

Open Geospatial Consortium

Approval Date: 2013-01-18

Posted Date: 2012-12-08

Publication Date: 2013-06-18

Reference number of this document: OGC 12-162r1

Reference URL for this document:
http://www.opengis.net/def/doc-type/per/ows9_wcs_conformance_testing

Category: Public Engineering Report

Editors: Jinsongdi Yu, Peter Baumann

OWS-9 WCS Conformance Testing Engineering Report

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 Engineering Report was prepared as a deliverable for the OGC Web Services, Phase 9 (OWS-9) initiative of the OGC Interoperability Program. The document presents the work completed with respect to the Conformance & Interoperability Testing & Evaluation sub-thread within OWS-9.

This Engineering Report describes and evaluates the specification of WCS 2.0 core corrigenda and extensions' Abstract Test Suite (ATS) and the implementation of ETS for use within an OGC SOA processing chain.

Keywords

ogcdoc, ogc document, wcs, ats, cite, ctl

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), CREAf-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		Page
1	Introduction.....	1
1.1	Scope	1
1.2	Document contributor contact points	1
1.3	Revision history.....	1
1.4	Future work	2
1.5	Forword	2
2	References.....	2
3	Terms and definitions	3
4	Conventions	3
4.1	Abbreviated terms	3
4.2	UML notation	4
4.3	Used parts of other documents	4
5	ER Topic overview	4
6	WCS 2.0 series: Core, extensions and EO-WCS profile	5
7	Reference Implementations: Rasdaman and EoxServer	6
7.1	How GetCapabilities related with the modules.....	6
7.2	How DescribeCoverage related with the modules	7
7.3	How getCoverage related with the modules.....	8
7.4	Arbitrary coverage processing beyond getCoverage.....	9
8	Conformance testing and enhancement	10
8.1	Test suite update.....	10
8.2	Capabilities Control Testing (CCT)	10
8.3	Validation of diverse coverage encodings.....	11
8.4	Proposed TEAM enhancements	12
8.4.1	Address multi-dimensional coverage encodings	12
8.4.2	Core and extensions handling	12
8.4.3	Asynchronous web service	13
9	Conclusion	13

WCS Conformance Testing

1 Introduction

1.1 Scope

The OWS-9 thread “Conformance & Interoperability Testing & Evaluation” has seen a series of important and useful progress in the Web Coverage Service (WCS) 2.0 standards suite. Corrigenda on both the coverage data model have been accomplished, abstract test suites (ATS) have been established for emerging WCS extension standards, and executable test suites (ETS) have been implemented for both existing and emerging WCS specifications. Test suites established for WCS 2.0 core corrigenda and extensions, such as format encodings, band sub-setting, scaling, interpolation, and WCPS, have been developed based on the revised version of the Testing, Evaluation, and Measurement (TEAM) engine in CTL. Additionally, the improvement of the ETS for the EO-WCS candidate standard continues the further refinement of EO-WCS for smooth interaction with the WCS 2.0 modular specification (core and extension packages). Suggestions for TEAM updates have been investigated, as the current TEAM engine turned out insufficient when it comes down to accurate testing – after all, WCS defines semantic interoperability down to the granularity level of single pixels.

1.2 Document contributor contact points

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

Name	Organization
Jinsongdi Yu	Jacobs University Bremen, rasdaman GmbH p.baumann<at>jacobs-university.de
Peter Baumann	Jacobs University Bremen, rasdaman GmbH j.yu<at>jacobs-university.de

1.3 Revision history

Date	Release	Editor	Primary clauses modified	Description
2012/12/8	0.0.1	JY, PB	All	Created

1.4 Future work

Investigate the common access model for the parsed coverage in formats other than pure GML.

1.5 Foreword

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

NOTE This OWS Common Standard contains a list of normative references that are also applicable to this Implementation Standard.

OGC 07-011. The OpenGIS® Abstract Specification Topic 6: Schema for coverage geometry and functions

OGC 09-153r1. OGC® Web Coverage Service 2.0 Primer: Core and Extensions Overview (2.0)

OGC 09-146r2. OGC® GML Application Schema - Coverages

OGC 09-110r4. [OGC® WCS 2.0 Interface Standard- Core: Corrigendum \(2.0.1\)](#)

OGC 09-147r2. OGC Web Coverage Service 2.0 Interface Standard -KVP Protocol Binding Extension

OGC 09-148r1. OGC Web Coverage Service 2.0 Interface Standard –XML/POST Protocol Binding Extension

OGC 09-149r1. OGC Web Coverage Service 2.0 Interface Standard –SOAP Protocol Binding Extension

- OGC 12-100. OGC® GML Application Schema - Coverages - GeoTIFF Coverage Encoding Extension
- OGC 12-108. OGC® GML Application Schema - Coverages - JPEG2000 Coverage Encoding Extension
- OGC 11-010. WCS 2.0 Interface Standard – NetCDF Encoding Format Extension
- OGC 10-140. OGC® Web Coverage Service 2.0 Interface Standard - Earth Observation Application Profile
- OGC 11-153. OGC® Web Coverage Service Interface Standard - CRS Extension
- OGC 08-068r2. Web Coverage Processing Service (WCPS) Language Interface Standard
- OGC 08-070. Web Processing Service (WPS) and Web Coverage Processing Service (WCPS)
- OGC 12-039. OGC® Web Coverage Service Interface Standard - Scaling Extension
- OGC 12-040. OGC® Web Coverage Service Interface Standard - Range Subsetting Extension
- OGC 12-049. OGC® Web Coverage Service Interface Standard - Interpolation Extension

3 Terms and definitions

For the purposes of this report, the definitions specified in Clause 4 of the OWS Common Implementation Standard [OGC 06-121r9] and in OpenGIS® Abstract Specification Topic 6: Schema for coverage geometry and functions shall apply. In addition, the following terms and definitions apply.

4 Conventions

4.1 Abbreviated terms

OGC	Open Geospatial Consortium
OWS	Open Web Service
OWS-9	OGC Web Services Initiative, Phase 9
WCS	Web Coverage Service
WCPS	Web Coverage Processing Service

EO-WCS OGC® Web Coverage Service 2.0 Interface Standard - Earth Observation Application Profile

ATS	Abstract test suites
ETS	Executable test suites
CTL	Conformance testing language
TEAM	Testing, Evaluation, and Measurement engine
RI	Reference Implementation
DRB	Data Request Broker
GPL	The GNU General Public License
MPL	MOZILLA PUBLIC LICENSE
XSL	The Extensible Stylesheet Language Family

4.2 UML notation

Diagrams that appear in this standard are presented using the Unified Modeling Language (UML) static structure diagram, as described in Subclause 5.2 of [OGC 06-121r9].

4.3 Used parts of other documents

OGC 09-153r1. OGC® Web Coverage Service 2.0 Primer: Core and Extensions Overview (2.0)

OGC 09-110r4. [OGC® WCS 2.0 Interface Standard- Core: Corrigendum \(2.0.1\)](#)

5 ER Topic overview

This report summarizes OWS-9 achievements on the WCS 2.0 series' testing, aiming to improve WCS implementation quality and achieving WCS conformance branding.

Overall, the following core results have been established in 2012:

- There are at least five open-source WCS Core implementations available now: rasdaman, MapServer, GMU, OPeNDAP, and GeoTrellis. Additionally, there is the EOxServer open-source EO-WCS implementation.
- Corrigenda on the OGC coverage data model (GMLCOV) and service model (WCS) have been devised, implemented, tested, and adopted, leading to GMLCOV 1.0.1 and WCS 2.0.1, resp.
- A series of new WCS Extension specifications are in the OGC pipeline, waiting to be processed (hopefully soon) by TCC and OAB.

- Fine-grain conformance tests have been established (ATS) and implemented (ETS).

The latter item is the main focus of this Engineering Report. ATSs are specified according to the requirements in the specifications and provides a basis for developing requirement-consistent ETSs. The ETSs is specified in OGC Compliance Test Language (CTL). Furthermore, dependencies are addressed to improve confidence on an implementation choosing to implement its set of preferred modules.

Currently, OGC Compliance & Interoperability Testing & Evaluation (CITE) program uses OGC's Test, Evaluation, and Measurement (TEAM) Engine, which is fed with the specified CTL scripts to test these implementations.

6 WCS 2.0 series: Core, extensions and EO-WCS profile

Any WCS 2.0 implementation is required to implement the Core, at least one protocol extension for the communication between clients and servers, and one encoding extension for the coverage returned; further, an implementation can choose to support further requirements classes from the extensions existing. Thereby, the mechanism achieves an individual, yet interoperable functionality package based on the same abstract coverage model, specifically, OGC07-011, which gives an abstract model and is further refined to become a practical usable encoding, such as the work which has been done on different GML coverage encodings (e.g., GeoTIFF, NetCDF, and JPEG2000 extensions). Further service extensions, such as Scaling, Interpolation, CRS and WCPS extensions provide additional functionality which may be added to a WCS. Alternatively, an implementation can choose to implement one of the pre-packaged modules as its Application Profiles, such as the work done by EO-WCS Application Profile. Whatever set of conformance classes a service implements is announced to the client in the response to a GetCapabilities request.

To support retrieval of a geospatial coverage, three basis operations are defined in WCS to probe information from both metadata and data levels. Specifically, these are:

- a) GetCapabilities retrieves an overview about server functions, including the available extension modules, and a list of available coverages.
- b) DescribeCoverage retrieves a comprehensive description of the addressed coverage, including its native format, which should be consistent with the available encoding extension modules as declared in the service capabilities document.
- c) GetCoverage retrieves the whole or part of an identified coverage, including both space/time and attribute information. In this way, WCS gives access to multi-dimensional coverage data, rather than rendering.

The WCS Core's mandatory functions are defined based on the GML Application Schema for Coverages. Service model extensions describe additional functionality which may be added to a WCS [OGC 09-153]. For example, Scaling and Interpolation extensions allow GetCoverage to scale coverage results; different interpolation techniques can be selected for the resampling performed during a scaling operation. CRS extension allow coverages to be requested in different CRSs; the server needs to be able to perform a CRS transform of the coverage prior to its delivery. This extension requires Scaling and Interpolation extensions. WCPS defines a query language which allows to combine and process coverages for navigation, extraction (download), aggregation, and ad-hoc analysis. However, this extension needs at least one encoding extension for its coverage response and one protocol extension, such as KVP, XML/POST, SOAP, WPS , for the communication. An investigation has been carried out in OGC 08-070 on how this binding can be done via the embedding approach.

Extensions and Application profiles add further request types, such as:

- d) ProcessCoverages (defined in the WCS Processing Extension) allows database-style processing and filtering on coverages through the multi-dimensional spatio-temporal raster query language WCPS (Web Coverage Processing Service).
- e) DescribeEOCoverageSet (defined in the EO-WCS Application Profile) allows spatial and temporal retrieval on 2D satellite imagery.

7 Reference Implementations: Rasdaman and EoxServer

WCS 2.0 series is known to be implemented by at least six Open Source software packages¹, namely: rasdaman, EoxServer, MapServer, OPeNDAP, GMU, and GeoTrellis. In this section, examples will be given based on the rasdaman Reference Implementation (RI) and EoxServer to demonstrate how core and extension modules helps to provide interoperable coverage access.

This section can safely be skipped if the reader's main interest is in the conformance testing aspects discussed in Section 8.

7.1 How GetCapabilities related with the modules

The Capabilities document returned contains a general service description and a summary of the coverages offered. As part of the service description, `ows:Profile` elements list the extension modules implemented, for example²:

¹ Plus there are further commercial packages supporting WCS 2, such as IGeS.

² demo: <http://www.earthlook.org>

```

<ows:Profile>
  http://www.opengis.net/spec/WCS_coverage-encoding/1.0/conf/application/gml+xml
</ows:Profile>
<ows:Profile>
  http://www.opengis.net/spec/WCS_protocol-binding_get-kvp/1.0
</ows:Profile>
<ows:Profile>
  http://www.opengis.net/spec/WCS_coverage-encoding_jpeg2000/1.0
</ows:Profile>
<ows:Profile>
  http://www.opengis.net/spec/WCS_protocol-binding_post-xml/1.0
</ows:Profile>
<ows:Profile>
  http://www.opengis.net/spec/WCS_service-extension_crs/1.0/conf/crs
</ows:Profile>
<ows:Profile>
  http://www.opengis.net/spec/WCS_protocol-binding_soap/1.0
</ows:Profile>
<ows:Profile>
  http://www.opengis.net/spec/WCS_coverage-encoding_geotiff/1.0
</ows:Profile>

```

Furthermore, the Capabilities document lists all coverage encoding formats, identified by their MIME type, in which the server on hand can return coverages, such as:

```

<wcs:formatSupported>application/gml+xml</wcs:formatSupported>
<wcs:formatSupported>image/tiff</wcs:formatSupported>
<wcs:formatSupported>image/jp2</wcs:formatSupported>

```

Only those formats listed in the `wcs:formatSupported` element can effectively be used by clients, independent from the format encoding extensions listed in the `ows:Profile` elements.

7.2 How DescribeCoverage related with the modules

A CoverageDescription consists of the metadata of the addressed coverage, including the coverage's Native Format, which takes effect when the format parameter is not present in a getCoverage request. The example is demonstrated as below:

```

<wcs:ServiceParameters>
  <wcs:CoverageSubtype>RectifiedGridCoverage</wcs:CoverageSubtype>
  <wcs:nativeFormat>image/tiff</wcs:nativeFormat>
</wcs:ServiceParameters>

```

Further extensions are supported via `wcs:Extension`, for example, EO-WCS extend coverage with the Earth Observation (EO) metadata³ via its `wcseo:EOMetadata`.

7.3 How `getCoverage` related with the modules

The `getCoverage` operation uses the return `Capabilities` and `CoverageDescription` documents to make its request. The supported functions are checked via the given `ows:Profile` list and the supported coverage encoding formats are checked via the given `wcs:formatSupported` list. Extra parameters are added to the core `getCoverage` operation to invoke these functions. For example, a sample XML/POST request with CRS support, is show as below:

```
<wcs:GetCoverage xmlns:wcs="http://www.opengis.net/wcs/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:rsub="http://www.opengis.net/wcs_service-extension_range-subsetting/1.0"
  service="WCS" version="2.0.1">
  <wcs:Extension>
    <crs:subsettingCrS>
      http://www.opengis.net/def/crs/EPSSG/0/4326
    </crs:subsettingCrS>
  </wcs:Extension>
</wcs:GetCoverage>
```

The extension is a wildcard, which is used to be replaced by future functions. These functions may be unknown or technically unavailable to us (happens a lot in the standardization process) currently. We need not have to inherit from it because it is not decidable to define a subtype by inheriting from a wildcard "any", which will create the case of "any" + "any". Corresponding constraints of the additional ingredients can be defined in schemas of the addressed extensions. The `ExtensionType` has been defined in the `wcsCommon.xsd` as a scalable solution to address the unknown. Extension combination is allowed too.

Obviously, an implementation may choose to implement more service functions, such as CRS, Scaling and Band Subsetting. A sample XML/POST request is show as below:

```
<wcs:GetCoverage xmlns:wcs="http://www.opengis.net/wcs/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:rsub="http://www.opengis.net/wcs_service-extension_range-subsetting/1.0"
  service="WCS" version="2.0.1">
  <wcs:Extension>
    <wscrs:subsettingCrS>
      http://www.opengis.net/def/crs/EPSSG/0/4326
    </wscrs:subsettingCrS>
  </wcs:Extension>
</wcs:GetCoverage>
```

³) demo: <http://ows.eox.at/cite/ows>

```

    </wscsrs:subsettingCrS>
    <wscsrs:outputCrS>
      http://www.opengis.net/def/crs/EPSSG/0/4326
    </wscsrs:outputCrS>
  </wcs:Extension>
  <wcs:Extension>
    <scal:ScaleByFactor>
      <scal:scaleFactor>2.0</scal:scaleFactor>
    </scal:ScaleByFactor>
  </wcs:Extension>
  <wcs:Extension>
    <rsub:rangeItem>
      <rsub:rangeComponent>band1</rsub:rangeComponent>
    </rsub:rangeItem>
    <rsub:rangeItem>
      <rsub:rangeInterval>
        <rsub:startComponent>band3</rsub:startComponent>
        <rsub:endComponent>band5</rsub:endComponent>
      </rsub:rangeInterval>
    </rsub:rangeItem>
  </wcs:Extension>
  <wcs:CoverageId>mean_summer_avgtemp</wcs:CoverageId>
  <wcs:format>application/gml+xml</ wcs:format >
  <wcs:format>multipart/related </ wcs:format >
</wcs:GetCoverage>

```

The encoding format in which the coverage will be returned is specified by the combination of `format` and `mediaType` parameter. Admissible values (i.e, formats supported) are those listed in the server's Capabilities document. Default is the coverage's Native Format [OGC 09-110r4].

7.4 Arbitrary coverage processing beyond `getCoverage`

The OGC Web Coverage Processing Service (WCPS) defines a language for retrieval and processing of multi-dimensional geospatial coverages representing sensor, image, or statistics data. The language is independent from any particular request and response encoding, as no concrete request/response protocol is specified by WCPS. For setting up a WCPS instance, therefore, a separate, additional specification establishing the concrete protocol is required. This allows embedding of WCPS into different target service frameworks. For example, OGC 08-070 describes the relation between WPS and WCPS, in particular, how WCPS embeds itself into WPS. A sample request in KVP is shown as below:

```

http://foo.bar/foo?
  request="Execute"&
  service="WPS"&
  version="0.3.0"&
  Identifier="ProcessCoverages"&

```

```
DataInput=  
  "for%20C%20in(mean_summer_avgtemp)%20  
  return%20  
  encode(C, %22TIFF%22)"
```

Obviously, more investigation is needed on how this powerful language can be binding with the existing protocols, such as XML/POST, SOAP and RESTFUL, and how its outputs can be coupled with format encoding extensions.

8 Conformance testing and enhancement

This section describes the WCS test suite updates performed, as well as new TEAM engine concepts proposed to address the needs of modular specification testing.

The WCS test suite is available from

<https://svn.openeospatial.org/ogc-projects/cite/scripts/wcs/2.0.1/>

The EO-WCS test suite is available from

<https://svn.openeospatial.org/ogc-projects/cite/scripts/wcseo/1.0/>

8.1 Test suite update

Test suites of WCS 2.0 series have been updated to address the recently released WCS 2.0 corrigendum, extensions (band subsetting, scaling, interpolation, CRS, format encodings, WCPS), and the EO Application Profile, EO-WCS. ATs have been added to extension specifications to complete them for roll-out, and corresponding ETs have been developed in OGC CTL.

Development of test suites follows a cross-testing approach. The reference implementations and the test suites, which are derived from the same standards suite, are developed in parallel and promote each other in conformance testing rounds. This effort has been complemented by work done in the DREAM project sponsored by the European Space Agency (ESA).

8.2 Capabilities Control Testing (CCT)

The core / extension paradigm leads to specifications which depend on each other in manifold and standard-specific ways.

This is a consequence of the fact that many independent capability facets may be provided via WCS extension implementations, such as band subsetting, scaling and interpolation. Moreover, coverages can be provided in a variety of different encodings.

Testing actually will be infinite if there are no specific constraints on what to test. Therefore, it is important to deduce the set of requirements which need to be tested as prerequisites before a particular testing starts.

Rather than hardcoding such dependencies, which would lead to a duplication of information in specification and test and, therefore, be quite error prone, it was chosen to perform dynamic capability sniffing: when testing a particular specification – say, some WCS extension – this test first checks possibly dependent prerequisites (such as protocol bindings) on the concrete service under test and runs the corresponding conformance tests prior to challenging the conformance class under inspection. The Core test is always run first.

Through the extraction of Capabilities document's `ows:Profile` elements, the tests to be invoked can be decided. For example, the below CTL shows how a KVP extension can be invoked if supported by the service under test:

```
<xsl:if test="contains(string-join(($Capabilities//*[local-name()='ServiceIdentification']/*[local-name()='Profile']), ' '), 'http://www.opengis.net/spec/WCS_protocol-binding_get-kvp/1.0')">
  <ctl:call-test name="wcs2:get-kvp-main">
    <ctl:with-param name="url" select="$url"/>
  </ctl:call-test>
</xsl:if>
```

This avoids enumerating the abundant space of service combinations which core and extensions establish. As an important consequence, this approach provides implementation flexibility to service vendors. For example, a service entry may only provide its WCS service via KVP and its coverage encoding in GML, and another entry may provide its WCS in SOAP and its coverage encoding in NetCDF.

8.3 Validation of diverse coverage encodings

WCS 2 is unique in that it establishes conformance testing down to pixel level, irrespective of the various formats in which a coverage can be communicated. To this end, CTL has to support validation of results which may come in some binary format.

A response in XML can be easily handled via CTL; for example, there is convenient support for schema validation, consistency verification between a coverage description reported and the GML coverage returned.

In OWS-9, CTL parser capabilities have been used to provide checking of coverages delivered in further encodings, such as PNG, TIFF, and JPEG. These capabilities can only be invoked if the test engine knows in advance what is going to be returned; in the WCS mechanics, this is ensured as the engine knows what formats are supported by the service under test and, based on this, requests the particular encoding itself.

The CTL snippet below shows the pattern for a proper coverage encoding validation:

```
<xsl:choose>
```

```

<xsl:when test="compare($format, 'application/gml+xml')=1">
  <ctlp:XMLValidatingParser>
    ...
  </ctlp:XMLValidatingParser>
</xsl:when>
<xsl:when test="compare($format, 'image/tiff')=1">
  <ctlp:ImageParser>
    ...
  </ctlp:ImageParser>
</xsl:when>
<xsl:otherwise>
  ...
</xsl:otherwise>
</xsl:choose>

```

8.4 Proposed TEAM enhancements

8.4.1 Address multi-dimensional coverage encodings

Coverages returned in some non-XML format are parsed via the CTL ImageParser. However, GIF, PNG, and JPEG are supported, which is by far insufficient for coverages which require formats such as GeoTIFF, JPEG2000, netCDF, and many more. OGC 06-126r5 Subclause 11.5.1 states that “The ImageParser extracts the body of a response message and treats it as a still image resource. If the entity does not correspond to a supported image format type like GIF, PNG or JPEG, an exception is reported in the test.” Further, this indicates already the CTL restriction to two-dimensional x/y coverages, whereas WCS can serve any-dimensional data.

Therefore, there is a need to support multi-dimensional coverage encodings in addition. GML coverages are readily testable with the current TEAM Engine. Actually, WCS format extensions – such as GeoTIFF, JPEG2000 and netCDF – specify their coverage encoding by mapping the GML coverage elements to the particular format facets. This allows for a convenient testing of these encodings by converting coverages received, in the TEAM engine, into GML format which subsequently is passed to the `XMLValidatingParser`. Hence, test support for a particular WCS format extension requires implementation of a corresponding conversion module. As generic entry point for such tests a `ConversionParser`, which accepts a particular WCS format and produces output in customized GML format, is proposed to address the GeoTIFF, JPEG2000 and netCDF conversion modules.

Establishing such a mechanism will be indispensable for WCS encoding format testing, so it is recommended as an OWS-10 task.

8.4.2 Core and extensions handling

There are manifold dependencies among specifications, which require dynamic orchestration of tests depending on the concrete capabilities of the service under test. For

example, WCS 2.0 Core requires that at least one protocol extension and one encoding extension need to be offered by the service. Consequently, Core testing can not bypass the protocol and encoding extensions offered.

However, the CTL code of a particular test can only be invoked in its individual repository, and by convention each extension test is sitting in its own directory. Therefore, these extensions CTL need to be copied into the core CTL's repository in order to test the dependencies.

Several possible solutions have been spotted. To avoid unnecessary duplication, all tests can be put together into one directory and invoke only the particular test subset as declared in the capabilities document. This contradicts current philosophy and is likely to lead to large, unwieldy test conglomerates. Another option is to keep tests separated, manually invoke them individually, and do the overall evaluation at the end. Clearly, manual interference is undesirable.

TestNG, which has already been applied in GML 3.2.1 and WFS 2.0 testing, is another proposal which incorporates the desired test classes into the test suite descriptor as a separate test group. For example, one or more test suites, each of which may include a collection of child test suites, may be run in a test execution. A requirement is that these test classes must be available in the classpath at TEAM invocation time, which poses no particular problem, as the test set is known at invocation time.

Establishing such a mechanism will be indispensable not only for WCS testing as such, but for any new OGC standard following the Core/Extension model. Hence, it is strongly recommended as an OWS-10 task.

8.4.3 Asynchronous web service

Currently, such a feature has not been explicitly supported by WCS 2.0, however, the key functions have been applied in WMS 1.3 client test suite, such as `allocate-monitor-url` and `create-monitor.examine-monitor` is yet to be implemented to address the incoming requests and output responses.

9 Conclusion

In OWS-9, maturation and completion of WCS has made significant progress. WCS functionality is close to complete with regard to planning, and conformance testing has reached the level where rigorous validation, down to pixel level, is possible for a series of specifications. Ultimately, OGC branding is possible now of several WCS specifications, including the Core. This confirms the position of the OGC Unified Coverage Model and WCS as today's most advanced, functionally complete, easy-to-use, and mature standards suite for multi-dimensional spatio-temporal coverage handling.

As more and more WCS format extensions are rapidly being established now, it is indispensable to establish corresponding ETSS. This is a work item highly recommended for OWS-10.

Annex A

Abstract Test Suites Accomplished

The ATS, as per OGC convention, is included in the Core and each extension's Annex A. Specifically, the following ATSS modified or established have been established or updated in the course of OWS-9:

- OGC 11-153. OGC® Web Coverage Service Interface Standard - CRS Extension
- OGC 08-068r2. Web Coverage Processing Service (WCPS) Language Interface Standard
- OGC 12-039. OGC® Web Coverage Service Interface Standard - Scaling Extension
- OGC 12-040. OGC® Web Coverage Service Interface Standard - Range Subsetting Extension
- OGC 12-049. OGC® Web Coverage Service Interface Standard - Interpolation Extension
- OGC 10-140. OGC® Web Coverage Service 2.0 Interface Standard - Earth Observation Application Profile

Annex B

Executable Test Suite

The WCS ETS is publicly accessible from:

<https://svn.opengeospatial.org/ogc-projects/cite/scripts/wcs/2.0.1/>

<https://svn.opengeospatial.org/ogc-projects/cite/scripts/wcseo/1.0/>