

Open Geospatial Consortium Inc.

Date: 2010-02-15

Reference number of this OpenGIS® Project Document: **OGC 09-147**

Version: 0.0.1

Category: OpenGIS® Interface Standard

Editor: Peter Baumann

WCS Extension -- KVP Protocol

Copyright © 2010 Open Geospatial Consortium, Inc.

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

Editorial note: revision numbers for the references to documents of the WCS 2.0 set are not yet adjusted; this will be done in the final editing step.

Warning

This document is not an OGC Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard.

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type:	OGC® Interface Standard
Document subtype:	
Document stage:	Draft
Document language:	English

Contents	Page
1 Scope	1
2 Compliance	1
3 Normative references	1
4 Terms and definitions	1
5 Conventions	2
5.1 UML notation	2
5.2 Data dictionary tables	2
6 HTTP/GET with KVP	2
6.1 General	2
6.2 GetCapabilities	2
6.3 DescribeCoverage	3
6.4 GetCoverage	3
7 Exceptions	4
Annex A (normative) Abstract test suite	6
A.1 Conformance Test Class: WCS KVP protocol	6
A.1.1 Extension identification	6
A.1.2 Encode special characters	6
A.1.3 Proper case handling	6
A.1.4 <i>GetCapabilities</i> request encoding	7
A.1.5 <i>GetCapabilities</i> response structure	7
A.1.6 <i>DescribeCoverage</i> request encoding	7
A.1.7 <i>GetCoverage</i> request encoding	7
A.1.8 <i>GetCoverage</i> subset specification	8
A.1.9 Disjoint <i>GetCoverage</i> subsetting parameter keys	8
A.1.10 Exceptions	9

Tables	Page
Table 1 — DescribeCoverage request URL encoding	3
Table 2 — <i>GetCoverage</i> request KVP encoding	3
Table 3 — Exception codes for XML/POST operations	5

i. Preface

This document specifies an extension to the OGC Web Coverage Service (WCS) 2.0 core to allow for client/server communication using HTTP GET with key/value pair (KVP) encoding.

ii. Terms and definitions

This document uses the specification terms defined in Subclause 5.3 of [OGC 06-121r3], which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

iii. Submitting organizations

The following organizations have submitted this Implementation Specification to the Open GeoSpatial Consortium, Inc.:

- Jacobs University Bremen
- rasdaman GmbH
- National Center for Atmospheric Research (NCAR)
- Oracle USA
- PCI Geomatics Inc.
- ERDAS, Inc.
- EOX IT Services GmbH
- Spot Image
- BAE Systems - C3I Systems
- Natural Environment Research Council (NERC)
- George Mason University

iv. Document Contributor Contact Points

Name	Organization
Peter Baumann	Jacobs University Bremen, rasdaman GmbH

v. Revision history

Date	Release	Author	Paragraph modified	Description
2009-08-22	0.0.1	PB	All	Created

vi. Changes to the OpenGIS[®] Abstract Specification

The OpenGIS[®] Abstract Specification does not require any changes to accommodate the technical contents of this (part of this) document.

vii. Future Work

Nothing foreseen currently.

Foreword

Some of the elements of this document may be the subject of patent rights. Open GeoSpatial Consortium Inc. shall not be held responsible for identifying any such patent rights.

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 Inc. 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.

Introduction

The OGC Web Coverage Service (WCS) supports electronic retrieval of geospatial data as "coverages" – that is, digital geospatial information representing space/time-varying phenomena.

This document specifies an extension to the OGC Web Coverage Service (WCS) 2.0 core to allow for client/server communication using HTTP GET with key/value pair (KVP) encoding.

WCS Extension -- KVP Protocol

1 Scope

This document specifies how Web Coverage Service (WCS) clients and servers can communicate over the Internet using HTTP GET with key/value pair (KVP) encoding.

2 Compliance

Annex A (normative) specifies compliance tests which shall be tested by any service claiming to implement an OGC Web Coverage Service using this extension.

3 Normative references

The following normative documents contain provisions that, through reference in this text, constitute provisions of this specification. 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.

IETF RFC 2387, *The MIME Multipart/Related Content-type*. IETF, 1998

IETF RFC 2396, *Uniform Resource Identifiers (URI): Generic Syntax*. IETF, 1998

W3C Note 11, *SOAP Messages with Attachments*. W3C Note 11, 2000

OGC 06-121r8, *OGC Web Services Common Specification*, version 1.2

OGC 07-036, *Geography Markup Language (GML) Encoding Standard*, version 3.2.1

OGC 09-110r1, *WCS 2.0 Core*, version 2.0

In addition to this document, the WCS specification includes normative XML Schema files. These are posted online at <http://schemas.opengis.net/wcs/2.0> as part of the OGC schema repository. These XML Schema files are also available bundled with the present document for download from **Error! Hyperlink reference not valid.** . In the event of a discrepancy between bundled and schema repository versions of the XML Schema files, the schema repository shall be considered authoritative.

4 Terms and definitions

For the purposes of this document, the terms and definitions given in the above references apply. In addition, the following terms and definitions apply.

4.1

coverage

feature which is a subclass (specialization) of `AbstractCoverage` as specified by GML 3.2 [07-036], based on the abstract definition of [OGC 07-111] and [ISO 19123]

4.2

offered coverage

→*coverage* stored on a server and accessible by clients via WCS operations

NOTE An offered coverage carries service relevant information in addition to the →*coverage* data.

5 Conventions

5.1 UML notation

All the diagrams that appear in this specification are presented using the Unified Modeling Language (UML) static structure diagram, as described in Subclause 5.2 of OGC Web Services Common [OGC 06-121r3].

5.2 Data dictionary tables

The UML model data dictionary is specified herein in a series of tables. The contents of the columns in these tables are described in Subclause 5.5 of [OGC 06-121r3]. The contents of these data dictionary tables are normative, including any table footnotes.

6 HTTP/GET with KVP

6.1 General

Req 1 A WCS service implementing this extension shall include the following URN in the Profile element of the ServiceDescription in a *GetCapabilities* response:
urn:ogc:def:extension:OGC-WCS:2.0:protocol:get

Req 2 Operation responses **shall** or URL-encode special characters as defined in [RFC 2396].

Example Use “%3F” to represent a question mark, “?”

Req 3 Keys **shall** be case insensitive, values **shall** be case sensitive.

6.2 GetCapabilities

Req 4 The KVP encoding of a WCS *GetCapabilities* operation request **shall** be as specified in Table 5 in Subclause 7.2.2 of [OGC 06-121r8], with the exception that values for the REQUEST and SERVICE parameter **shall** be interpreted in a case insensitive manner.

Example To request a Capabilities document, a client can issue the following minimal *GetCapabilities* operation request encoded as KVP:

`http://hostname:port/path?service=WCS&request=GetCapabilities`

Req 5 The response to a successful *GetCapabilities* request **shall** be a valid XML document of type `wcs:CapabilitiesType`.

Example See [OGC 09-110].

6.3 DescribeCoverage

Req 6 The KVP encoding of a *DescribeCoverage* request **shall** be as defined in Table 1. Values for the REQUEST and SERVICE parameter **shall** be interpreted in a case insensitive manner.

Table 1 — DescribeCoverage request URL encoding

Name	Definition	Data type	Multiplicity
Service	Identifier of the OGC service	String	One (mandatory)
Version	Request protocol version	String	One (mandatory)
Request	Request type name	String	One (mandatory)
Id	List of coverage identifiers to be described	Comma-separated gml:id list, possibly empty	One (mandatory)

Example The following KVP structure requests information about the coverages with identifiers 42, 43, and 44, resp.:

```
http://www.myserver.org:port/path?
  service=WCS
  &version=2.0
  &request=DescribeCoverage
  &idlist=42,43,44
```

6.4 GetCoverage

Req 7 The KVP encoding of a *GetCoverage* request **shall** be as defined in Table 2. Values for the REQUEST and SERVICE parameter **shall** be interpreted in a case insensitive manner.

Table 2 — GetCoverage request KVP encoding

Name	Definition	Data type	Multiplicity
service	Identifier of the OGC service	String, fixed to “WCS”	One (mandatory)
version	Request protocol version	String	One (mandatory)
request	Request type name	String, fixed to “GetCoverage”	One (mandatory)
id	Identifier of coverage to be inspected	gml:id	One (mandatory)
subsetd	boundaries of coverage subset where <i>d</i> is some non-empty string which serves as disambiguator, ie, makes subset keys unique within the request on hand	SubsetSpec	Zero or more (optional)

Req 8 Each SubsetSpec **shall** adhere to this EBNF syntax based on [IETF 2616]:

```
SubsetSpec : dimension [ _ crs ] ( intervalOrPoint )
dimension : NCname
```

```

crs :                anyURI
intervalOrPoint: interval | point
interval :           low ⏟ high
low :                number | *
high :               number | *
point :              number | ⏟

```

A *crs* element **shall** occur only if a WCS extension enabling CRSs is included in the implementation.

Note Syntax rules are as follows: underlined tokens represent literals which appear “as is” (“terminal symbols”), other tokens represent sub-expressions to be substituted (“non-terminals”). A vertical bar (“|”) denotes alternatives, items in brackets (“[]”) are optional. The non-terminals *NCname*, *number*, and *anyURI* follow the resp. XML definitions.

Req 9 All subset key disambiguators **shall** be pairwise disjoint in any given *GetCoverage* request:

Note There is no semantics associated with these key disambiguators, their only purpose is to achieve unique keys, which is required by HTTP GET.

Example The following KVP-encoded *GetCoverage* request addresses service *path* on server *www.myservice.org* at port *port* retrieves all range fields of coverage *42* in the coverage’s native array index coordinates:

```

http://www.myserver.org:port/path?
  service=WCS
  &version=2.0
  &request=GetCoverage
  &id=42
  &subset1=x(10,20)
  &subset2=y(*,20)
  &subset_TIME=t(0)

```

Example The following KVP-encoded *GetCoverage* request addresses the same service as above, but with the location specified by the bounding box with lower corner (-71,47) and upper corner (-66,51), expressed in spatial CRS WGS84 2D (which is assumed to be in the coverage’s CRS list) and temporal CRS ISO:8601:

```

http://www.myserver.org:port/path?
  service=WCS
  &version=2.0
  &request=GetCoverage
  &id=42
  &subset_wgs84x=x,urn:ogc:def:crs:OGC:2:84(-71,47)
  &subset_wgs84y=y,urn:ogc:def:crs:OGC:2:84(-66,51)
  &subset_time=t,urn:ogc:def:crs:OGC:2:84(2009-11-06T23:20:52Z)

```

7 Exceptions

Req 10 In addition to the exception codes defined in the WCS core [09-110], the code(s) in Table 3 shall be supported.

Table 3 — Exception codes for XML/POST operations

exceptionCode value	Meaning of code	“locator” value
InvalidEncodingSyntax	Document received does not conform with protocol syntax.	key of violating element

Annex A (normative)

Abstract test suite

A WCS implementing this extension shall pass all of the following tests, plus those of the WCS core [09-110], to be conformant with this specification.

A.1 Conformance Test Class: WCS KVP protocol

A.1.1 Extension identification

Test Purpose: A WCS service implementing this extension shall include the following URN in the *Profile* element of the *ServiceDescription* in a *GetCapabilities* response:
 urn:ogc:def:extension:OGC-WCS:2.0:protocol:get

Reference: Req 1

Test method: Send a *GetCapabilities* request to the server under test, verify that the response contains a *Profile* element with said URN.

A.1.2 Encode special characters

Test Purpose: Operation responses **shall** URL-encode special characters as defined in [RFC 2396].

Reference: Req 2

Test method: For each request type, send a request which contains special characters and send a request such that the response contains special characters. Check correct handling of the special characters.

NOTE Client testing of this property is not currently addressed in this Abstract Test Suite.

A.1.3 Proper case handling

Test Purpose: Keys **shall** be case insensitive, values **shall** be case sensitive.

Reference: Req 3

Test method: For each request type:

- send requests to the server under test containing lower, mixed, and upper case keys. Check proper response.
- Send requests to the server under test with different case in values. Check that the server differentiates in its response.

A.1.4 *GetCapabilities* request encoding

Test Purpose: The KVP encoding of a WCS *GetCapabilities* operation request **shall** be as specified in Table 5 in Subclause 7.2.2 of [OGC 06-121r8], with the exception that values for the REQUEST and SERVICE parameter **shall** be interpreted in a case insensitive manner.

Reference: Req 4

Test method: Send *GetCapabilities* requests testing server response on the cases distinguished in said reference. Check proper response.

NOTE Client testing of this property is not currently addressed in this Abstract Test Suite.

A.1.5 *GetCapabilities* response structure

Test Purpose: The response to a successful *GetCapabilities* request **shall** be a valid XML document of type `wcs:CapabilitiesType`.

Reference: Req 5

Test method: Send a valid *GetCapabilities* request. Pass test if an XML validator reports validity of the response document against its schema definition.

A.1.6 *DescribeCoverage* request encoding

Test Purpose: The KVP encoding of a *DescribeCoverage* request **shall** be as defined in Table 1. , Values for the REQUEST and SERVICE parameter **shall** be interpreted in a case insensitive manner.

Reference: Req 6

Test method: Send *DescribeCoverage* requests testing server response on the cases distinguished in said reference. Check proper response.

NOTE Client testing of this property is not currently addressed in this Abstract Test Suite.

A.1.7 *GetCoverage* request encoding

Test Purpose: The KVP encoding of a *GetCoverage* request **shall** be as defined in Table 2. Values for the REQUEST and SERVICE parameter **shall** be interpreted in a case insensitive manner.

Reference: Req 7

Test method: Send *GetCoverage* requests testing server response on the cases distin-

guished in said reference. Check proper response.

NOTE Client testing of this property is not currently addressed in this Abstract Test Suite.

A.1.8 GetCoverage subset specification

Test Purpose: Each SubsetSpec **shall** adhere to this EBNF syntax based on [IETF 2616]:

```
SubsetSpec :    dimension [   crs ] (   intervalOr-
Point )
dimension :    NCname
crs :          anyURI
intervalOrPoint: interval | point
interval :     low   high
low :          number | *
high :         number | *
point :        number | -
```

Reference: Req 8

Test method: Send *GetCoverage* requests to the service under test, evaluate whether responses are adequate (based on knowledge about a pre-existing coverages with at least 3 dimensions). Exercise tests for the following situations:

- No subsetting parameter
- Zero/one/two trimmings, no slicing
- Zero/one/two slicings, no trimming
- trim operations with trim coordinates and with “*” for low and high bound (independently)

Each test in the above set shall be performed

- Once for each dimensionality supported by the server
- Without CRS parameter provided in the request, and with a valid CRS parameter provided (if the server supports CRS handling).

Pass if coverage responses indicate (by range set inspection) that the operation has been recognized and executed properly.

A.1.9 Disjoint *GetCoverage* subsetting parameter keys

Test Purpose: All subset key disambiguators **shall** be pairwise disjoint in any given *GetCoverage* request:

Reference: Req 9

Test method: NOTE This property describes client behavior, something currently not addressed

in this Abstract Test Suite.

A.1.10 Exceptions

Test Purpose: In addition to the exception codes defined in the WCS core [09-110], the code(s) in Table 3 shall be supported.

Reference: Req 10

Test method: Send requests of all types supported to the server under test. Each request shall include all (mandatory and) optional parameters and shall be valid except and for one parameter which shall contain an encoding error.

-- end of ATS --