

Open Geospatial Consortium

Publication Date: 2013-06-18

Approval Date: 2013-03-22

Posted Date: 2012-07-21

Reference number of this document: OGC 12-095

Reference URL for this document: <http://www.opengis.net/def/doc-type/per/ows9-wcs-opendap>

Category: Public Engineering Report

Editors: James Gallagher, Peter Baumann

OGC® OWS-9 Innovation - Coverages: Coverage Access (OPeNDAP) Study

Copyright © 2013 Open Geospatial Consortium.

To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>.

Warning

This document is not an OGC Standard. This document is an OGC Public Engineering Report created as a deliverable in an OGC Interoperability Initiative and is not an official position of the OGC membership. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard. Further, any OGC Engineering Report should not be referenced as required or mandatory technology in procurements.

Document type:	OGC® Engineering Report
Document subtype:	NA
Document stage:	Approved for public release
Document language:	English

Abstract

This document represents the OWS-9 OWS Innovations Coverage Access Study Engineering Report. It contributes knowledge based on the experience prototyping the WCS 2.0 Service – Access Innovations component, established in close collaboration with the OPeNDAP group. To this end, accessing a variety of coverage data types considering WCS 2.0 and DAP 2.0 interfaces have been implemented and demonstrated. The final result is a WCS 2.0 interface for the DAP 2.0 suite.

Keywords

ogcdoc, OGC document,ows9, ows-9, wcs, opendap

What is OGC Web Services 9 (OWS-9)?

OWS-9 builds on the outcomes of prior OGC interoperability initiatives and is organized around the following threads:

- **Aviation:** Develop and demonstrate the use of the Aeronautical Information Exchange Model (AIXM) and the Weather Exchange Model (WXXM) in an OGC Web Services environment, focusing on support for several Single European Sky ATM Research (SESAR) project requirements as well as FAA (US Federal Aviation Administration) Aeronautical Information Management (AIM) and Aircraft Access to SWIM (System Wide Information Management) (AAtS) requirements.
- **Cross-Community Interoperability (CCI):** Build on the CCI work accomplished in OWS-8 by increasing interoperability within communities sharing geospatial data, focusing on semantic mediation, query results delivery, data provenance and quality and Single Point of Entry Global Gazetteer.
- **Security and Services Interoperability (SSI):** Investigate 5 main activities: Security Management, OGC Geography Markup Language (GML) Encoding Standard Application Schema UGAS (UML to GML Application Schema) Updates, Web Services Façade, Reference Architecture Profiling, and Bulk Data Transfer.
- **OWS Innovations:** Explore topics that represent either new areas of work for the Consortium (such as GPS and Mobile Applications), a desire for new approaches to existing technologies to solve new challenges (such as the OGC Web Coverage Service (WCS) work), or some combination of the two.

- **Compliance & Interoperability Testing & Evaluation (CITE):** Develop a suite of compliance test scripts for testing and validation of products with interfaces implementing the following OGC standards: Web Map Service (WMS) 1.3 Interface Standard, Web Feature Service (WFS) 2.0 Interface Standard, Geography Markup Language (GML) 3.2.1 Encoding Standard, OWS Context 1.0 (candidate encoding standard), Sensor Web Enablement (SWE) standards, Web Coverage Service for Earth Observation (WCS-EO) 1.0 Interface Standard, and TEAM (Test, Evaluation, And Measurement) Engine Capabilities.

The OWS-9 sponsors are: AGC (Army Geospatial Center, US Army Corps of Engineers), CREA-GeoViQua-EC, EUROCONTROL, FAA (US Federal Aviation Administration), GeoConnections - Natural Resources Canada, Lockheed Martin Corporation, NASA (US National Aeronautics and Space Administration), NGA (US National Geospatial-Intelligence Agency), USGS (US Geological Survey), UK DSTL (UK MoD Defence Science and Technology Laboratory).

License Agreement

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD.

THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications.

This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable.

Contents

1. Introduction	5
1.1 Scope	5
1.2 Document contributor contact points.....	5
1.3 Revision history	5
1.4 Future work.....	7
1.5 Forward.....	7
2. References.....	7
3. Terms and definitions	7
4. Conventions	8
4.1 Abbreviated terms	8
5. Overview	8
6. Comparison of WCS 2.0 and DAP 2.0.....	8
6.1 The Conceptual Models	8
6.2 Metadata Digression	10
6.3 The Operational Models	10
7. Updating a WCS 1.x Server to Version 2.0	11
7.1 The OPeNDAP WCS 1.x Service Design	12
7.2 Semantic processing	12
7.3 The WCS 2.0 service.....	13
7.3.1 New Binary response formats.....	13
7.3.2 Issues in building the binary responses.....	14
7.3.3 GMLJP2	14
7.3.4 Satellite data are complicated	15
8. Future Work	15
8.1 Satellite swath data challenges.....	15
8.2 Semantic modeling of/for WCS 2.0	16
9. Conclusions	16

Figures

Figure 1. Main Components in the OPeNDAP WCS Service Implementation	13
--	-----------

OGC® OWS-9 Innovation - Coverages: Coverage Access (OPeNDAP) Study

1. Introduction

1.1 Scope

This document is the OWS-9 OWS Innovations Coverage Access Study Engineering Report. It documents the experience prototyping the WCS 2.0 Service – Access Innovations component, based on a close collaboration with the OPeNDAP group. To this end, accessing a variety of coverage data types, considering WCS 2.0 and DAP 2.0 interfaces, has been demonstrated. The final result is a WCS 2.0 interface built using a DAP 2.0 server as the primary data source, including data sources that require on-the-fly reprojection from satellite to geographic coordinate systems and returns types completely new to OPeNDAP's server software (although there may be DAP servers developed by others that return some of these types).

This OGC® document is applicable to OGC GML 3.2.1 Application Schema – Coverages [09-146r2] and OGC Web Coverage Service 2.0 – Core [OGC 09-110r4] and its extensions.

1.2 Document contributor contact points

All questions regarding this document should be directed to the editor or the contributors:

Name	Organization
Peter Baumann	Jacobs University rasdaman GmbH
James Gallagher	OPeNDAP

1.3 Revision history

Date	Release	Editor	Primary clauses modified	Description
2012-07-21	0.0.1	Peter Baumann	All	Created
2012-08-24	0.1.0	James Gallagher	5,6,7	Wrote about DAP, WCS and provided an overview for OPeNDAP's WCS service
2012-12-19	0.2.0	James Gallagher	5,6,7,8	Added information based on experience developing the prototype WCS interface

2013-02-20	0.2.1	Peter Baumann	All	Some reformatting, partly due to Word errors
-------------------	-------	---------------	-----	--

1.4 Future work

We have identified two major areas for future work, described more completely in Section 8. First, WCS 2.0 will benefit from more thorough standards for satellite swath data. Much of the important data provided by NASA for the ‘Haiti’ scenario of OWS-9 was in swath form and its use would have been far simpler if the WCS 2.0 standard more directly addressed it. Second, for much of the data served by OPeNDAP, metadata encoded in the data files can be used to build, automatically, the WCS metadata responses (including catalogs). However, to implement this requires more complete semantic modeling of the WCS 2.0 standard than time available in this test bed.

1.5 Forward

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

2. References

The following documents are referenced in this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

OGC 06-121r9 *OGC Web Services Common Standard*, version 2.0

OGC 09-146r1, *GML 3.2.1 Application Schema for Coverages*, version 1.0

OGC 09-110r3, *OGC®Web Coverage Service 2.0 Interface Standard Core*, version 2.0

Other documents are also referenced in the narrative. OGC documents are cited using OGC document identifiers in square brackets and are listed as such in the Bibliography. Non-OGC documents are cited using a number and can also be found in the Bibliography.

3. Terms and definitions

For the purposes of this report, the definitions specified in Clause 4 of the OWS Common Implementation Standard [OGC 06-121r3] and in OpenGIS® Abstract Specification Topic 6: The Coverage Type and its Subtypes, version 7.0 [07-011] (identical to ISO 19123:2005), and OGC GML 3.2.1 Application Schema for Coverages, version 1.0 [09-146r1] shall apply. In addition, the following terms and definitions apply.

3.1 OPeNDAP

The non-profit corporation based in Rhode Island, USA, maintaining DAP

3.2 DAP (Data Access Protocol)

The Data Access Protocol, version 2.0, a NASA Community standard computer protocol that uses HTTP to provide access and subsetting services for structured data

4. Conventions

4.1 Abbreviated terms

GMLCOV OGC GML 3.2.1 Application Schema – Coverages

WCS OGC Web Coverage Service [4]

5. Overview

This report provides information on both WCS 2.0 and DAP 2.0 web services. It is intended to help people choose which (or both) of these to use when making data available. Because web services are a kind of platform to which both applications and end users may interact, the choice of which to support is primarily dependent on the needs of those users. While this report cannot tell a data provider about their user population, its goal is to provide enough information about each web service, and its use in practice, so that a data provider can make an informed choice regarding the service(s) most apropos to their users.

In addition, the report includes a description of updating a WCS 1.x server so that it supports WCS 2.0 core, along with a common set of extensions. This is germane to the report because the example server uses a DAP server as the data store for coverages that are served by WCS.

6. Comparison of WCS 2.0 and DAP 2.0

While WCS 2.0 and DAP 2.0 (hereafter simply ‘WCS’ and ‘DAP’) are different in many small ways – as is the case with almost any two different protocols - the conceptual differences are few but significant. At their core, the two protocols differ in the conceptual models used for the data they serve.¹ Beyond this, there are a handful of operational details that separate the two protocols and while some of the operation details are result of having to choose between several equivalent options, others reflect genuine differences between the protocols and their prospective uses.

6.1 The Conceptual Models

The DAP data model [2] is based on data types found in general purpose programming languages and provides a domain-neutral conceptual model. It contains scalars and N-

¹ In the past, the terms ‘data model’ or ‘information model’ have been used to describe what we choose to call the ‘conceptual model’ here.

dimensional arrays of the common data types such as integers, floating point numbers and strings. It also contains a constructor type that is used to form simple aggregate types and types for tabular data, both of which can be used to recursively define new types. Lastly it contains a type used to represent discrete rectangular coverages. All but the last of the data types found in DAP have clear counterparts in the world of general-purpose programming languages, and discrete rectangular coverages (i.e., a subset of sampled functions) correlate nicely to the C++/STL map class (or Perl's associative array or Python's dictionary). Thus, DAP's conceptual model provides a way to express any data that can be expressed using a general purpose programming language. That is a considerable upside but brings with it the drawback that similarly organized data can be expressed in large number of syntactically different forms. These syntax differences can hide the semantic similarity of the data – a similarity that is obvious to a person but which can be inscrutable to a program.

The WCS data model is based on sampled functions, which map a domain in N-dimensions onto a range. In practice, the domain is either spatial in latitude and longitude or spatiotemporal (latitude, longitude and time). While it is not a requirement that the domain of the sampled functions made available by WCS is spatiotemporal, this is the typical use pattern and many of the response formats and (most importantly) client applications expect this. Note that the sampled functions are also known as 'coverages' and WCS provides a way to access all of the discrete coverages defined in the OGC Abstract Specification Topic 6 [OGC 07-011]. Because coverages do not include the capability to define recursive types, they lack the capability to encode information with arbitrarily complex lexical structure unless additional conventions are adopted beyond what the coverage model(s) provide. In trade for this limitation on the conceptual model, WCS can achieve a greater degree of uniformity in the data it serves than DAP because the latter allows semantically equivalent information to be represented in syntactically different forms.

What these two different conceptual models represent, in addition to their differing views on data organization, are two slightly different sets of operations. This follows closely from the abstract datatype or object-oriented notion that datatypes are defined not only by their structure but also by the kinds of operations they provide for those structures. In addition to simple access to the entire coverage (WCS) or variable (DAP), each of the protocols provides ways to subset and sample those data values. For array cutouts ("domain subsetting") WCS provides trimming and slicing operations that can extract regions and reduce the rank of coverages, respectively. DAP provides equivalent operations. DAP also provides predicate-based subsetting of tabular data, which does not have a direct match to an operation in WCS². Such predicates allow the substitution of selected values by other values, such as zero or null, thereby effectively "masking" data sets. For the extraction of bands (or variables) from a coverage, WCS offers the Range Subsetting extension [OGC 12-040]. Any number of bands can be selected and recombined; while this is independent from any encoding format, the final encoding of the coverage result obviously may constrain the number of band that can be combined (see discussion about GeoTIFF and JPEG2000).

² However, such operations are part of the functionality the Web Coverage Processing Service (WCPS) [OGC 08-068r2] provides.

6.2 Metadata Digression

DAP provides a container for metadata in much the same sense as it provides structured containers for data. Metadata are stored as a number of name-type-value tuples that are either bound to a dataset (aka ‘global attributes’), to a specific variable within a dataset, or in other ‘attribute containers.’ The datatypes supported are all of the cardinal types available for data (integer, float, etc.) and the values may be either scalar or vector. In addition, a structure type is supported, and as with data, this may be recursively nested. The metadata model has three restrictions not placed on the data model: Relations (i.e., tabular data), Grids and N-dimensional objects are not supported.

Within DAP’s metadata model, a wide range of metadata can be encoded.

However, DAP does not extend its metadata model to provide support for specific domains. Instead, DAP relies on other standards and/or conventions to provide the semantic mapping between sets of terms, their acceptable values and the meanings of those in the context of a given domain (e.g., Oceanography, Aviation, etc.). Because DAP requires no particular vocabulary, it has no associated minimal set of metadata. While every data source must have the names and types of all accessible data variables clearly defined, there are no such requirements for metadata. Metadata may be present, it may follow a given standard, it may be absent, or it may represent a collection of values drawn from a number of standards and conventions.

While DAP’s metadata model can be frustrating because it lacks uniformity, this also enables the protocol to easily adapt to a variety of different domains. However, one cost is that the protocol requires those domains to have useful metadata standards and/or conventions, and that those communities of practice form and use those standards/conventions. A related issue is that conformance testing for DAP cannot provide a measure of a given server’s metadata conformance.

At a practical level, WCS encodes metadata specific to the geospatial and temporal domain in the protocol, in both the *GetCapabilities* and *DescribeCoverage* metadata responses and in the *GetCoverage* request and the kinds of responses it returns. Note that it is not strictly true that WCS requires geospatial and temporal metadata, because it does support a range of coordinate systems. However, in practice client applications expect WCS service endpoints to support data expressed as coverages with domain in Latitude-Longitude-Time and, of the three main response forms for WCS 2.0 (GeoTIFF, GMLJP2 and NetCDF), two are explicitly designed for geospatial data.

6.3 The Operational Models

Beyond these differences in the conceptual models, the operational models of WCS and DAP differ. DAP defines one required return format for metadata³ and one for data.

³ Technically, there are two metadata responses in DAP 2.0: One is the dataset’s datatype definition and the other is the attribute-based metadata. However, these are provided as two distinct responses because of quirks in the early development of the protocol. Many servers provide a third metadata response that combines these.

These responses could be classified as ‘proprietary’⁴, at least in the sense that they are unique to DAP. All servers must support these responses. Most DAP 2.0 servers also provide other kinds of responses that package data so common client programs can use them or so that people can save the responses in disk files. Two examples are CSV (comma-separated values ASCII) and netCDF3 files. Note that there’s no mention of a catalog of datasets or inventories.⁵ DAP provides no support for catalogs of any kind. Early DAP servers provided information about data objects by presenting them as files in the directory pages returned by HTTP daemons. Later, many DAP servers adopted the THREDDS catalog format and most current DAP servers implement the THREDDS catalog protocol.

WCS provides no one required response format. The core specification defines a data response based on GML where a GML document either contains data values as ASCII in the XML markup or functions as an envelope for a data file that contains binary data. Various annex specifications provide for additional response formats (e.g., netCDF3, GMLJP2 or GeoTIFF) files. Note that in WCS, the data and metadata for a specific coverage are bound to a single response. In addition to the data response (aka the response to a *GetCoverage* request), WCS provides two other responses. The *GetCapabilities* response provides general information about a particular server including both its operation characteristics and a catalog of available coverages. The *DescribeCoverage* response provides information about a specific coverage listed in the *GetCapabilities* response.

One interesting aspect of WCS (both 1.x and 2.0) is that the *GetCapabilities* response provides both server metadata and a catalog of available coverages. However, there is considerable latitude in how a particular site providing WCS access can address the ‘granularity’ issue. That is, there is no requirement in the specification that all coverages sharing a common domain, for example, be accessed through a single service endpoint. Implementers are free to use as many ‘servers’ as they want. While this fits in well with the web model that each endpoint is ‘just a URL’ it also means that two servers can catalog/organize 10 coverages in two very different ways, one of which includes all ten in a single *GetCapabilities* response and another with ten distinct responses, each listing exactly one coverage. In fact the WCS 1.x from OPeNDAP and Unidata take these two approaches, respectively [5].

7. Updating a WCS 1.x Server to Version 2.0

This portion of the ER describes our experiences in updating our open-source WCS 1.x server so that it supports the WCS 2.0 specification. First we describe how our WCS service was implemented and then we describe how the WCS 1.x responses were modified to match the WCS 2.0 specification and the OWS-9 testbed clients’ expectations. Our service was implemented so that the distinction between the ‘compute

⁴ Since DAP was developed as part of an open project and its documentation and reference implementation are open-source, this word is not entirely correct.

⁵ The word ‘inventory’ is used in the same sense as NASA uses the word; to describe collections of files that, taken together, form a single dataset from a person’s perspective. Each file typically differs in space (tiled datasets) or time (time-series datasets). The ‘inventory’ provides a mapping between the (typically) spatiotemporal domain and files.

engine’ that extracted data from one or more sources and built the binary responses was distinct from the software that built the XML response documents defined by WCS. In fact, in its initial version, in response to a *GetCoverage* request the service only returned an XML Coverage document which contained a URL that was then dereferenced by a client to access the binary (data) component of the response. The response was not actually generated until the (second) URL was dereferenced. In later versions of the service, the data object (a netCDF3 file) was included in the *GetCoverage* response, but the nature of the service’s operation reveals the extent to which the XML documents were built as distinct from the actual data returned. During the OWS-9 effort, we focused almost exclusively on the response formats (implementing two new formats for our server –GeoTIFFandGMNLJP2 – and moderately complex transformations of data so the testbed clients could use them.

7.1 The OPeNDAP WCS 1.x Service Design

The OPeNDAP WCS 1.x web service was designed as a platform to experiment with providing a WCS interface for data served using DAP. As such, the web service assumes that there exists a DAP server that can provide certain kinds of data responses using nothing more than a URL. In Figure 1, the WCS web service provides the interface that client software sees. Through this interface, clients ask for the three different WCS responses. Those responses are built using information from both a ‘Coverages Information’ data store and one or more DAP servers. The coverages information data store provides information used in all three of the responses while the DAP server provides information directly to the WCS interface only for the *GetCoverage* response. Indirectly, however, the DAP server is the source for all of the information since its metadata responses are used to populate the coverages information data store. An offline process performs this metadata extraction and filtering step.

7.2 Semantic processing

The main research goal of the WCS 1.x development was to demonstrate using semantic web technology to translate metadata encoded using a variety of standards and conventions into the representation needed for WCS 1.x.⁶ Because DAP servers provide metadata, but not *predictable* metadata, the project explored describing different standards and conventions using ontologies expressed in RDFS and using crosswalks of those with an ontology for WCS 1.x. The crosswalks provided a way to translate metadata expressed so that it could be used to build WCS 1.x responses with little or no configuration on the part of the data provider. Funding limitations meant that ontologies for only CF and UDD (in addition to one for WCS) were written. However, the resulting software proved very easy to run for data that follow those particular schemes. While this goal is not particularly important in the context of this ER or WOS-9, it demonstrates that there is often significant semantic overlap between the information served by both WCS and DAP.

⁶ The software was developed in collaboration with Columbia University.

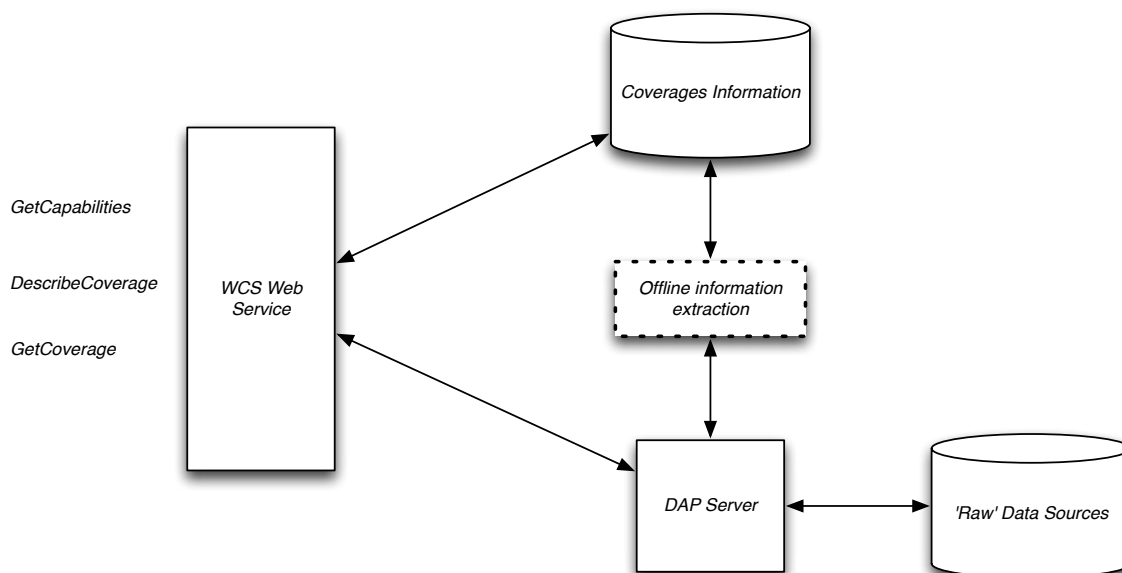


Figure 1. Main Components in the OPeNDAP WCS Service Implementation

7.3 The WCS 2.0 service

While the WCS 1.x service development centered on using semantic web technology to build a metadata translation engine, the WCS 2.0 development focused on supporting WCS clients and the needs of the GIS audience. As noted in the *future work* section of this ER, we will likely revisit the *GetCapabilities*, ..., responses for WCS 2.0 and extend the semantic processing engine to support them. For the OWS-9 testbed, we used hand-written XML for both the *GetCapabilities* and *DescribeCoverage* responses, an approach that will not scale up to many of the dataset we would like to support. However, this approach left us free to devote time to new binary response formats (GeoTIFF and GMLJP2) and more compute-intensive data processing. An additional benefit is that the handwritten XML provides an easy roadmap for the semantic engine, something that will speed its implementation.

7.3.1 New Binary response formats

Initially we provided responses to the OWS-9 clients using the one WCS-conformant binary response our server (Hyrax) already supported – NetCDF3. Unfortunately, none of the testbed clients could read that format. While that may have changed since the initial end-to-end tests, it is an important result because it demonstrates that the majority of GIS clients really expect data to be in representations they can easily process.⁷ We began developing both GeoTIFF and GMLJP2 responses so those clients would be able to read from our server. We found that there is relatively less but sufficient open source support for these formats than more common formats such as GIF or PNG. Our responses are built using our DAP server (Hyrax) to build a data response using the DAP Grid datatype

⁷ Of course, nothing is so simple; ArcGIS will read netCDF files and can read directly from DAP servers.

(a type that is equivalent to the OGC Discrete rectangular coverage [OGC 07-011]) and then filter that to package the data in a GeoTIFF or GMLJP2 file that is then streamed back to the WCS client (we can also return a URL that will return the file when dereferenced).

7.3.2 Issues in building the binary responses

Both of GeoTIFF and GMLJP2 are built using the GDAL [6] library – an open source library that no doubt sees wide use in implementations of the OGC web services specifications. For GeoTIFF, GDAL provides good support although the data sources we are providing tend to use very large or very small values to indicate missing data values and these throw the GeoTIFF driver off when it transforms data to build a photometric image for the response. We found it necessary to compute a histogram for the image data and change the missing data values in the data to something ‘close to’ the valid data values.

Note that OGC 12-100[14] and OGC 12-101 [15] describe GeoTIFF responses in the context of WCS 2.0.

A more persistent problem is how to best represent vector data using GeoTIFF. Nominally, these files hold either one, three, four or N bands of data. Clients typically will interpret a one-band image as a photometric grayscale image and a three- or four-band image as a RGB or RGBA image. In other cases client software makes various assumptions about what the different bands mean. For example, the *gdalinfo* program will assume that a two-band image holds on band of gray scale data and one of *unknown* data. A very relevant question is how best to represent wind vectors using GeoTIFF— a reasonable way would be to encode the vector components using two bands, but which clients would understand this?

7.3.3 GMLJP2

Building GMLJP2 responses proved more challenging and the response proved more limiting, at least in as much as the GMLJP2 files built using GDAL go. There are five drivers available for GDAL that will build JPEG2000 files [7]. However, of these three drivers based on ECW, Kakadu and the MrSID SDK are commercial software that requires licensing fees. We wanted to see how far we could get using free software, so we tried both the Jasper and OpenJPEG2.0 libraries [12]. The Jasper-based driver does not produce GMLJP2, only JPEG2000, so we switched to the driver based on OpenJPEG. This driver will build a GMLJP2 file, but only using GDAL 1.10 or later.⁸ We encountered a number of issues building this software and given the time spent, it might actually have been cheaper to use one of the commercial libraries. Of course, if GMLJP2 gains acceptance, and the fact that GDAL already supports several drivers that build these kinds of files indicates that is going to be the case, then this driver software will likely have its rough edges burnished.

⁸ AS of 12/18/12, this has not been released, but is available via GDAL’s source code revision control system.

A deeper issue with GMLJP2 is the representation of multi-band data. The GDAL driver based on OpenJPEG2.0 only builds responses that have one or three bands; it will not build a response with N bands. Because of this limitation on the number of bands in a GMLJP2 response, wind vector data cannot be easily represented, at least not using the tools we adopted for this project. This limitation seems to fly in the face of the 2005 OGC definition of GMLJP2 [13].

7.3.4 Satellite data are complicated

NASA provided a number of datasets for OWS-9 useful for hurricane forecasting [8][9]. However, essentially all of the datasets provided were in satellite coordinates. That is, NASA either categorized these data as Level 1B or Level 2 data, which means that they have not yet been projected onto a geographic coordinate system, among other things [11]. The problem of building level 3 datasets from the level 1B or level 2 data is known to be hard and recent work presented in preliminary form at the Fall 2012 AGU meeting indicates that different developers can come up with valid processing software that arrives at different values using the same initial data [10]. Unfortunately, this was requirement was discovered late in the OWS-9 timeline and we were not able to devote the time needed to do a thorough job of providing on-the-fly projections of these data.

A complete discussion of the development of processing algorithms for this is beyond the scope of this paper, but simply projecting the data values is not sufficient. In addition to reprojecting the data onto a geographical coordinate system, physical parameters of the sensors involved in collecting the data, environmental factors that may affect the accuracy of those sensors and errors flagged for certain specific data samples are a partial list of things to be considered. The effort required to *correctly* build the level 3 data is substantial.

Although it is complicated to process these lower-level data products and produce responses that can be easily used by more general-purpose software, it appears that this is exactly the correct role for WCS. WCS provides an excellent interface for accessing these higher-level data products because of the semantic and transport-format limitations it imposes.

8. Future Work

8.1 Satellite swath data challenges

The representation of satellite swath data in WCS would benefit from further work. Such data are generally encoded as three two-dimensional arrays, one for the latitude values, one for longitude and one for the measured value (i.e., this maps exactly onto the coverage notions of domain and range). As Raskin, et al. [17] point out, swath data can be thought of as three raster images with the dimensions of ‘along the satellite track’ and ‘across the track.’ While it should be possible to encode these data in a number of the response formats that WCS 2.0 supports, this is not enough. Imagine that a GeoTIFF file were used for these data. The resulting file would contain three bands (e.g., latitude, longitude and surface temperature) but would instead be interpreted by most client software as the RGB bands of a single raster image.

Further complicating the swath data issue is that there are several different kinds of data products that are collectively called ‘swath data’ and potential variations between different swaths (i.e., passes) that cover the same area. NASA categorizes satellite data using a scale from 0 to 4 [18](but realize that NOAA also uses ‘levels’ for these data but with subtle differences in their meanings) and either level 1b or level 2 could be considered ‘swath data’ for the purposes of this discussion. Furthermore, depending on intended use, some passes may be more suited than others. An example can be seen with MODIS data. The AQUA satellite has a MODIS sensor and flies in a sun-synchronous orbit with a period of 99 minutes covering the entire planet every 36 hours [19]. Any particular area on the Earth may have coverage from a given scan, but the usefulness of that coverage will also depend on whether the pass is an ascending (south to north pole) or descending as well as on the sun angle at the time of the pass (low angle reveal more surface relief while high angles provide more uniform coverage).

In order to more completely represent swath data, WCS will need to not only provide a format for transmitting both those data and the ancillary information like sun angle, orbit type, etc. that most users need. Note that others have examined the issues surrounding this problem [17] and have proposed a solution for accessing swath data in ArcGIS, so this is both a known problem and one with, at least, candidate solutions.

8.2 Semantic modeling of/for WCS 2.0

During the course of this work we made several choices regarding implementation techniques. OPeNDAP had previously developed a WCS 1.x implementation in conjunction with researchers at Columbia University where the *GetCapabilities* and *DescribeCoverage* responses were completely machine generated using metadata retrieved from the DAP server. For work on WCS 2.0 we tried a completely different approach; all of those documents were hand written. Of course, that makes for a simpler bit of software to write! But our motivation was to focus the OWS-9 resources on more sophisticated source and response formats. Having done so, we expect that the convenience of having the WCS server automatically build the metadata responses is one our users will want because it simplifies configuration of large collections. The most important characteristic of this feature is that it is transforming metadata that *already* exist in digital form and often are encoded in data files (e.g., HDF5 files). Not using this information is wasteful of data provider’s time; they have already supplied the information once. Semantic tools provide the means to transform this already-encoded information from the myriad of standards in which it appears to a form that can be used with WCS.

9. Conclusions

DAP and WCS compliment each other. The wide range of data that DAP can represent makes it well suited as an abstraction for a data store. Because the interface supports subsetting, it is fairly easy to use DAP to ‘reach into’ files or other datasets and access their individual components while only needing to understand the DAP interface syntax and semantics (and not those of the often *ad hoc* local representations). WCS

complements this by providing a uniform and highly specialized interface for geospatial data.

The WCS specification supports a number of distinct responses, but in practice WCS clients expect data to be returned using file formats that are commonly used with GIS systems. Thus while WCS supports netCDF and other responses, when implementers provide GeoTIFF and GMLJP2 responses, the resulting service is usable by a wider range of clients.

There are complex issues that arise when level 1 or 2 satellite data are used to build WCS responses. These issues include georeferencing the data of course, but also include many factors that depend vary according to sensor, space-borne platform and measured parameter. Future work is needed to develop software that returns higher quality responses.

Bibliography

- [1] OGC 00-014r1. “Guidelines for Successful OGC Interface Standards.”
- [2] Gallagher, J., Potter, N., Sgouros, T., Hankin, S., and Flierl, G. “The Data Access Protocol — DAP 2.0.” NASA ESE-RFC-004.1.1. Oct. 10, 2007. Accessed on 12/18/12 at: <http://earthdata.nasa.gov/sites/default/files/esdswg/spg/rfc/ese-rfc-004/ESE-RFC-004v1.1.pdf>.
- [3] OGC 07-011. “The OpenGIS Abstract Specification Topic 6: Schema for coverage geometry and functions.” Version 7.0. Jan 30, 2006.
- [4] OGC 09-110r4. “OGC Web Coverage Service 2.0 Interface Standard – Core.” Version 2.0.1. May 5, 2012.
- [5] Private communication with Ethan Davis, Technical lead for TDS. 17 Dec 2012.
- [6] The Geospatial Data Abstraction Library. Accessed on 12/18/12 at <http://www.gdal.org/>.
- [7] GDAL Raster Formats. Accessed on 12/18/12 at: http://www.gdal.org/formats_list.html.AIR
- [8] Atmospheric Infrared Sounder (AIRS). Accessed on 12/18/12 at: <http://airs.jpl.nasa.gov/>
- [9] “NASA Study Could Improve Hurricane Strength Forecasts.” Accessed on 12/18/12 at: <http://www.jpl.nasa.gov/news/news.php?release=2012-373>.
- [10] Smith, D. Mears, C., Hilburn, K., Ricciardulli, L., Wentz, F., Schulte, R. “Contribution of Data Set Construction Methodology to Data Set Uncertainties.” AGU FallMeeting 2012, IN11A-1461.
- [11] Wikipedia. “Remote Sensing: Data Processing Levels.” Accessed on 12/19/12 at: http://en.wikipedia.org/wiki/Remote_sensing#Data_processing_levels.
- [12] OpenJPEG, Accessed on 12/19/12 at: <http://www.openjpeg.org/>.
- [13] OGC 05-047r2. “GML in JPEG 2000 for Geographic Imagery (GMLJP2) Implementation Specification.” Version 0.3.0. June 26, 2005.
- [14] OGC 12-100. “OGC® GML Application Schema - Coverages - GeoTIFF Coverage Encoding Extension.” Version 0.0.5. Oct. 24, 2012.

- [15] OGC 12-101. “OGC® Web Coverage Service 2.0 Interface Standard - GeoTIFF Coverage Encoding Extension.” Version 0.0.7. Oct. 29, 2012.
- [16] OGC 12-040. “OGC® Web Coverage Service Interface Standard - Range Subsetting Extension.” Version 0.0.4. Nov 30, 2012.
- [17] Raskin, R., Kumler, M. and Kopp, S. “Representing Satellite Swath Data in ArcGIS.” American Geophysical Union, Fall Meeting 2007, abstract #IN13A-0897.
- [18] Wikipedia. “Remote Sensing.” Accessed on Feb 20, 2013 at http://en.wikipedia.org/wiki/Remote_sensing.
- [19] NASA. “Aqua Project Science.” Accessed on 20 Feb, 2013 at <http://aqua.nasa.gov/about/instruments.php> and <http://modis.gsfc.nasa.gov/about/srca.php>