR 05-112r1 Symbology Management, which uses a Catalog to store, search and retrieve in the format described in 05-077r2 Symbology Encoding. This will also build on the work described in OWS-3 IPR 05-110 Feature Portrayal Service, OWS-4 IPR 06-140 Feature Styling, and OWS-5 ER 08-064 Component WMS Interface. In OWS-6, this thread will also consider SLD Schema and Symbology Encoding for local topographic Data Content Specifications (DCS). The first version of SLD (OGC 02-070), Annex F (pages 104-106) describes specific conditions which should be met to achieve and test harmonization of SLD and ISO 19117. A WMS will be required to implement the revised SLD/SE documents using GML data.

4.4.2.2 3D Portrayal of GML with Fly-through

OWS-6 DSS will develop approaches and implementations of 3D portrayal of GML features. This is a cross-thread requirement for both AIM and GPW. OWS-6 GPW is concerned with development of an NSG Application Schema (NAS) for urban Data Content Specifications (DCS) based on OGC CityGML. OWS-6 AIM is concerned with 3D AIXM which is a GML Application Schema, and with CityGML for airport and surrounding environment visualization. The DSS thread is particularly concerned with using CityGML data accessible via WFS. The DSS Thread will develop methods for 3D visualization of GML including routing and fly-through. The visualization will range seamlessly from the landscape scale to the interior of buildings. The client for this 3D visualization must be available in a web browser, which may include a plug-in to the browser.

In order to meet these objectives this thread will consider two main portrayal approaches and technologies for 3D features: Web Perspective View Services (WPVS) and Web 3D Services (W3DS). WPVS takes the approach of performing all the feature styling at the server, using best available hardware acceleration, then publishing a simple and efficient image of the result that can work with any level of client sophistication. This approach positions WPVS much like an enhanced form of WMS. W3DS, on the other hand, is specifically designed around using X3D and a web browser for the client, and not requiring that styling be carried out on the server. The CityGML specification (OGC 08-007r1) specifically mentions X3D support, but both WPVS and W3DS approaches have pros and cons. It will also be interesting to chain web services related to feature portrayal, e.g., WFS / CityGML. Symbolizers for 3D geometry elements (SE), and positioning of 3D layer elements (SLD) are also needed. SLD may also be useful in defining portrayal rules for CityGML Level of Detail 0 and 1 (landscape and extruded building scales). KML 2.2 could provide guidance for 3D layer orientation options. When considering extensions to SLD and/or SE specifications to support 3D, the issues related to harmonization with ISO 19117 mentioned in section 4.4.2.1 above must be considered.

4.4.2.3 Outdoor and Indoor 3D Routing and Tracking Services

A key part of this thread of OWS-6 is the ability to perform efficient path routing seamlessly between the outdoor and indoor environments. Outdoor 3D routing includes the ability to accurately incorporate and visualize 3D terrain and path elements such as overpasses, underpasses, tunnels, bridges, and other such features in the route display, see Figure 4-35 for one example. Note that CityGML of itself does not include a data model for routing information (path elements, travel time, relationship with building-related features), which will need to be developed.

In addition to generating efficient and accurate routes, it is also important to be able to track mobile subjects such as people moving through the landscape and building environments. There are at least four current approaches to automated location tracking of mobile subjects: through triangulation and signal strength analysis of Wifi radio, cellular phone, or TV broadcast signals (depending on what type of receiver is available to be tracked); or direct monitoring of active RFID tags. In a hostage crisis (as mentioned in one use case scenario, Section 4.4.4.3) the efficacy of any of these four methods would be highly variable, depending greatly on the available networking infrastructure as well as on the capacity and readiness to use such infrastructure. However, in some controlled settings, such as a university campus or airport, policy assumptions may be made that could facilitate the choice of one or another of these methods. For example, it might be assumed that first responders to any crisis situation could be expected to wear active RFID tags to enable direct monitoring of their positions. Depending on the setting in which the crisis is taking place, eg, a university campus or an airport terminal, passive RFID tags may have been installed throughout the facilities. Then, as each individual responder passed by the passive RFID sensors, the active RFID tag would signal the responder's presence at those locations. While it might be feasible and practical for emergency responders to have such RFID tags, however, the same would not necessarily be true for university students, airport travelers, or emergency perpetrators such as a hostage taker. In these cases, tracking by cellphone or other means may work in some cases. A fifth approach to location tracking would simply be to monitor surveillance cameras previously installed.

Location tracking involves more than just identifying where a particular subject may be in 3D space; it is also necessary to relate that location to specific objects in the 3D built environment, such as "in which direction and how far is it from this subject to the nearest light switch?" This relationship may be analogous to the geocoding of addresses along a transportation route, except that in this case, each significant feature within a building would be geocoded to locations along a given route through the building.

Open Location Services (OpenLS) [OGC document 05-016] is a specification for a relatively lightweight protocol for sending and receiving spatial/temporal data with mobile devices such as handheld computers and cellphones. By encoding the full 3D coordinates for associated terrain and transportation network components, as well as building structural plans, route-determination services can begin to consider elevation as a key component.

OWS-6 should consider the functionality of the OpenLS Route Service with bindings defined for use with the Web, in particular these bindings shall be defined to work with the 3-D portrayal developments to be done in the DSS thread. However, work done so far with OpenLS for 3D routing may not yet have been tested (or perhaps even developed) for location-enabled handheld computers, PDA's and cellphones. Further development work may be required to complete this task item.

The OWS-6 DSS thread shall develop an Engineering Report describing the approaches used or most likely to be effective in routing and tracking across the boundaries between outdoor and indoor environments, and demonstrate these approaches to the extent possible.

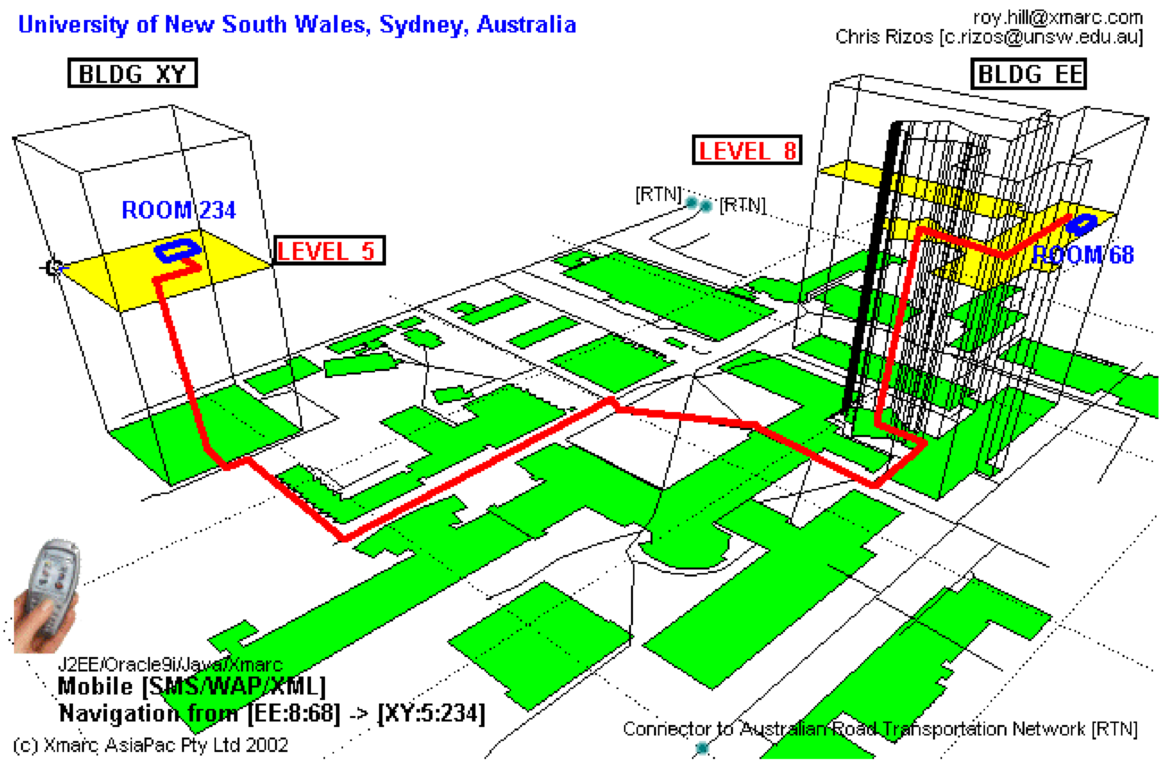


Figure 4-35. Indoor and Outdoor Tracking

4.4.2.4 WMS Performance

The following is from OGC Document 07-057r4, OpenGIS Web Map Tiling Service Discussion Paper. The Web Map Server (WMS) specification was developed to allow cartographic maps to be served over the internet in an interoperable manner. One of the primary goals of the WMS specification (other than interoperability) was flexibility. And in this respect it has been very successful. A WMS client is able to request the overlay of an arbitrary number of map layers in an arbitrary bounding window with an arbitrary background color at an arbitrary scale in a number of coordinate systems with a number of styles (and in some cases, with arbitrary user-defined styles). However, with this flexibility comes a price.

Since a WMS server is required to generate each requested map image on the fly, it does not scale well. It is theoretically possible for a WMS to cache the map images that it generates, but in practice, due to the unbounded nature of map requests, this doesn't aid performance much if at all because cache hits will be rare. Many implementations have demonstrated that it is possible to serve most cartographic layer requests within a few seconds, but this is generally in a single-request-at-a-time environment. By the very nature of what a WMS server is required to do, it is unreasonable to expect a WMS server to be able to handle the load of dozens or hundreds of simultaneous requests.

Therefore, WMS servers will never be able to achieve the popularity or ubiquity of what the industry is beginning to demand. In an attempt to tackle this issue, an alternative type of WMS service, called a Web Map Tiling Service (WMTS), has been in development. By limiting the rendering parameters in the request to a discreet set of values, it becomes possible for a WMTS server to serve pre-rendered images (called tiles) that require no processing whatsoever. The WMTS server can simply return the appropriate pre-generated image (e.g., PNG or JPEG) file. This mechanism is highly scalable (as similar systems such as

Google Maps illustrates). The currently proposed WMTS specification consists of a GetCapabilities operation, a GetTile operation, and a GetAlternateSources operation. It does not yet define a GetFeatureInfo operation that would be similar in nature to the WMS GetFeatureInfo operation.

The objectives for this thread in OWS-6 include the following:

- Develop SOAP interface for WMS Tiling
- Develop REST for WMS Tiling
- Stand up a service for integration into workflow scenario, to include GetFeatureInfo.

This task depends on resolution of discussions on the way forward for WMS Tiling capabilities between two competing approaches: Web Map Tiling Service (WMTS) and Web Map Service-Cached (WMS-C).

4.4.2.5 Integrated DSS Client

At the core of the integrated client concept is the requirement to provide a unified environment that allows a user to simultaneously visualize, analyze, and/or edit data from multiple sources. The development of this client will build on the results of the OWS-3 and OWS-4 Integrated Client for Multiple OGC-compliant Services (OGC Discussion Paper, document 03-021). The Integrated Client is defined as a software application that provides common functionality for the discovery, retrieval, and handling of data from WMS, Component WMS, FPS, WMS Tiled, WFS, WFS-T, WCS, SPS, SOS, video, and GeoRSS. Search is implemented via support for the CS/W ebRIM profile, and state is handled by support for WMS Context and/or OWS Context documents. The Integrated Client will extend the OWS-3 and OWS-4 Integrated Client capabilities to include the services developed and enhanced through OWS-6.

For OWS-6 this particularly includes the ability for the client to:

- ingest CityGML and render it;
- visualize CityGML buildings in the context of other (non-CityGML) georeferenced objects;
- support outdoor and indoor routing and tracking;
- support visualization of surveillance camera feeds;
- support multiple languages; and
- support OWS Context (OWC) [OGC 05-062], an extension to Web Map Context that allows the user to associate the client's multiple sources with a project. OWS Context shall be extended to use SLD for local as well as remote rendering, ie, without needing a WMS call.

Finally, among the client products from this thread there shall be a thin client that supports 3D visualization over the web, such as with WPVS or W3DS as mentioned in section 4.4.2.2, or other similar capability.

In cross-thread coordination with the GPW team (see GPW Requirements 4.2.2.9.), the DSS thread shall evaluate the use of Feature Portrayal Service (Component WMS) for a visual map display of DCS content from WFS. Recommendations, solutions and open issues shall be documented in a Data Accessibility Engineering Report for which the GPW thread is primarily responsible. This ER should include an annex written by the DSS thread to describe any recommendations and issues with the use of FPS, SLD and XSLT for data content specification portrayal from a WFS output to a WMS.

4.4.3 DSS Deliverables

The OWS-6 DSS thread requires two types of deliverables:

- **Engineering Reports and Documents**: shall be prepared in accordance with OGC published templates as defined on the OGC portal. Engineering Reports shall be delivered by posting on the OGC Portal Pending Documents list when complete and the document has achieved a satisfactory level of consensus among interested participants, contributors and editors. Engineering Reports are the formal mechanism used to deliver results of the Interoperability Program to sponsors and to the OGC Specification Program Domain and Specification Working Groups for consideration. All documents created in response to this program shall include "OWS-6" in the title, to facilitate later literature searches.
- **Services, Clients and Tools**: shall be provided by methods suitable to its type and stated requirements. For example, services and components (ex. WFS) are delivered by deployment of the service or component for use in the testbed via an accessible URL. A Client software application may be used during the testbed to exercise services and components to test and demonstrate interoperability; however, it is most often not delivered as a license for follow-on usage.

Note that certain draft deliverables will be required by the Interim Milestone at the date shown in the Master Schedule (Main Body, Section 4.6), for use in cross-thread development. These early deliverables will be designated and handled on a thread-by-thread basis.

4.4.3.1 Engineering Reports (ERs) and Documents

The following Engineering Reports (ERs), and Schemas will be developed in the DSS thread and submitted to the OGC Specification Program at the completion of the OWS-6 Testbed.

1)	OWS-6 WMS – Tiling ER (SOAP, REST)
2)	OWS-6 WMS – Tiling CR
3)	OWS-6 ISO 19117 / SLD Harmonization ER, including OWS-6 SLD Schema (DCS, ISO 19117) if possible
4)	OWS-6 SE CR
5)	OWS-6 SLD CR
6)	OWS-6 3D Flythrough ER
7)	OGC Web Context ER

4.4.3.2 Services, Clients, and Tools

Implementations of the following services, tools and data instances will be developed in this OWS-6 thread, tested in Technology Integration Experiments (TIEs) and invoked for cross-thread scenarios for OWS-6 demonstration events:

1)	WMS – Tiling, Portrayal SOAP Server
2)	WMS – Tiling, Portrayal RESTful Server
3)	WMS, WPS Client
4)	3D/Flythrough Portrayal of GML Server
5)	3D/Flythrough Client
6)	Integrated Client for multiple OWS services
7)	3D Outdoor and Indoor Route Service

8)	3D Outdoor and Indoor Location Tracking Services
9)	Web service for video camera feeds

4.4.4 DSS Enterprise Viewpoint

4.4.4.1 Objectives

The DSS Enterprise Viewpoint captures the capabilities that must be present in support of Geospatially enabled 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.

4.4.4.2 System Concepts

The fundamental concept underlying Geospatial DSS is that a decision maker is able to sit down at a single workstation, identify any resource anywhere, access that resource, bring it into their operational context, and integrate it with other resources to support the decision process. All of this takes place in a global enterprise made up of many different organizations and many different information communities. Each of them have their own information models and semantics as well as their own policies and procedures. GeoDSS tools allow the decision maker to navigate this heterogeneous environment with minimal distraction from the issue at hand.

Portals and Application Clients represent the tip of DSS which aims to provide interoperable access to distributed geospatial web services to aid decision makers in forming, analyzing, and selecting alternatives. DSS includes workflow management to produce context specific results from information and knowledge from multiple communities. One objective of geospatial web services is to allow decision makers to access and use information that may have been collected for other purposes.

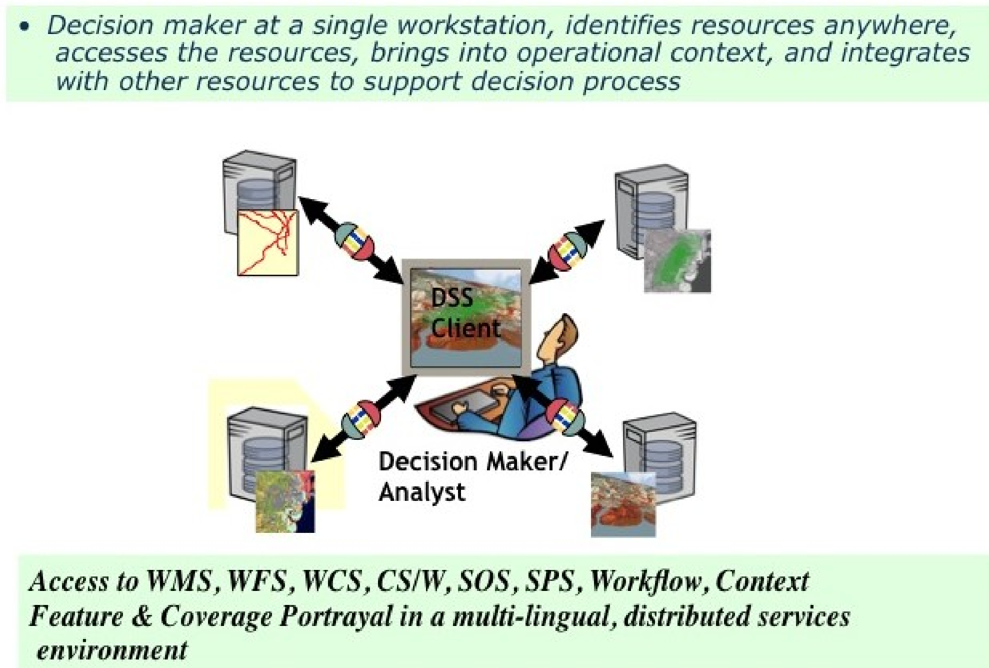


Figure 4-36. DSS Fundamental Concept

4.4.4.3 DSS Use Case #1: Situational Awareness / Emergency Management

Use Case Id:	OWS-6 DSS #1	Use Case Name:	Situational Awareness / Emergency Management
Use Case Domain:	OWS-6 DSS Emergency Response	Status:	Draft 2008-07-18
Use Case Description:	This use case describes an armed hostage-taking incident at a university campus, where authorities use location-specific data accessed via web services to isolate and differentiate the hostage-taker from the hostages in order to apprehend the perpetrator and free the hostages.		
Actors (Initiators):	- Hostage-taker - Emergency Management Control Center (EMCC)	Actors (Receivers)	- Hostages (students) - Local police, SWAT team - Emergency mgt (EM) analyst
Pre-Conditions: - Security camera locations are available and transmitted via web services, allowing localization of visualized scenes. - Web services for university asset location information exist, accessible to central command systems.		Post-Conditions: - Hostage taker in custody. - Hostages in medical care.	

Use Case Id: OWS-6 DSS #1	Use Case Name: Situational Awareness / Emergency Management
System Components: - CS-W: Catalog Service Web Profile - WFS-T: Web Feature Service – Transactional - Video feed web server - Sensor system: the collection of instruments that detect phenomena and generate the metadata associated with the readings as well as the underlying network and application infrastructure - Data Server: A web service that stores and disseminates data - Notification Service: Informs the user of events related to the requested automated processing	
Basic Course of Action 1. 911 cell call is placed by student witness who escaped a hostage-taker on a university campus. 2. Local police force responds on the scene with SWAT and other assets. 3. National emergency management control center is notified, takes control of situation. 4. Emergency management analyst at control center connects to 3D campus web services. 5. Hostage taker makes cell call to media with demands. 6. Emergency management analysts visualize location of hostages, based on web services from fixed security camera locations, the hostage-taker (based on cell phone derived location) and other EM assets in the field, all via web services. 7. Key tactical locations are identified by EM analyst at control centre using visualization software. 8. EM analyst calculates and assigns 3D travel routes from current asset locations to tactical locations. 9. Assignments and routes transmitted by EM analyst to assets on the ground using appropriate EM symbology. 10. EM assets visualize, then follow assigned 3D routes to tactical locations. 11. **(Optional)** Hostage taker moves, assets are re-tasked dynamically**.** 12. **(Optional)** Some EM assets are re-located**.** 13. Hostage taker is neutralized, situation is resolved.	

4.4.4.4 Use Case #2: Building Fire at an Airport

Use Case Identifier: OWS-6 DSS #2	Use Case Name: Building fire and Airport
Use Case Domain: OWS-6 DSS	Status: Draft 2008-07-18
Use Case Description: The intent of this scenario is to set up three levels of security environments for testing and evaluation. An	

<table>
<tr><td colspan="2">open environment (EMT Call Center), two closed secure environments (Airport and DHS) in which one (Airport) will provided temporary access to the open environment and the other (DHS) will provide temporary access to the other closed and trusted environment (Airport).</td></tr>
<tr><td>Actors (Initiators): Emergency Control Center, Airport Authority, Department of Homeland Security</td><td>Actors (Receivers) Air Traffic Control Tower, police, fire fighter team, airport travelers, incoming aircraft, outgoing aircraft</td></tr>
<tr><td>Pre-Conditions:
- The available information resources have been cataloged, and pertinent catalogs are known
- The Emergency Control Center has been activated, and authority for decisions and actions assigned.</td><td>Post-Conditions:
All available relevant information has been reviewed and a decision made.</td></tr>
<tr><td colspan="2">System Components
- 3D Client(s); browser based (EMTs) and thick (operations centers)
- Surveillance cameras available by Sensor Observation Services
- WCS (weather data) (OGC non SOAP)
- WCS/JPIP (Image data) (SOAP)
- CS/W ebRIM Profile (registered and catalogued data, conflation and portrayal rules)
- WMS tiling; providing city level map data (REST)
- WFS (SOAP) serving building blueprints/models
- WFS (SOAP) serving airport data
- WFS (SOAP) serving CityGML (UrbanMSD)
- Conflation Service</td></tr>
<tr><td colspan="2">Course of Action:
1. The Emergency Management Team (EMT) call center receives a call that a large fire has broken out at the Federal Building.
a. The EMT call center queries for and downloads both city scale maps and 3D digital data. This data is sent out to both police and fire personnel in route.
b. The call center tracks the movement of the response team as it approaches the fire. It is determined that a number of persons are still present in the building (RFID).
c. Fire rescue persons on the scene request blueprints of the building. Fire rescue personnel make an assessment of weather conditions at the fire and request weather data. High winds are threatening to expand the fire and smoke in the direction of the airport.
2. The EMT call center contacts the airport authority to notify them of a possible fire threat and requests access to their current weather data and airport facilities data. The Airport Authority agrees to provide temporary access to their data holdings and web services to the EMT call center.
3. The EMT call center contacts the Department of Homeland Security to notify them of a major fire at the Federal Building and requests building blueprints and building models.
a. The DHS states it would be impossible to set up secure access for the EMT call center in the amount of time required to support this crisis but they have already established a level of trust on a need to know basis with the Airport Authority. It is agreed that that Airport</td></tr>
</table>

Authority will be allowed access to the blueprints and building models for that specific building.

b. Airport Authority provides limited access to building blueprint and model data. The blueprints are available as imagery data exclusively. The size of the imagery files is very big. As time is crucial at this stage, the blueprints will be transferred using JPIP streaming servers, available via Sensor Observation Service or Web Coverage Services. This allows the EMT to select the most urgently required subsets from a low resolution but full size blueprint. Thus the EMT receives the important parts of the blueprint in high resolution in a fast and efficient way.

4. EMT review of airport data indicates fuel storage tanks in close proximity to the building fire. Concerns are realized in that Fuel Tanks did not exist on original EMT city level data.
    a. EMT call center makes the determination that the original city level data provided to EMT personnel is out of date over the airport and that the two datasets need to be conflated and merged.
    b. EMT notifies Airport Authority of building fire status and recommends constant monitoring of wind conditions and possible fire spread in the direction of the fuel storage tanks.
5. Airport Authority monitors the fire situation and notifies incoming and outgoing aircraft of possible fire spread and smoke obscuration diversions if the conditions worsen.

4.4.4.5 Use Case #3: Thin Client for 3D Fly-through

Use Case Identifier: OWS-6 DSS #3	Use Case Name: Thin Client for 3D Fly-through
Use Case Domain: OWS-6 DSS	Status: Draft 2008-07-18
Use Case Description: An analyst uses a client application to view 3D GML data.	
Actors (Initiators): Analyst (3DA)	Actors (Receivers) Same as initiator
Pre-Conditions: - The available information resources have been cataloged, and pertinent catalogs are known	Post-Conditions: All available relevant information has been reviewed and a decision made.
System Components - 3D Viewer Client - 3D GML data server	
Course of Action: 1. 3DA activates the 3D Client 2. 3DA queries catalog to see what feature data is available 3. The catalog accepts the query and sends a result set 4. 3DA selects data sets meeting specific format requirements, such as CityGML 5. 3DA selects symbology to display the features 6. 3DA views feature data 7. 3DA selects viewer content, sends gerFeatureInfo request to check feature attribution	

4.4.4.6 Use Case #4: GML Viewer with OWS Context

Use Case Identifier: OWS-6 DSS #4	Use Case Name: GML Viewer with OWS Context
Use Case Domain: OWS-6 DSS	Status: Draft 2008-07-18
Use Case Description: A non-specialized professional uses a client application to view GML feature data.	
Actors (Initiators): User	Actors (Receivers) Same as initiator
Pre-Conditions: - GML data in well-defined application schemas have been packaged with the client	Post-Conditions: GML data is viewed and explored.
System Components - see section 4.4.2.5 – Integrated DSS Client	
Course of Action: 1. User activates the Integrated DSS Client 2. Using a pre-packaged Context document, various local data sets are immediately portrayed in the client 3. User zooms, pans, and queries the data in view 4. User performs point-and-click feature queries of local GML data 5. User performs data searches using a built-in, predefined CS/W catalog 6. User adds remote WMS, WFS and WCS data to the view 7. User saves the state of the session into a new Context document	

4.4.5 DSS Information Viewpoint

The Information Viewpoint considers the information models and encodings that will make up the content of the services and exchanges to be extended or developed to support OWS-6. This discussion will consider these according to the application areas discussed in the Enterprise Viewpoint.

- ISO 19117 and OGC SLD Portrayal
- 3D Portrayal of GML with Fly-through
- Hosting CityGML data with WFS
- Outdoor and indoor 3D route and tracking services
- WMS performance (tiling)
- Integrated Client for multiple OWS services

4.4.5.1 ISO 19117 and OGC SLD Portrayal

Relevant Specifications:

- ISO 19117:2005 – Geographic Information - Portrayal
- OGC 02-070 – Styled Layer Descriptor (SLD), Annex F
- OGC 05-078r4 and 08-064 – SLD Profile of Web Map Service (WMS), with CR
- OGC 05-077r2 – Symbol Encoding (SE)

ISO 19117:2005 presents a simple model of portrayal (Figure 4-37). The portrayal catalogue consists of the feature portrayal, portrayal rule and external function. To produce different products, several portrayal catalogues may exist, portraying one or more datasets. The portrayal catalogue relates to one or more portrayal specifications, and one portrayal specification may be used in one or more portrayal catalogues. A portrayal rule consists of two parts, a query statement that can use one or more external functions, and one or more action statements.

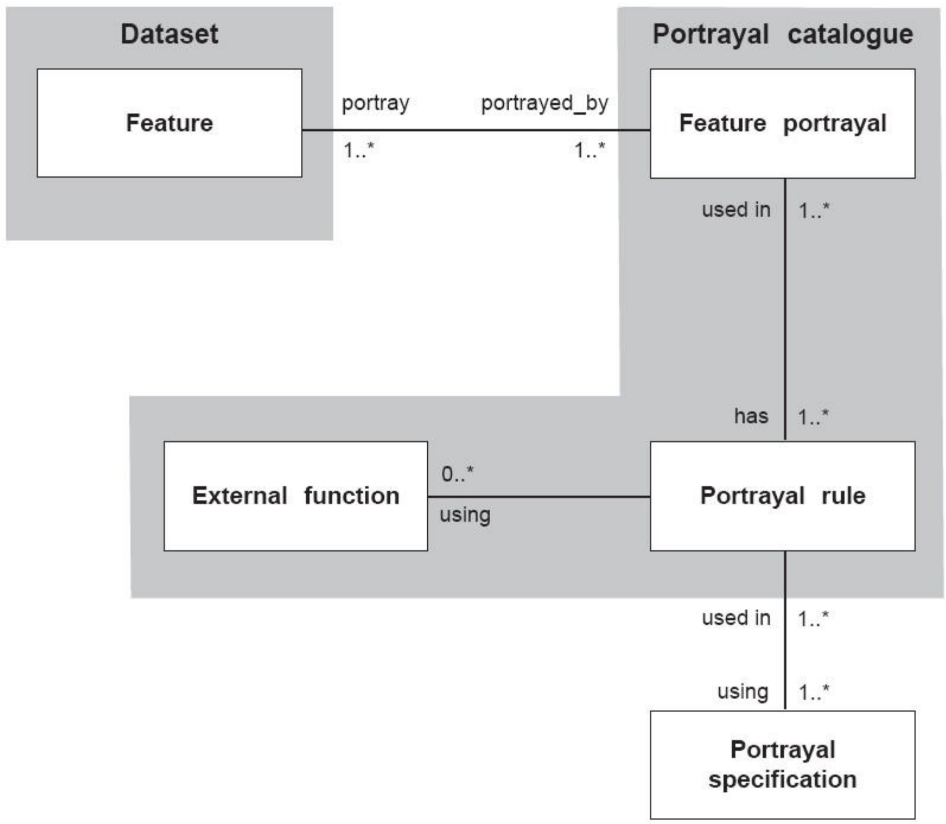


Figure 4-37. Overview of Portrayal (ISO 19117:2005)

EXAMPLE 1 A Dataset contains instances of the feature class Road. The feature class Road contains two attributes, classification and segment. The classification attribute is of string data type, and may have the value “country road” or “town road”. The segment attribute is of GM_Curve type and contains the spatial description of the road. The Portrayal Specification used is called N50_specification. The two Portrayal Rules in this example look like this (The “quotes” in this example are used to show the contents of a string):

- IF (Road.classification EQ “country road”) THEN drawCurve (“N50_specification.Solid_red_line”, Road.segment)
- IF (Road.classification EQ “town road”) THEN drawCurve (“N50_specification.Solid_yellow_line”, Road.segment)

In this example the THEN separates the query and action statements. The drawCurve is an action statement drawing an actual curve using geometry from Road.segment and colour, line width etc. information from

N50_specification.Solid_red_line and N50_specification.Solid_yellow_line.

EXAMPLE 2 If the portrayal varies with the scale, an External Function is needed as part of the query statement. One of the Portrayal Rules then may look like this (The "quotes" in this example are used to show the contents of a string.):

- IF (Road.classification EQ "country road" AND Scale (<=20000)) THEN drawCurve ("N50_specification.Solid_thin_red_line", Road.segment)

Here Scale is a function that gets the display scale from the display device.

The portrayal rule refers to the appropriate attributes, functions and relationships defined in an application schema. The portrayal catalogue also lists the external functions used, including the parameters and returned values.

EXAMPLE 3 In these cases external functions are necessary:

- The electronic map in a car navigation system has to be displayed so that the up-direction of the map is always in the direction the car is moving. To be able to specify the rotation of the map, the current position of the car must be retrieved continuously from an external position device using an external function.
- For electronic chart displays onboard a vessel some of the symbols are only valid for certain scale-intervals. To be able to turn the symbols on and off the system must be told what scale the map is displayed in by the display part of the chart system. A danger zone is defined spatially as a surface. Below a certain scale the danger zone is better displayed by a point symbol. An external function may be used to compute the centroid of the area and the coordinates of the centroid used to position the point symbol.
- An external function may be used to avoid visual conflicts between text and symbols placed on a map, or to handle the placement of text along curves.

The model just described has not been adequate for many applications and is currently being revised. Note that the above overview and examples address features broadly by type, and not by attribute values. Extending the portrayal model to encompass potentially complex attribute value mapping of features to symbols shall be coordinated with the ISO 19117 revision editing team. The current draft of the revised Portrayal Process is shown in Figure 4-38. The sections of Figure 4-38 in gray are the parts of ISO 19117 subject to the current revisions and most applicable to SLD and SE work in this thread of OWS-6. The revision team expects to complete the UML model in the near future.

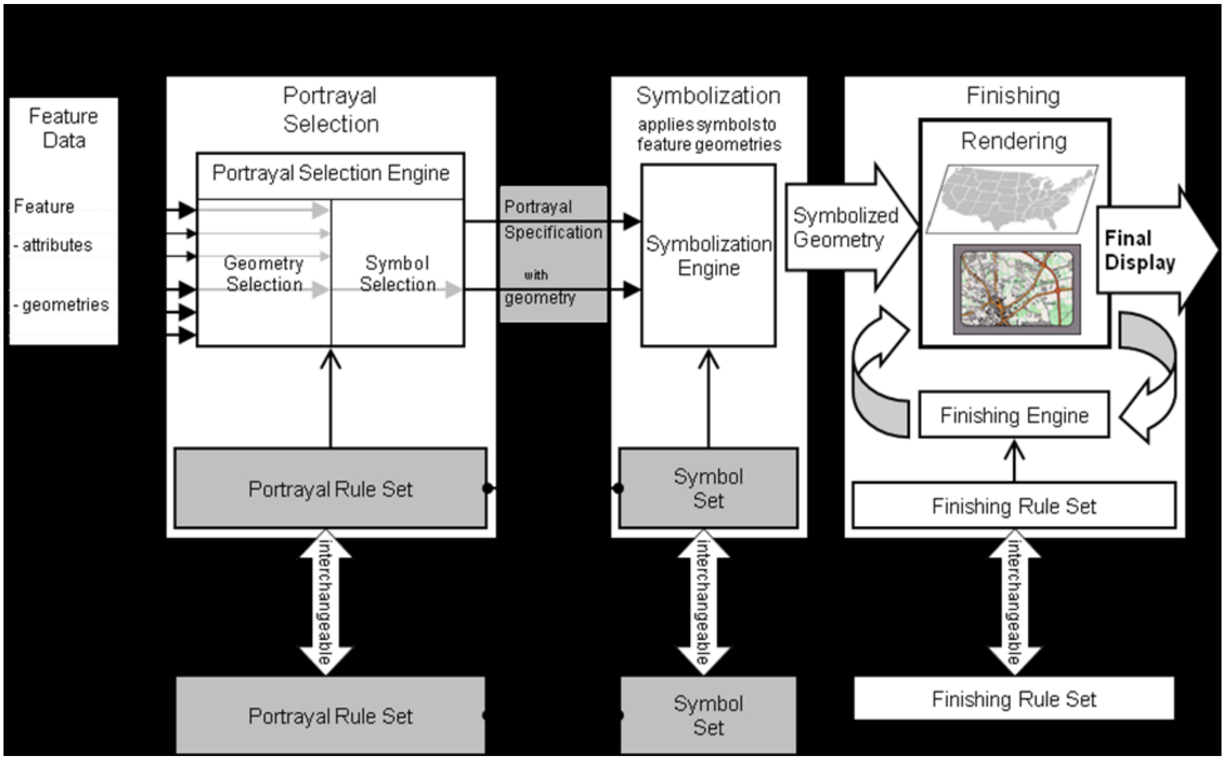


Figure 4-38. Overview of Portrayal (ISO 19117 revision draft)

Explanation of terms used in Figure 4-38:

- Feature Data – Render ready data. May have been processed by applying generalization and cartographic thinning. Features and feature geometries may exist at multiple levels of detail.
- Portrayal Selection – The process that selects the portrayal for a given feature. This is primarily a mapping of features with a given delineation to a specific symbol. The selection process may also select or calculate geometry as a prerequisite for the symbol selection as well as specifying parameters for symbols.
- Portrayal Rule Set – A set of rules, which, when evaluated for a feature and a context, selects a portrayal specification.
- Portrayal Selection Engine – An implementation which evaluates a rule set for a given feature and context and returns a portrayal specification with actual parameters and the feature geometry. The portrayal specification is used to portray the feature geometry.
- Portrayal Specification – Specifies a portrayal for a feature by referencing a symbol and passing actual parameters.
- Symbol Set – A collection of symbols that may be referenced by a portrayal specification and applied to a feature geometry.
- Symbolization – The act of applying a symbol to a feature geometry as well as the result of this act.
- Symbolization Engine – An implementation which applies symbols as specified in a portrayal specification to feature geometries.
- Symbolized Geometry – The result of symbolization. Feature geometry with symbols applied.
- Finishing – The process which refines the presentation of symbolized geometry, primarily ameliorating symbol conflicts such as over printing. This process may be iterative, taking a symbolized geometry and applying finishing rules, which modifies the geometry, resulting in a new

symbolized geometry. This process may be applied repeatedly until the iteration input remains unchanged by the finishing process.

- Finishing Rule Set – Rules that specify the finishing process such as over print priority and displacement rules.
- Finishing Engine – An implementation which evaluates finishing rules and applies them to a given symbolized geometry.
- Rendering – Depiction on a printed map or on a video display.
- Final Display – The end result of the portrayal process; a printed map or an on-screen display.

The primary concern in this DSS thread is to separate the Portrayal Rules, Symbol Sets, and Portrayal specifications from the feature data. While portrayal rules and symbol specifications may be expressed inline with the Feature Portrayal Service (FPS) invocation, it is important to also be possible to refer to each portrayal rule and symbol specification if they are external to the FPS call, ie, stored in a Portrayal Rule Registry and/or Symbol Set Registry Note that this approach appears to inherently support feature and symbol subtypes, one of the requirements stated in Section 4.4.2.1. The Finishing and Rendering sections of the Portrayal Process are out of scope for OWS-6, but are included in the diagram for completeness of context.

Another limitation of ISO 19117 in these examples is that it does not specify how to organize the layers relative to each other, or how to perform actions on just a selection from a given feature class. Suggestions for changes to accomplish these tasks in ISO 19117 shall be included in the ISO 19117 / SLD Harmonization Engineering Report.

The following is a draft UML model for Feature Portrayal.

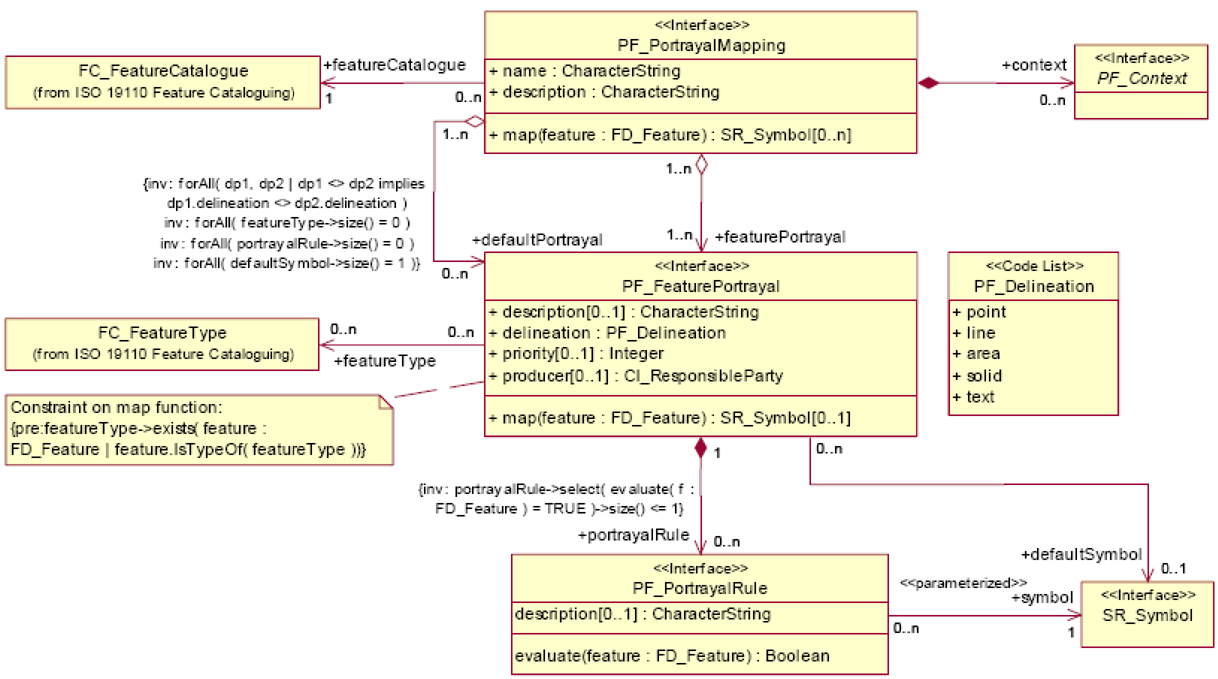


Figure 4-39. Portrayal Schema

4.4.5.2 3D Portrayal of GML with Fly-through

CityGML is an information model for the representation of 3D urban objects. It defines the classes and relations for the most relevant topographic objects in cities and regional models with respect to their geometrical, topological, semantic and measured appearance properties. Included are generalization hierarchies between thematic classes, aggregations, relations between objects, and spatial properties. This thematic information goes beyond graphic exchange formats and allows users to employ virtual 3D city models for sophisticated analysis tasks in different application domains such as simulations, urban data mining, facility management, and thematic inquiries.

CityGML is realized as an open data model and XML-based format for the storage and exchange of virtual 3D city models. It is implemented as an application schema for GML 3. CityGML is intended to become an open standard and therefore can be used free of charge. The CityGML specification document is currently an OGC Candidate Implementation Specification. (http://www.citygml.org/)

CityGML is extensible through the use of external codelists and dictionaries. These will be useful for the rendition of specific Data Content Specifications (DCS), subsets of more broad Application Schemas.

Figure 4-40. Visualization of a CityGML document

The information model for working with CityGML for DSS in OWS-6 is guided by the choices of scenarios for which to use the DSS Integrated Client. One scenario is a hostage crisis on a university campus; another is a building fire near an airport. For each of these situations, information sources include surveillance camera feeds, building structural plans, transportation routes to approach and enter the scene of the emergency, strategic and tactical plans for resolving the situation, communications channels among the various law enforcement, SWAT, fire, and other responder units present, and possibly other information flows. Web based catalog services are also important for enabling analysts and decision makers to find out what information is available, what level of quality it may have, and so on.

Ideally, the initial building data source with the best quality to use for creating CityGML datasets would be digital Building Information Model (BIM) data assembled during the architectural design phase and maintained during construction. Tools have been developed for extracting CityGML datasets from BIM data. It is outside the scope of OWS-6 to focus research effort on this extraction process; for OWS-6 there will be ESRI Shapefile data from which to construct the CityGML representations.

Presumably, the CityGML data will include multiple levels of detail to facilitate visualization and routing through the urban landscape. CityGML includes data elements for representing multi-modal passageways.

However, the CityGML data model does not include sufficient information on its own for capturing and storing the routes within buildings. The pathways, time durations, and the very relationships between the outdoor landscape, the route and building objects such as doorways and pathways, will need to be designed and represented.

CityGML contains considerable styling information for the higher levels of detail (2-4), but very little styling for the lowest levels (0-1). Since at level 1, CityGML buildings are simple extrusions from the ground level, consisting of collections of faces (patches) rather than volumetric solids, it may be feasible and interesting to enable SLD and SE to portray limited styling information on the extruded faces, such as simple lines on building faces to indicate the number of floors or presence of windows in the building.

Access to CityGML via WFS is a requirement for OWS-6. This may be requested to use the digital vector data directly, or for conversion to a simple image for portrayal by a viewer client. Key issues to consider when serving CityGML with WFS are discussed in further detail in Sections 4.4.6 and 4.4.7 below.

4.4.6 DSS Computational Viewpoint

The computational viewpoint is concerned with the functional decomposition of the DSS architecture into a set of services that interact at interfaces. It reflects the components, interfaces, interactions and constraints of the Service Architecture without regard to their distribution. For the DSS thread, those services are

- Web Map Service – Portrayal
- Web Map Service – Tiling
- Web 3D Service (W3DS) including X3D
- WTS and WPVS
- OpenLS Routing and Tracking Services
- The DSS Clients access these services as well as other OWS services provided by other threads

4.4.6.1 WMS including Component WMS

"Component WMS" is distinguished from "Integrated WMS" in that it is intended to be decoupled from any particular geospatial feature store, but rather designed so that it can be connected to any such store without prior knowledge of the feature types. Three approaches are defined to allow a Component WMS client to take advantage of SLD symbology:

- The client interacts with the WMS using HTTP GET but the request can reference a remote SLD.
- The client uses the HTTP GET method but includes the SLD XML document inline with the GET request in an SLD_BODY CGI parameter (with appropriate character encoding).
- The client interacts with the WMS using HTTP POST with the GetMap request encoded in XML and including an embedded SLD, as described in Subclause 6.6.

The third method is technically superior but there has been a great lack of vendor support in the past for an XML-POST GetMap-request method. Use of the second method, which is a compromise between the first and third methods, can encounter problems resulting from excessively long URLs.

It is important to note that in all cases the WMS has no prior knowledge of the SLD contents. There is a wide spectrum of possible clients. Some may allow a user to switch between a number of predefined maps, each specified by its own pre-defined SLD.

Others may allow a user to interactively define how they wish a map to appear and construct the necessary SLD 'on-the-fly'. All of the approaches described above allow a client application to do this but the first one requires that the client be able to place the SLD document in a Web location accessible to the WMS.

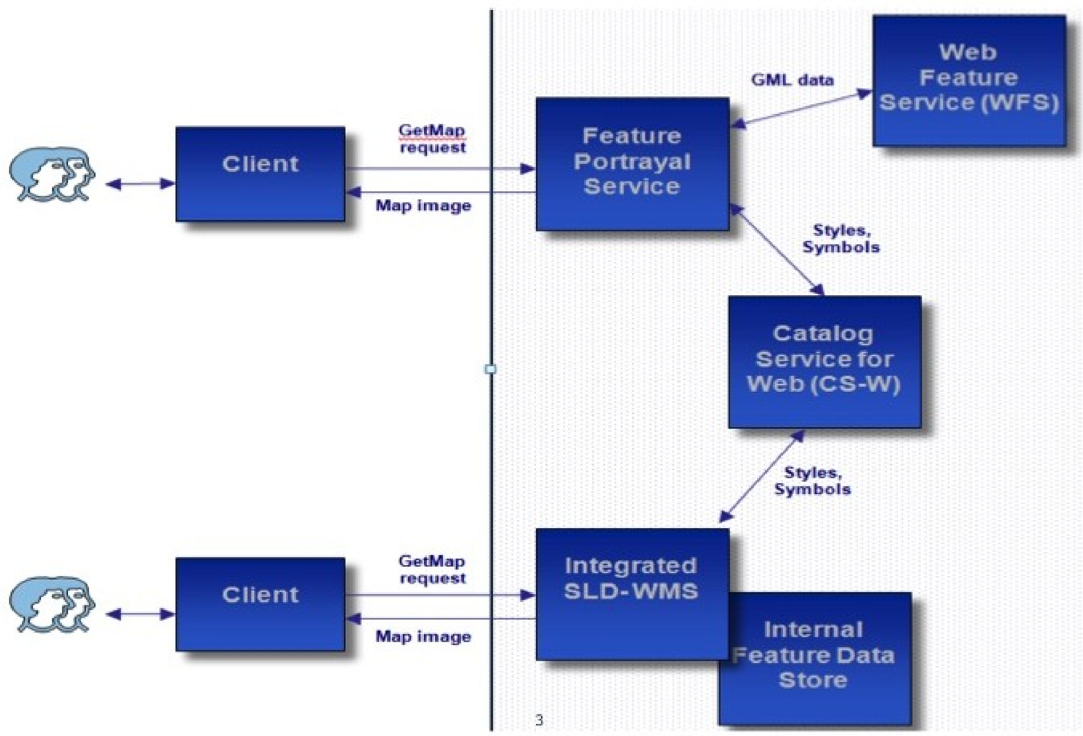


Figure 4-41. Comparison of component WMS and integrated WMS

4.4.6.2 WMS Performance (Tiling)

The Web Map Tiling Services (WMTS) specification draft (OGC 07-057r4) presents a basic set of requirements: WMTS service must be scalable. Therefore, servers must be able to quickly return pre-rendered tiles without any image manipulation being necessary. Support for arbitrary tile sizes, scales and alignments should be possible. This would allow existing tilesets to be served. It would also allow for future adaptation as to which tile sizes, scales and alignments are considered optimal on a project-by-project basis. However, for maximum interoperability, a standard set of tile sizes and scales should be recommended by the WMTS specification.

With the growing number of OGC web services being defined, implemented and deployed, and with client applications for these services becoming more and more sophisticated, it is becoming increasingly important to provide client applications with the ability to combine services. One way to provide this ability is to introduce an operation called GetAlternateSources, which, for a specified set of layers or feature types served by the server, reports other OGC web services that serve the same content (but perhaps in different forms). This would be a very useful operation, as it would essentially link services together. It would, for example, allow a client application to support browsing data on a WMTS server and then bridging over to the appropriate WFS server to download the actual features that are depicted in the map.

Note that the SLD 1.1.0 specification defines an operation called DescribeLayer that is somewhat similar to this. It allows an SLD-enabled WMS client to determine what feature types a layer is comprised of, and which WFS or WCS serves these feature types. However, the purpose of this operation is to provide the client with enough information about the layer so that it can perform SLD operations. It does not necessarily report an accessible WFS or WCS. Furthermore, it is not generic enough to link together all of the various types of OGC web services. Ideally, any OGC web service should be able to link to any other OGC web service. For example, a WFS should be able to report which WMS(s) and/or WMTS(s) a client application can go to in order to request map images representing the features of a given feature type.

In an effort to provide this functionality, the WMTS discussion paper proposes the introduction of a GetAlternateSources operation. In the future this may become appropriate to move into the OWS Common specification so that other services can make use of it without having to define their own version of the operation.

4.4.6.3 W3DS including X3D

Relevant specifications:

- "Web 3D Service" OGC Discussion Paper, OGC document 05-019.

The Web 3D Service is a portrayal service for three-dimensional geodata, delivering graphical elements from a given geographical area. In contrast to the OGC Web Mapping service (WMS) 3D scene graphs are produced by a W3DS server. These scene graphs will be rendered by the client and can interactively be explored by the user. The W3DS merges different types (layers) of 3D data in one scene graph. (Figure 4-42).

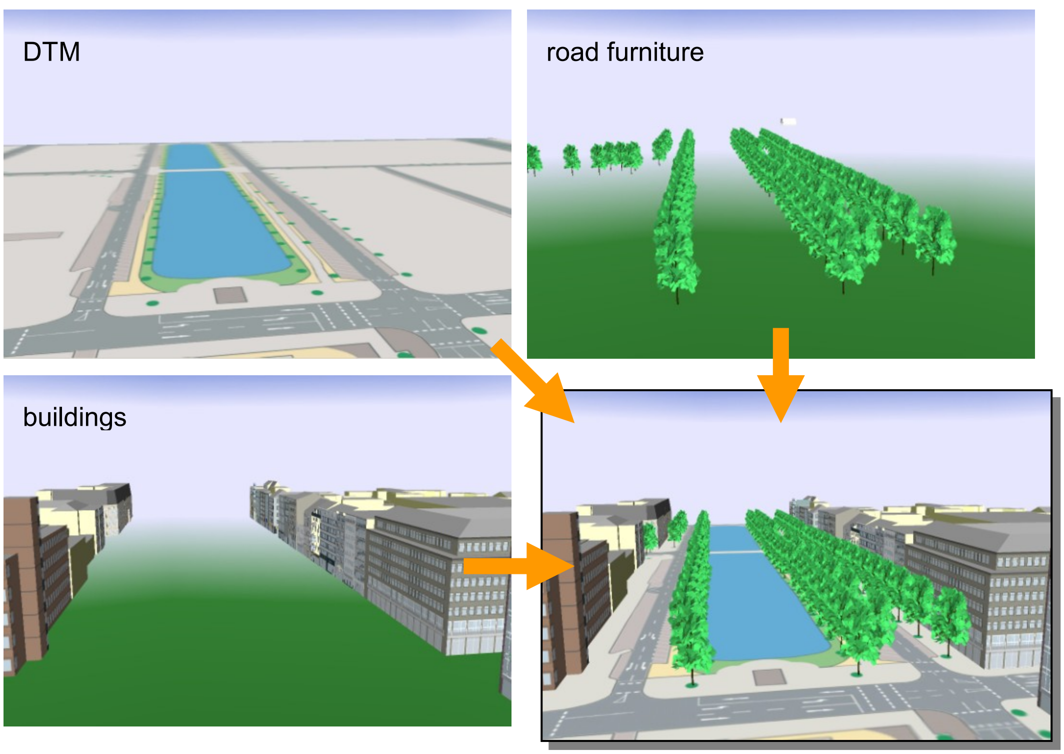


Figure 4-42. Different types of geodata are merged in one 3D scene graph

3D scene graphs represent visual illustrations of 3D geodata and are not the geodata themselves. They do not contain the semantic characteristics and relations of the basic geodata. For gaining access to the 3D geodata OGC's Web Feature Service is recommended. For OWS-6, X3D is to be evaluated as an output format.

The W3DS fills the gap between thin and thick servers (Figure 4-43). The use of standard formats and standard plug-ins allows the widespread and easy use of data visualization including interaction by the user.

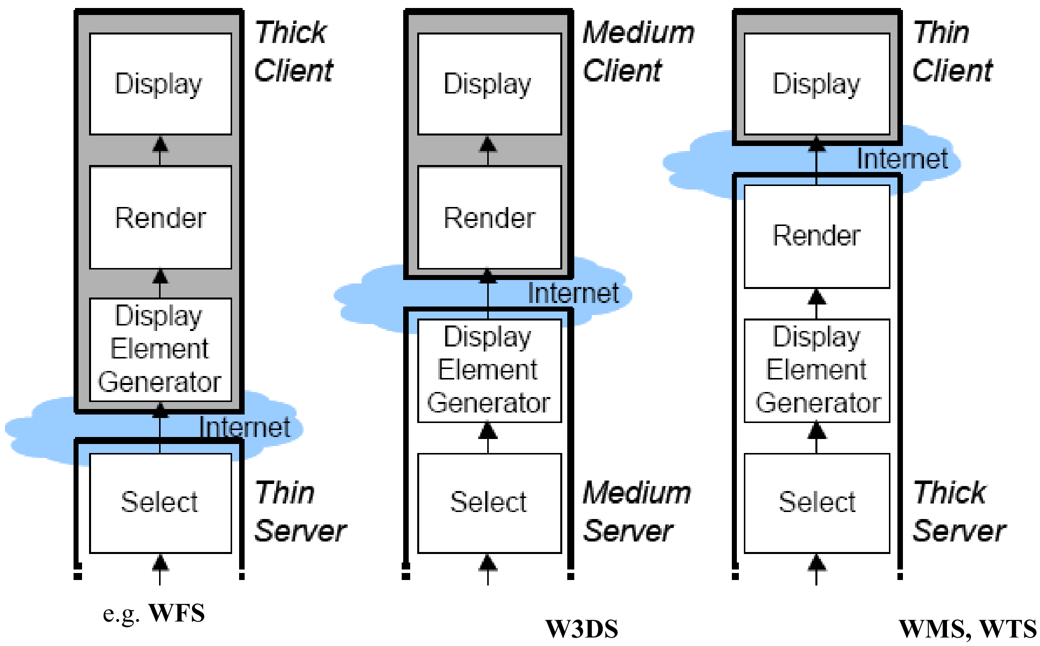


Figure 4-43. Different balancing schemes between client and server

4.4.6.4 WTS and WPVS

Relevant specifications:

- "Web Terrain Server," OGC Discussion Paper, version 0.3.2, OGC Document 01-061

A Web Terrain Service (WTS) produces views of georeferenced data. Here a "view" is a visual representation of geodata; a view is not the data itself. A View is a picture of a place that represents the normal viewpoint of a person or at least of a camera. It is a three-dimensional, perspective (i.e. non-orthographic) view of the world. WTS responses are generally rendered in a pictorial format such as Portable Network Graphics (PNG), Graphics Interchange Format (GIF) or Joint Photographic Expert Group (JPEG) format.

The WTS discussion paper has been the basis of work more recently gathered under the name of Web Perspective View Service (WPVS). Additional details about WPVS is available upon request.

Use of WTS (or WPVS) to achieve "flythrough" functionality is not currently defined in the WTS discussion paper.

4.4.6.5 Outdoor and indoor 3D routing services

Relevant specifications:

- OGC 05-016 OpenGIS Location Service (OpenLS) Implementation Specification: Core Services
    - Part 1-Directory Service, Part 2-Gateway Service, Part 3-Location Utility Service (Geocoder/Reverse Geocoder), Part 4-Presentation Service, Part 5-Route Service

The OpenLS Route 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.

In order to determine and store 3D routes based on CityGML data, additional tools and techniques will be needed, as mentioned in the Requirements section above. Participants may use OpenLS 3D route services for handheld computers, although this may require some additional development.

4.4.6.6 Outdoor and indoor 3D tracking services

Relevant specifications:

- OGC 05-016 OpenGIS Location Service (OpenLS) Implementation Specification: Core Services
    - Part 1-Directory Service, Part 2-Gateway Service, Part 3-Location Utility Service (Geocoder/Reverse Geocoder), Part 4-Presentation Service, Part 5-Route Service
- OGC 06-024r4 OpenLS Tracking Service

In OWS-3 and OWS-4 the Open Location Services (OpenLS) thread developed basic principles for tracking services (OLS-TS) that are part of the ongoing OpenLS 2.0 Revision Working Group activity. The term "Tracking" implies that there is a central repository of dynamic location information, a mechanism for updating the content of the repository, and a means for querying the location of some or all of the objects. In the development of the service, it was also found useful to be able to relay or "proxy" dynamic tracking information.

The proposed Tracking Service supports three functional capabilities:

1) Position Update to a central Tracking Server.
2) Position Query of the central Tracking Server by subscriber or Mobile Terminal ID.
3) Relay of Position Update messages from Tracking Server to other downstream consumers in a proxy server arrangement.

Figure 4-44 illustrates the basic tracking terminology. Figure 4-45 shows the key use cases associated with tracking.

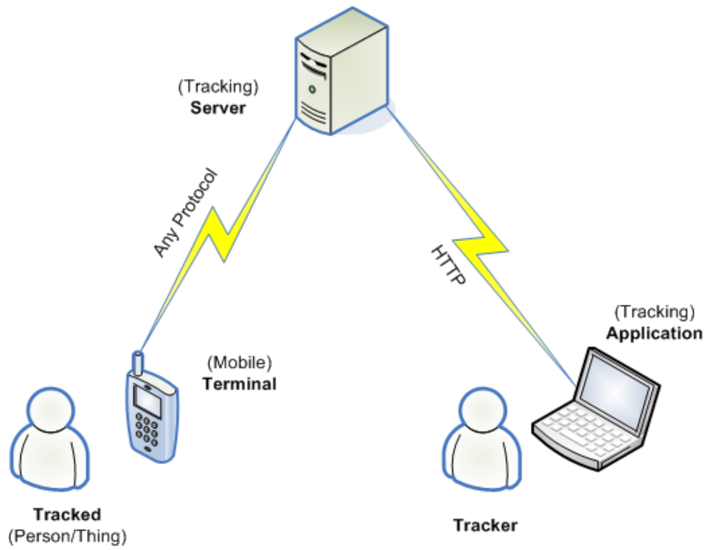


Figure 4-44. Illustration of Tracking Service Terminology

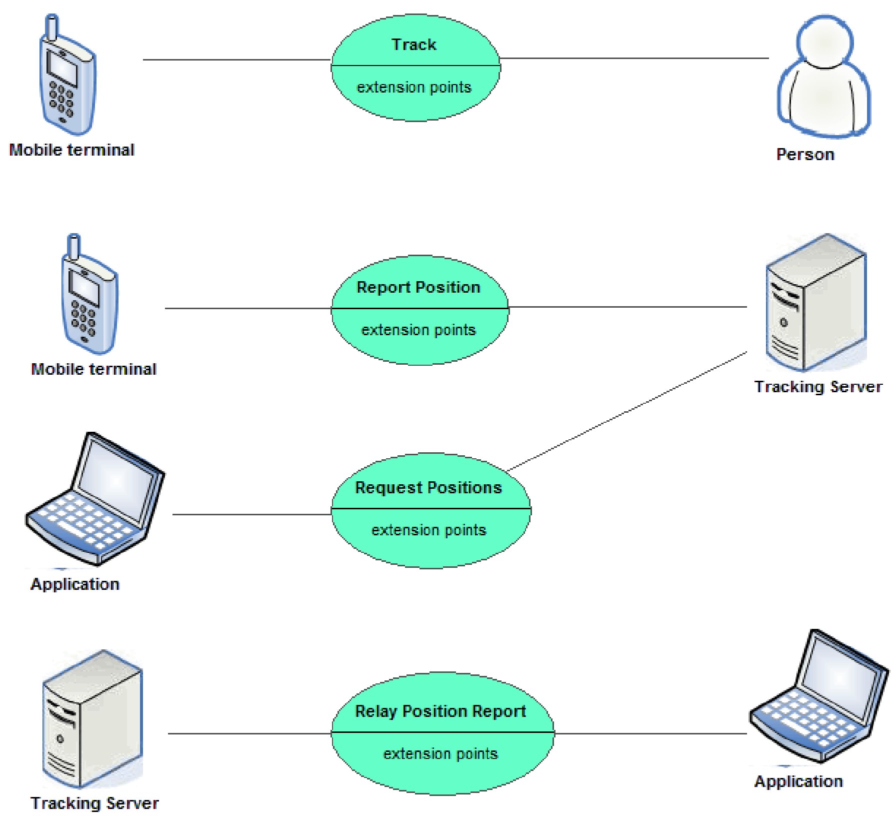


Figure 4-45. Tracking Server Use Cases

For OWS-6, web services are needed for tracking mobile subjects such as the hostage-taker, based on mobile phone usage.

Another form of tracking is to detect and report first responders and other beneficial participants wearing active RFID tags in an area that already has passive RFID sensors installed throughout the facilities.

Data modeling is required for this task area to determine the best data structures and workflow for transitioning from the outdoor landscape to the indoor environment, as mentioned in the Requirements section above.

4.4.6.7 OWS Context

Relevant Documents:

- OWS Context IE Final Report (OGC 05-062)
- Web Map Context Documents Implementation Specification (OGC 05-005)
- Web Map Context Corrigendum (OGC 08-050)

OWS Context document is an XML encoding that references remote and/or local OGC Web Services. OWS Context documents are related to, but more powerful than, Web Map Context Documents (05-005). Web Map Context is limited to referencing OGC Web Map Services (WMS). OWS Context can reference other OGC Web Services such as Web Feature Services (WFS), and Web Coverage Services (WCS).

Web Map Context is a companion specification to the OGC Web Map Service Interface Implementation Specification version 1.1.1. The WMS specifies how individual map servers describe and provide their map content. The present Web Map Context specification states how a specific grouping of one or more maps from one or more map servers can be described in a portable, platform-independent format for storage in a repository or for transmission between clients.

Considerable discussion has taken place about whether or not HTTP GET or HTTP POST was to be used for service requests, e.g. WMS "GetMap", and whether or not the onlineResource element in an OWS (or WMS) Context document was that of the "GetCapabilities" request or the final request: "GetMap" for WMS; "GetFeatures" for WFS; or "GetCoverage" for WCS. The following result was proposed and accepted by the revision working group:

- WMS Context are "lightweight" context documents.
- OWS Context are "industrial strength" context documents.
- Both WMS and OWS Context document clients should regard the OnlineResource as the GetCapabilities, not the GetMap, etc. This will be made clear in WMS Context 1.1.0, and future OWS documents.
- WMS Context 1.1.0 will also mention that this was not previously specified, so behaviour of WMS Context 1.0 clients cannot be relied upon.

4.4.7 DSS Engineering Viewpoint

4.4.7.1 Components for OWS-6 DSS

The initial components identified for OWS-6 DSS are shown in the following figure. The components interoperate using the services identified in the Computational Viewpoint. The components process, cache and store data as defined in the information viewpoint.

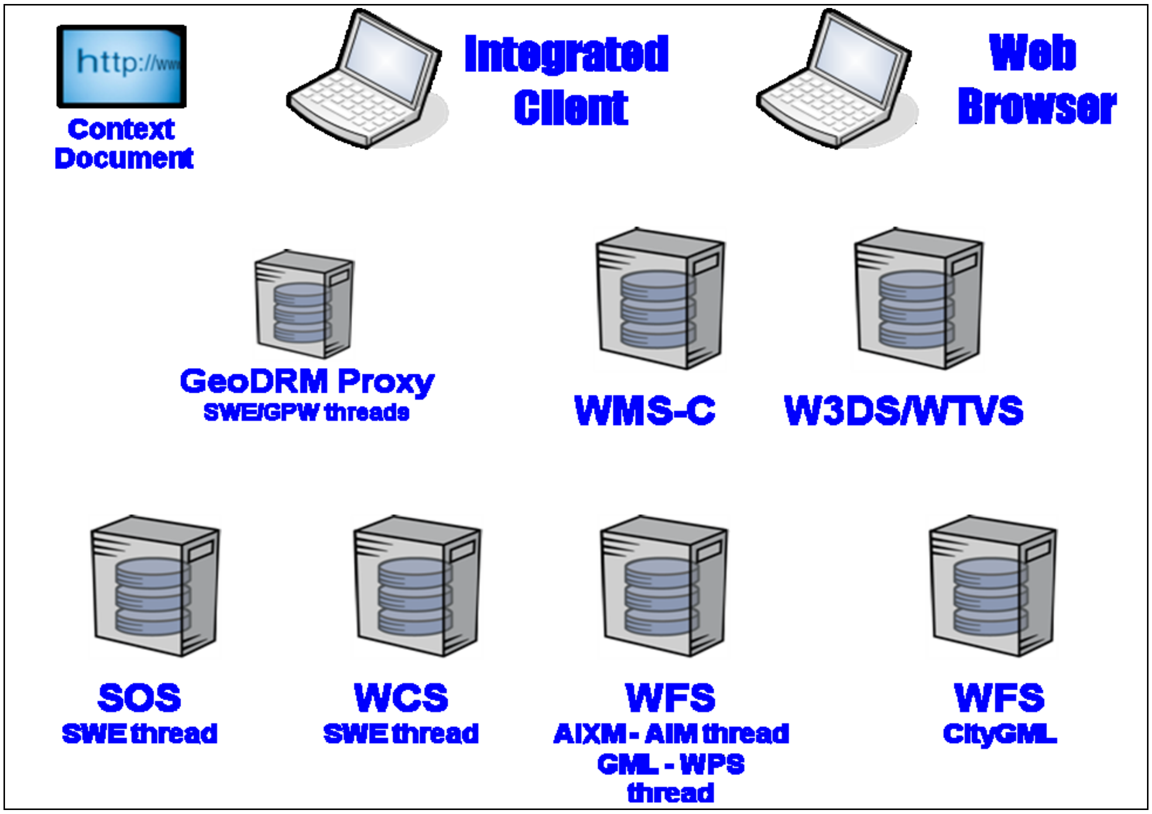


Figure 4-46. DSS Components and Interfaces

4.4.7.2 Integrated Client

Relevant Documents:

- OWS Context IE Final Report (OGC 05-062)
- OWS Integrated Client GeoDSS Client (OGC 05-116)

The core purpose of an integrated client is to provide a unified environment that allows a user to visualize, analyze, and/or edit data from feature, imagery, and sensor web data sources simultaneously. Within the context of the OGC, this means that the integrated client allows a user to publish, discover, access, integrate and apply all types of spatial data (e.g., raster, vector, coverages, and sensor observations) from a wide range of vendor "web services" through OGC standard interfaces.

The functionality of an integrated client can be divided into the following five categories:

a. Service Discovery & Binding

b. Feature Production

c. Imagery Production/Exploitation

d. Sensor Web Planning/Exploitation

e. Project Persistence and Sharing

For each integrated client, the implementation must harness specific technologies and adopt particular architectural approaches. Each technology/architecture pairing presents different reliability, availability, serviceability, usability, security, and performance characteristics. And as such, different technology/architecture pairings may be more or less suitable for various purposes across an enterprise.

The figure below is an example showing how an integrated client accesses multiple OWS servers.

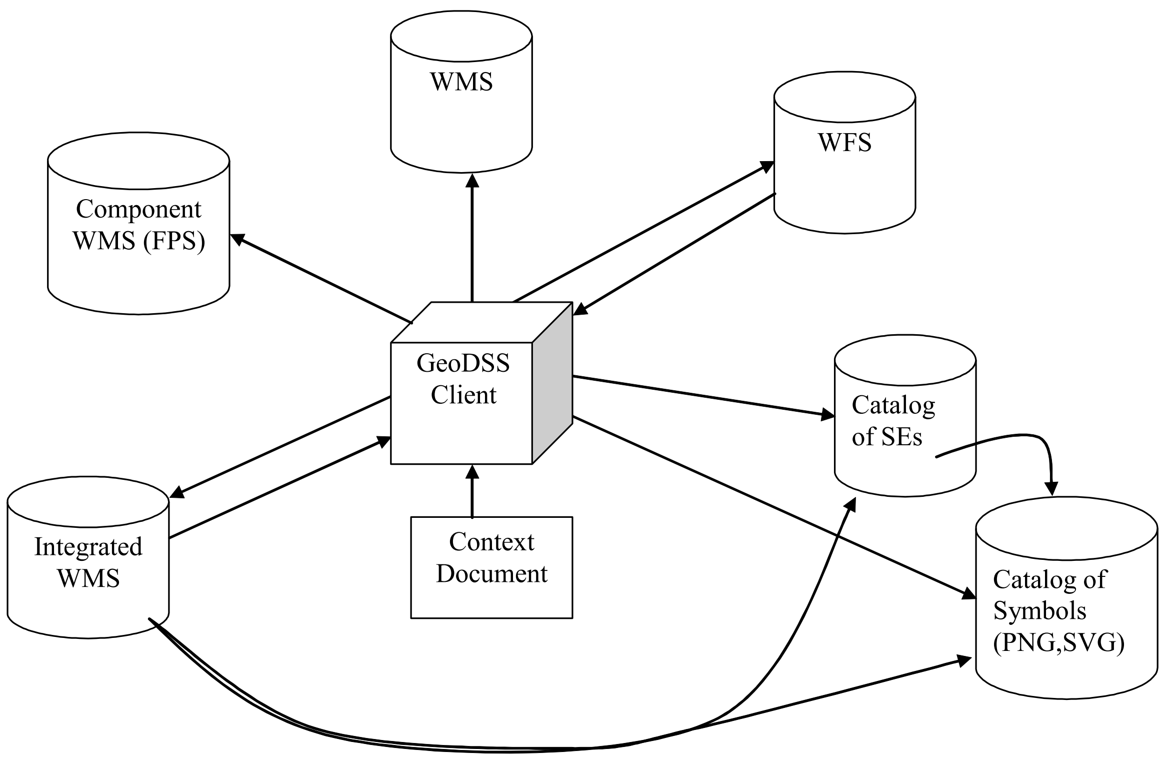


Figure 4-47. OWS Integrated Client

4.5 Compliance and Interoperability Test and Evaluation (CITE)

4.5.1 CITE Scope

Validating compliance with an OGC specification means verifying that a software product has implemented the specification correctly by testing the software interface for response and behavior that is outlined in the specification. Verifying compliance to the standard is necessary in order to achieve interoperability. As a result, geospatial application vendors desire to provide their potential costumers a means to verify adherence to OGC standards as a measurable discriminator for the interoperability of software products. Similarly, users desire assurance that acquired software components will interoperate with their existing investments in OGC-compliant technology. The Conformance and Interoperability Test and Evaluation (CITE) thread is intended to provide the geospatial industry (consumers and vendors) a methodology and tools that will test compliance with OGC web services.

The OGC Interoperability Program and the OGC Specification Program have achieved a great deal of momentum as a result of the multiple OGC web service specifications that have recently been published. Key consumers in the geospatial industry are modernizing their enterprises based on the applicability and interoperability of OGC web services. The major geospatial industry consumers require verifiable proof of compliance with OGC specifications in order to reach the desirable outcome of interoperability. Furthermore, as the OGC technology stack has matured, a group of interfaces has emerged that represents a baseline of technology needed to implement a fully interoperable, end-to-end *spatial data infrastructure*. The CITE threads in previous OWS projects have made significant progress towards having a complete suite of compliance tests for this baseline of interfaces.

Compliance Test scripts exist for the following web services:

- Catalog Service/Web v. 2.0.1
- Catalog Service/Web 2.0.2 and ebRIM/19115 profile
- Web Coverage Service v. 1.0.0* and 1.1*
- Web Feature Service v. 1.0.0 and 1.1.0*
- Web Map Service v. 1.1.1 and 1.3.0
- Sensor Observation Service 1.0*
- Sensor Planning Service 1.0*

* indicates a test that was developed in OWS5

A major focus of OWS-5 was on achieving consensus on the format and content of an Abstract Test Suite (ATS). The OWS5-CITE participants agreed to follow the ISO guidance for writing Abstract Test Suites. The ATS are used to develop Executable Test Suites which are the scripts that the TEAM Engine runs to conduct an automated compliance test. A major focus of OWS6-CITE will be in clearly documenting the approach to defining Abstract Test Suites. This will be a great benefit to the OGC community as the OGC Architecture Board (OAB) requires that new specifications be published with an accompanying ATS. In addition, a focus of OWS6-CITE will be to expand the usability of the existing OGC compliance tests by "tailoring" these tests for specific profiles and/or data.

The CITE thread will develop a suite of compliance test tools for testing and validation of products with interfaces implementing OpenGIS® specifications, as listed below. These scripts will be written for the approved Test, Evaluation, and Measurement (TEAM) engine. The participants in this thread will develop Engineering Reports (ERs) outlining the developed test guidelines and test scripts. These ERs will be presented to the Technical Committee and Planning Committee for approval. All development activities within the CITE thread should result in products that are fully functional on a Linux platform on the OGC IT infrastructure, excluding reference implementations which may be served from a vendor's facility.

In addition to development of tangible compliance testing tools, as listed in the deliverables section below, OWS6-CITE activity will include writing two focused Engineering Reports: 1. Best Practice paper to document/standardize Abstract Test Suite and 2. User specified CITE Test Data Engineering Report. These are described in the deliverables section below.

Abstract Test Suites need to be created in a consistent and repeatable manner. To accomplish this task OGC requires the development of a OGC Abstract Test Suites Development Guideline Engineering Report. This ER should be structured such that the intent is to progress as an OGC Best Practice document.

The ability to test a user implementation of an OGC compliant web service is tied to the ability to verify the output as being compliant to some user's guidelines. For example, a WFS that serves up NAS compliant Local MSD must include OGC interface compliance tests and also check for Local MSD data content compliance, that is the ability to determine not only that a service is OGC compliant but that it also serves data which is NAS compliant. OGC requires evaluation, documentation and guidance on the ability to insert user defined "data" in lieu of the standard OGC static compliance test datasets and schemas used in OGC CITE executable tests. The participant is encouraged to explore different strategies, methods and languages to support the concept of compliance testing of content. Guidance, recommendations and issues shall be documented as part of a User's Guide to Adding Test Data to OGC CITE Engineering Report.

In general, the planned activities for the CITE thread involve creating compliance test scripts for the required technologies. However, due to the fact that the TEAM engine is open source, if additional functionality is needed to complete the test requirements, it is expected that the participant will be able to make enhancements to the engine at the code level. In your proposal, please describe your ability and willingness to work with the technologies the TEAM engine is built upon—i.e. Java and the Saxon XSL interpreter. The workflow of a CITE thread is described in more detail associated with the CITE Workflow graphic, included below.

4.5.2 CITE Deliverables

The following Engineering Reports (ERs) will be developed in OWS-6 CITE and submitted to the OGC Specification Program at the completion of the testbed. All ERs will include a reference to the source code for the completed compliance test script suites.

Engineering Reports
1) Document WMS 1.3 Change Requests as result of updating ATS/ETS
2) WMS 1.3 ATS Change Request
3) WMS 1.3 DGIWG Profile ATS Change Request
4) Engineering Report to summarize TEAM Engine changes for profile-specific testing

Implementations of the following services, tools and data instances will be developed in this OWS-6 thread and tested in a beta test period. Please note, new ETS must have three "supporting" implementations (three implementations that pass 100% of the compliance tests0 in order to become approved compliance tests.

Web Map Service - Compliance Tests for Profiles & Reference Implementation
1) Update Web Map Service 1.3 ATS to include all mandatory and optional elements of the standard (the ATS is submitted in the form of a Change Request, listed above)
2) Update Web Map Service 1.3 ETS to include all mandatory and optional elements of the standard
3) Enhance WMS 1.3 Reference Implementation to align with WMS 1.3 ATS enhancements

4) Establish WMS 1.3 DGIWG Profile ATS & ETS
5) Develop DGIWG WMS 1.3 Profile Reference Implementation
TEAM Engine
8) Enhance GUI and Engine to allow for profile testing, both on the fly (user identified) and automatic (to add "check box" selection or complete profile test to execute. To be accomplished in conjunction with WMS 1.3 ETS above.)
9) Maintenance

4.5.3 CITE Enterprise Viewpoint

The CITE thread provides a framework to test conformance of software components to OGC specifications. Past OWS initiatives have refined the process by which new Abstract Test Suites, and Executable Test Scripts and Reference Implementations are developed and made available to the OGC community. The foundation of this framework lies in strong collaboration between the CITE Thread participants and appropriate Standards Working Groups and the OGC Technical Committee as a whole. The CITE participants rely on the SWGs and the TC to provide reviews and to participate in beta test periods for all of the compliance testing components that are developed during an OWS project.

The overarching goal of CITE threads is to develop compliance tests where passing a compliance test means that a product should be interoperable with any other product that passes the same tests. In practice, compliance is not sufficient to guarantee interoperability, due to slight differences in implementations and imprecision in the original specification, combined with variances in performance that certain data sets may expose. This is why integration experiments and plugfests are key components of interoperability testing. However, compliance testing is a necessary, crucial step along this continuum.

Compliance tests developed in the CITE thread become the baseline of OGC's Compliance Test Program. The primary purpose of the testing program is to permit vendors and users to take advantage of the valuable standards that OGC has created, by providing a process whereby compliance can be tested. When compliance has been confirmed, participants who agree to the terms of the trademark license that accompanies this Program document may affix OGC's certification mark to their products, thus indicating to their customers that compliance with OGC Implementation Standards has been achieved and providing incentives for potential customers to preferentially purchase such products.

4.5.3.1 CITE Thread Process

The OGC Compliance Test Initiatives have come to use the following process to bring a new compliance test and reference implementations into the Compliance Test Program:

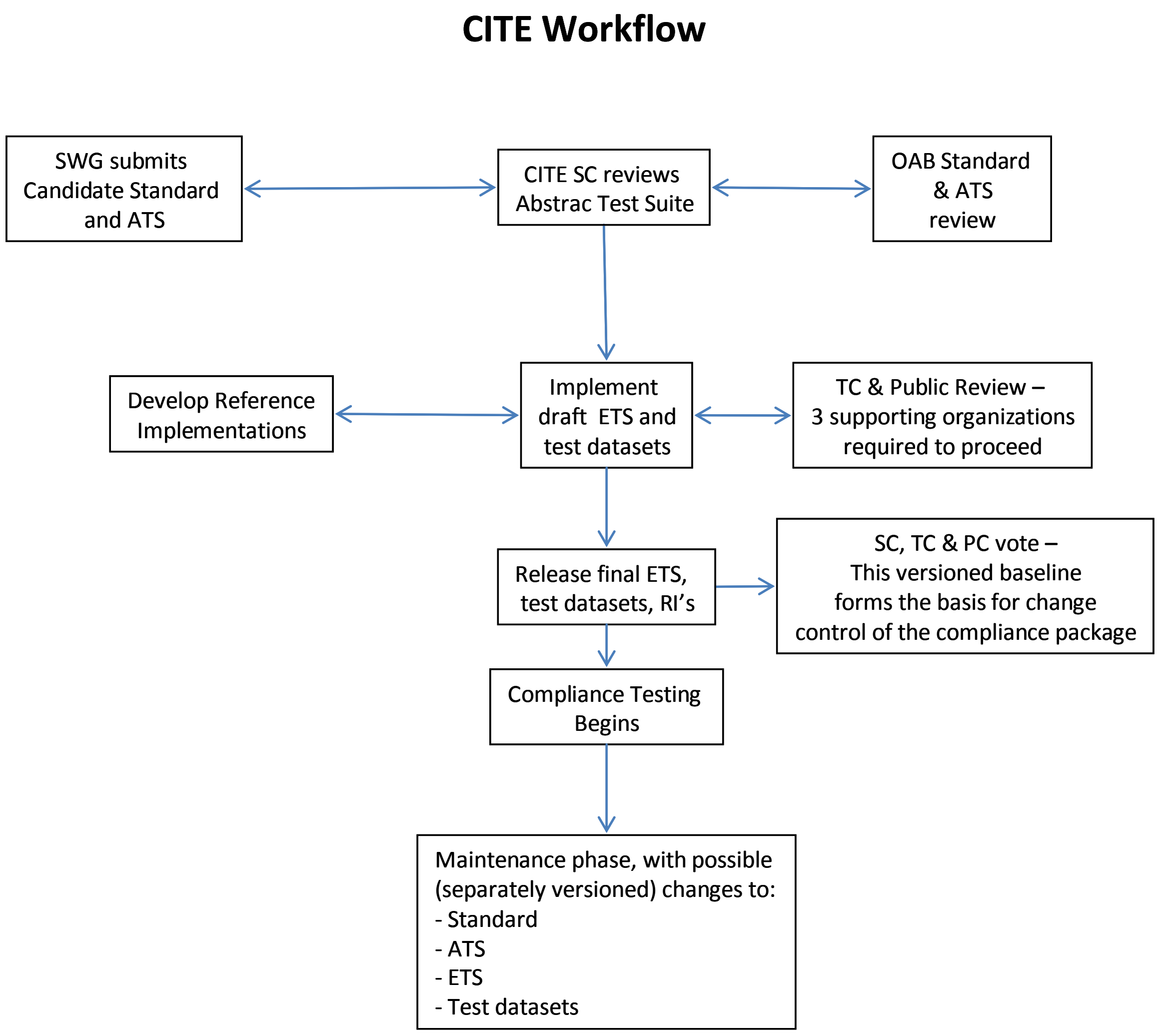


Figure 4-48. CITE Workflow

The process outlined above assumes that Abstract Test Suites are submitted when the specification is submitted. Most Abstract Test Suites have been developed, to date, by the CITE thread in OWS initiatives. Test Assertions are derived from the Abstract Test Suite. A suite of test assertions form an Executable Test Suites, which is then implemented, along with development and continual testing of the Reference Implementations.

Open reviews of Abstract Test Suites are announced to the TC and PC and sent directly to the spec editors and Standards Working Group chairs for a thirty day period during the course of the OWS project. An open beta period review of Executable Test Suites is also announced to the TC and PC and also held for a period during the course of the OWS project. Before a new compliance test can be voted to become an official OGC Compliance Test, three "supporting" implementations must pass 100% of the associated compliance tests. After three "supporting" implementations pass, the test will be provided to the TC and PC for a vote.

4.5.4 CITE Information Viewpoint

The primary information concepts in CITE are listed in the table below.

COMPONENT	DESCRIPTION
Compliance Test Engine	Test Evaluation And Measurement (TEAM) engine is a flexible platform for executing test scripts (ETS) written in CTL (defined below) to test a web service instance's compliance to a specification
Compliance Test Language (CTL)	Compliance Test Language is an XML grammar for documenting and scripting suites of compliance tests for verifying that an implementation of a specification complies with the specification. A suite of CTL files is installed in the compliance test engine, which executes the scripts and determines whether the implementation being tested passes or fails.
Abstract Test Suite (ATS)	Testable assertions extracted from specification document; defined as mandatory or optional; test cases are specified independently of any particular test procedure (ISO 19105, 4.4); may be used to create an ETS for a particular test harness
Executable Test Suite (ETS)	Compliance test which is executed by the Compliance Test Engine; a set of code which could be in the form of Java and/or CTL (Compliance Test Language) that tests the assertions defined by the ATS
Test Data	Static dataset provided through compliance test program; loaded into service implementation to be tested; this data is necessary for the ETS to test the service implementation
Reference Implementation	Open source, fully functional, 100% compliant implementation of a specification; in reference to which other implementations can be evaluated. OGC provides open source reference implementations to ensure maximum transparency of its specifications for both vendors and customers
Standards Package	includes the Standard, all appropriate schemas supporting the standard, the conformance clauses, and the ATS. This is versioned to represent a particular state of each component (standard, schemas, conformance clauses). Note that changes to Standards Package components may not necessitate a change in the ruling Standard
Test Package	includes the ETS, test data sets, and a reference to the Standards Package and TEAM Engine build. This is versioned to represent a particular state of each component (ETS, test data). Note that changes to Test Package components may not necessitate a change in the Standards Package.

Figure 4-49. List of Compliance Test Components

New compliance test scripts are written by first defining an Abstract Test Suite following guidance that is available in the document published as "Compliance Test and Interoperability Engineering Report" submitted to OGC pending documents in May 2008. This process includes following ISO 19105:2000 "Geographic information -- Conformance and testing". The next step is to define an Executable Test Suite by following guidance that was published in a paper titled "Compliance Test Language Discussion Paper" submitted 6 Sep 2006 to the Open Geospatial Consortium. This paper documents the Compliance Test Language as a whole and defines the structure of a test suite and tests in the test suite, the elements that can be used to form the test script code, the interfaces for producing Java classes to extend the basic CTL functionality and provides informative examples of test scripts and sample script output as well as guidelines on terminology used to properly document tests.

4.5.4.1 Writing Abstract Test Suites

An Abstract Test Suite describes the level of compliance testing being addressed within each Executable Test Script (i.e. identify whether or not a specific compliance test addresses all capabilities as described by particular web service specification such as WFS (Basic, Transactional, XLINK)) as well as whether the script is written to evaluate all mandatory and optional elements of a specification. Where limitations in the test script exist with respect to testing full specification compliance they are noted within the ATS.

The CITE participants in OWS5 worked closely with several experts in the OGC community, in particular the CITE Subcommittee, Arliss Whiteside and Charles Roswell, to reach consensus on the content and format for an Abstract Test Suite.

The conclusion was for the OGC Compliance Test Program to follow ISO 19105 - Geographic Information Conformance and Testing guidelines (Geographic information -- Conformance and testing). An XHTML template was defined. The following process was followed to develop ATS:

1. Read through the associated specification.

2. Document any testable assertions found in the specification.

3. Work with reference implementors and specification authors to refine the testable assertions and clarify any encountered ambiguities/issues.

4. Document any ambiguities or issues in the specification that arise during ATS development (for CRs).

5. Develop an XHTML ATS file using the template (ats-template.html) found in the following .zip file (developed by Richard Martell): https://portal.opengeospatial.org/twiki/pub/OWS5/CiteHome/ats-templates.zip.

6. Review the ATS document and solicit feedback from the reference implementers/specification authors.

7. Update the XHTML ATS based on feedback.

8. Use xhtml2ctl.xsl (in the previously referenced .zip file) to translate the XHTML ATS file into a CTL document skeleton that can filled in during ETS development.

9. Begin developing the ETS based on the ATS.

10. Execute ETS against the reference implementation.

11. Work with the reference implementers/specification authors to clarify any issues encountered during ETS development.

12. Document any ambiguities or issues in the specification that arise during ETS development (for CRs).

13. Refine the ATS if necessary based on ETS development.

14. Fix ETS issues.

15. Repeat steps 10 - 14 as necessary.

16. Document what is tested/what is not tested, namespaces, schemas, etc. with respect to the ETS – this could be more detailed than the ATS version because one can describe exactly how a test tests a particular assertion and why that test may or may not be able to test certain things.

4.5.4.2 Executable Test Suites

The OGC policy states that prior to approval as official OGC Compliance Tests, three implementations must "support" the ETS and be fully compliant. It is recommended that the CITE threads participants work to identify three supporting implementations early in the ETS development process to speed the time for approval.

An Executable Test Suite is a collection of test assertions that validate compliance with a particular Web service operation. A test is an XML file covering one assertion, or statement of required behavior that is derived from a specification. Tests typically contain one or more requests that are sent to the service begin tested, and XPath expression(s) that are evaluated against the results to determine whether the test passed or failed.

The test files also contain textual information describing the assertion being tested and how the test is performed. The engine automatically assembles this information to generate an assertions document describing each test the test suite is capable of performing.

In addition to the tests, the engine can be configured to prompt the user for information in the form of "scopes" which determine which tests will be executed for the session, or "variables", which can be used in the tests themselves.

4.5.5 CITE Computational Viewpoint

The TEAM Engine is an automated standards compliance testing software application. It is a cross-platform application that runs in a Java Virtual Machine. The TEAM Engine is an XSLT based engine with both a web and desktop interface. The following graphic depicts the compliance test engine and its interface to executable tests and services to be tested.

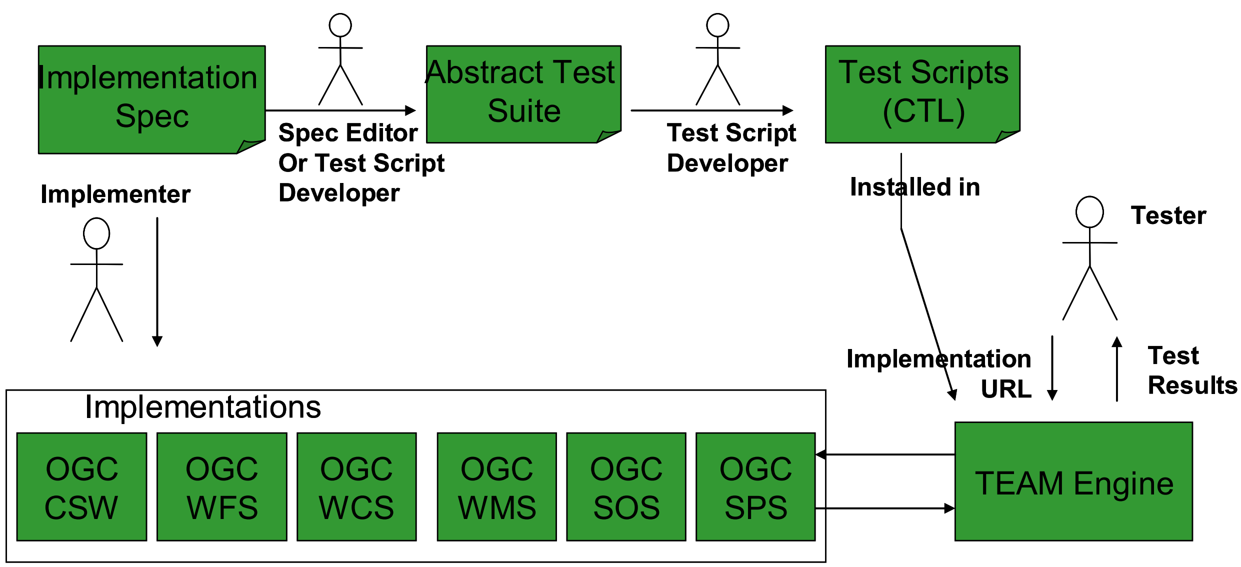


Figure 4-50. Interaction of Compliance Test Components

4.5.6 CITE Engineering Viewpoint

Figure 4-51 below depicts the architecture of the TEAM engine. In developing tests for the engine, participants have the option to extend the engine's functionality by writing custom functions and parsers. These may either be developed in XSL, or when needed, Java 1.4 with the Saxon XSL interpreter [http://saxon.sourceforge.net/].

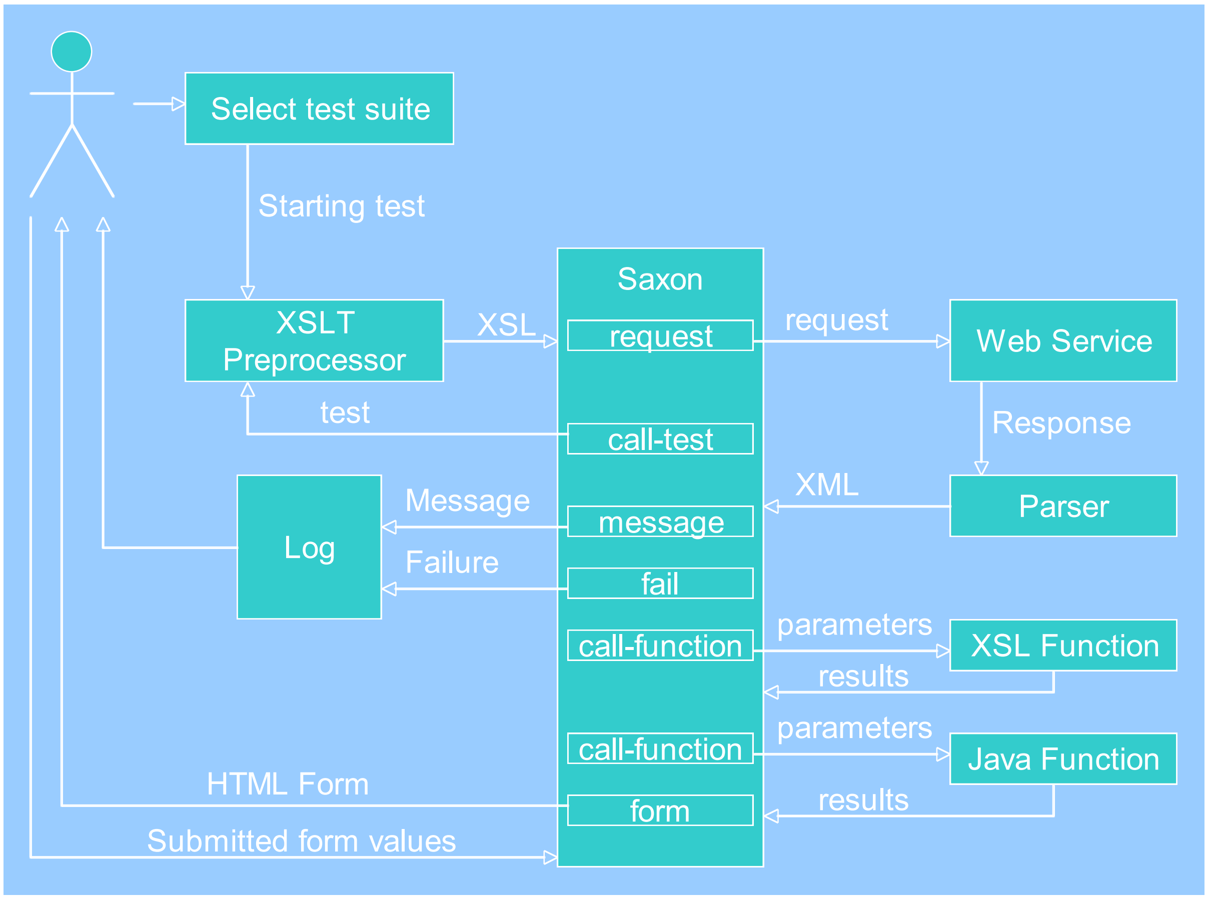


Figure 4-51. TEAM Engine internal architecture