

Open Geospatial Consortium Inc.

Date: 2005-04-22

Reference number of this OGC™ project document: **OGC 04-049r1**

Version: 0.1.0

Category: OGC™ Discussion Paper

Editor: Philippe Duschene & Jerome Sonnet

WCS Change Request: Support for WSDL & SOAP

Copyright notice

Copyright © Open Geospatial Consortium (2005)

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™ Publicly Available Discussion Paper
Document subtype: if applicable
Document stage: Draft
Document language: English

Contents

1	Background	1
2	References.....	Error! Bookmark not defined.
3	Proposal	Error! Bookmark not defined.

1 Background

The OpenGIS has been a precursor in Web Services matter, nevertheless, the pattern that has been used is not recognized by the industry as a standard “XML Web Services”. The work done during the the OpenGIS Web Service 2 initiative has provided the OpenGIS with interfaces that use the XML-related technologies supported by the industry, as SOAP for the communication protocol, WSDL for the interface description language, and UDDI for registering and searching services.

This change proposal present the required change to the WCS specification to interoperate with the industry standards.

2 References

[1] *Web Coverage Service (WCS), Version 1.0.0 [03-065r6]*, John D. Evans, 2003-08.

3 Proposal

Change Request 1: Define fixed parameter names for the Range subsetting names and values parameters.

3.1.1 Affected section(s), table(s), figure(s):

Subclause 9.2.2.1, 9.2.2.9, getCoverage.xsd.

3.1.2 Purpose of the proposed change:

Change the KVP encoding to ensure that the parameter names used in request are fixed in the interface definition and are not subject to the content provided by the WCS anymore.

3.1.3 Reason for change:

WSDL does not support the definition of dynamic parameter names, all parameters have to be defined at interface description stage.

3.1.4 Specific Suggested Changes:

The WCS 1.0 GetCoverage operation requires the use of dynamic parameter names for the range subsetting (e.g. BAND=...) making exhaustive definition of the parameter names impossible in interface description languages (and a fortiori in WSDL).

The intent is to change the GetCoverage request parameters definition to use fixed parameter names for the Range subsetting names and values.

For example :

&GroundTemperature=0/10&AirTemperature=-5/0,5/10

becomes

AXISNAMES=GroundTemperature,AirTemperature&AXISVALUES=0/10,(-5/0,5/10)

AXISNAMES parameter values are comma separated.

AXISVALUES parameter values are comma separated, complex values are enclosed in bracket.

3.1.4.1 Update Table 9

Remove

PARAMETER= val1,val2,...& or PARAMETER= min/max/res	(Included only for range sets with compound values) Request a range subset defined by constraining parameter PARAMETER. The PARAMETER key is a variable string; it must match the name of a parameter listed in the range set description of the selected coverage. For instance: band=1,5,3 (e.g., radiance values in bands 1, 5, 3) age=0/18 (e.g., counts of people with ages under 18 yrs.)
---	--

	Optional if the chosen range component has default values for the parameter.
--	--

And replace by

AXISNAMES=name1,name2,...	<p>A list of string which must match the name of a parameter listed in the range set description of the selected coverage.</p> <p>For instance:</p> <p>AXISNAMES= GroundTemperature, AirTemperature</p>
AXISVALUES=value1,value2,...	<p>A list of values (simple or complex) that constraint the range component that has been defined using the AXISNAMES parameter.</p> <p>For instance:</p> <p>AXISVALUES=0/10,(-5/0,5/10)</p> <p>Note that both AXISNAMES and AXISVALUES must have the same number of items.</p>

3.1.4.2 Update section 9.2.2.9

Replace section 9.2.2.9 by

If the range set of the selected coverage consists of compound values, GetCoverage requests may include constraints defined on the parameter(s) of the compound range set.

Such constraints are expressed using tow parameters, AXISNAMES and AXISVALUES, where

- a) Each item of the list of string in AXISNAMES matches the name of an AxisDescription element defined on that range set, and
- b) Each corresponding value in AXISVALUES is one of the acceptable values defined in the corresponding AxisDescription element.

Using constraint is optional if the selected range set has a default value on the corresponding AxisDescription.

3.1.5 Consequences of the change

This change will allow description of the WCS interface using interface description language and in particular WSDL.

3.1.6 Consequences if not approved

Without this modification it would be impossible to use WSDL to describe the WCS interface.

Change Request 2: Add a normative WSDL description as a Annex to the WCS specification.

3.2.1 Affected section(s), table(s), figure(s):

Annex shall include WSDL definition and schema folder shall include wcs_abstract.wsdl and wcs.wsdl, as well as the common schemas as provided by OWS 2 Common Architecture work.

3.2.2 Purpose of the proposed change:

Add WSDL description as a normative part to the specification.

3.2.3 Reason for change:

To provide the normative definition of the WSDL description of the WCS interface to be used each time a WCS is describe using this language.

3.2.4 Specific Suggested Changes:

Add a WSDL description of WCS as an Annex to the document and provide the electronic description integrated with the existing XSD description. OWS 2 Common Architecture has published such integrated schemas and WSDL documents.

3.2.5 Consequences of the change

WCS implementation shall use this WSDL definition to describe their WCS instances ensuring a better interoperability of the WCS interface while using the commonly used WSDL technology.

3.2.6 Consequences if not approved

Different WCS provide might define different way to describe WCS interface using WSDL reducing the interoperability of the interface.

Change Request 3: *Add a clear normative clause that state WSDL definition shall be used by all service providers that make a WSDL description available.*

3.3.1 Affected section(s), table(s), figure(s):

???

3.3.2 Purpose of the proposed change:

Add a normative clause that explain the intended use of the WSDL.

3.3.3 Reason for change:

It is important to explain the intended use of the normative WSDL description to avoid misused.

3.3.4 Specific Suggested Changes:

Add a normative clause saying:

“All WCS service described using a WSDL document shall use the standard WSDL as described in Annex X. It is not required to import the document provided by the OpenGIS but it is required to use exactly the same definition as in the provided WSDL. ”

Note: This part should probably be part of OWS Common specification

3.3.5 Consequences of the change

WCS instances describe using WSDL will use the standard WSDL, improving interoperability.

3.3.6 Consequences if not approved

Different WCS providers might define different way to describe WCS interface using WSDL reducing the interoperability of the interface.

Change Request 4: Include an optional link from the Capabilities to the WSDL.

3.4.1 Affected section(s), table(s), figure(s):

N/A

3.4.2 Purpose of the proposed change:

This optional link will ease discovery and use of WSDL description.

3.4.3 Reason for change:

The access to a WSDL description may facilitate the binding phase of Web Service access.

3.4.4 Summary of change:

Add an element in the Capabilities document that refers to the WSDL.

3.4.5 Consequences of the change

A link to the WSDL description will be available in the Capabilities document.

3.4.6 Consequences if not approved

There will be no way the advertise in the Capabilities that a service is described by a WSDL document.

Change Request 5: Include the description of the SOAP binding.

3.4.1 Affected section(s), table(s), figure(s):

Adding SOAP support require creation of a new section. Proposal is to add a section 6.3.5 SOAP that is presented here under.

6.3.5 SOAP

A Web Map Service may support the “SOAP” protocol.

The implementation of the SOAP protocol is using document/literal encoding and the messages are the one defined in the Annex E of this document.

There is two step in supporting SOAP,

- Adding definition of the SOAP binding to the WSDL description of the service
- Adding support for SOAP document/literal envelope

6.3.5.1 WSDL description of the SOAP binding

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
  targetNamespace="http://www.opengis.net/wms/soap"
  xmlns:wms-
  soap="http://www.opengis.net/wms/soap"
  xmlns:wms-
  req="http://www.opengis.net/wms/requests"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
```

```

<wsdl:documentation>
</wsdl:documentation>

<!-- import WMS interface definitions -->
<wsdl:import
  namespace="http://www.opengis.net/wms/requests"
  location="./wms-xml-interfaces.wsdl"/>

<!-- *****
                        Bindings
***** -->

<wsdl:binding name="WMS_SOAP_Binding"
  type="wms-req:WMS_XML_Port">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="GetCapabilities">
    <soap:operation/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
    <wsdl:fault name="exception">
      <soap:fault name="exception"
        use="literal"/>
    </wsdl:fault>
  </wsdl:operation>
  <wsdl:operation name="GetMap">
    <soap:operation/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <mime:content type="image/*"/>
    </wsdl:output>
    <wsdl:fault name="exception">
      <soap:fault name="exception"
        use="literal"/>
    </wsdl:fault>
  </wsdl:operation>
</wsdl:binding>
</wsdl:definitions>

```

6.3.5.2 Declaring SOAP support in Capabilities

SOAP support shall be declared in the Capabilities of the services. This will allow usage of SOAP independently of WSDL.

If SOAP is supported, the services shall advertise

<DCPType>

<HTTP>

.....

**<SOAP
onlineResource="http://webservices.ionicssoft.com/ionicweb/wfs/BOSTON_SHAPE?"/>**

</HTTP>

</DCPType>

6.3.5.3 SOAP document/literal envelope

A Web Map Service that support SOAP binding shall accept document/literal SOAP messages which contains a message as defined in the XML encoding for Web Map Service. It is not required for the Web Map Service to support any other feature or extension of the SOAP protocol.

```
<?xml version="1.0" encoding="UTF-8"?>
<env:Envelope
xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <Operation>
```

```

        ...
    </Operation>
</env:Body>
</env:Envelope>

```

If the result of the operation is an XML document, it shall be included in a SOAP envelope as well,

```

<?xml version="1.0" encoding="UTF-8"?>
<env:Envelope
  xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <Result>
      ...
    </Result>
  </env:Body>
</env:Envelope>

```

If the result is a Binary, the content of the Body is not specified and the binary is attached to the SOAP message using SOAP with attachment [SOAP-Attachment]. Implementation may use the Body as a container for metadata about the result.

References :

[SOAP-Attachment] <http://www.w3.org/TR/2004/NOTE-soap12-af-20040608/>

Adding an informative Annex with some example is important too,

```

<?xml version='1.0' encoding='utf-8' ?>
<env:Envelope
  xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <wcs:GetCapabilities
      xmlns:wcs="http://www.opengis.net/wcs"
      service="WCS" version="1.0.20" />
  </env:Body>
</env:Envelope>

```

GetCapabilities Request

```
<?xml version='1.0' encoding='utf-8' ?>
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <wcs:Capabilities version="1.0.20" ...
  </soap:Body>
</soap:Envelope>
```

GetCapabilities Response

```
<?xml version='1.0' encoding='utf-8' ?>
<env:Envelope
  xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <wcs:DescribeCoverage
      xmlns:wcs="http://www.opengis.net/wcs"
      service="WCS" version="1.0.20" />
  </env:Body>
</env:Envelope>
```

DescribeCoverage Request

```
<?xml version='1.0' encoding='utf-8' ?>
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <CoverageDescription
      xmlns:ows="http://www.opengis.net/ows"
      xmlns:wcs="http://www.opengis.net/wcs"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.opengis.net/wcs
http://www.ionicsoft.com:8080/wcs/coverage/SPOT5J/REQUEST/get/DATA/LPR/wcs/1.0.20/wcsDescribeCoverage.xsd"
      xmlns="http://www.opengis.net/wcs"
      xmlns:xlink="http://www.w3.org/1999/xlink"
      xmlns:gml="http://www.opengis.net/gml">
      <wcs:CoverageOffering>
        <wcs:description>
          Coverage
        </wcs:description>
        <wcs:name>TIF</wcs:name>
      </wcs:CoverageOffering>
    </CoverageDescription>
  </soap:Body>
```

```
</soap:Envelope>
```

DescribeCoverage Response

```
<?xml version='1.0' encoding='utf-8' ?>
<env :Envelope
  xmlns :env= »http ://schemas.xmlsoap.org/soap/env
  elope/ »>
  <env:Body>
    <wcs:GetCoverage
      xmlns:wcs="http://www.opengis.net/wcs"
      xmlns:gml="http://www.opengis.net/gml"
      xmlns:ows="http://www.opengis.net/ows"
      service="WCS" version="1.0.20">

      <wcs:sourceCoverage>MOD_Grid_L2g_2d</wcs:sourceCo
      verage>
      <wcs:domainSubset>
        <wcs:spatialSubset>
          <ows:WGS84BoundingBox
            xmlns:ows="http://www.opengis.net/ows"
            xmlns="http://www.opengis.net/ows"
            dimensions="2">
            <LowerCorner>-119.99999997
            39.999985</LowerCorner>
            <UpperCorner>-62.228953063
            59.999999995</UpperCorner>
            </ows:WGS84BoundingBox>
            <gml:Grid dimension="2">
              <gml:limits>
                <gml:GridEnvelope>
                  <gml:low>0 0</gml:low>
                  <gml:high>500 500</gml:high>
                </gml:GridEnvelope>
              </gml:limits>
              <gml:axisName>X</gml:axisName>
              <gml:axisName>Y</gml:axisName>
            </gml:Grid>
          </wcs:spatialSubset>
          <wcs:temporalSubset>
            <gml:timePosition>2003-12-
            07T20:50:00Z</gml:timePosition>
          </wcs:temporalSubset>
        </wcs:domainSubset>
        <wcs:rangeSubset>
          <wcs:measureSubset measureName="band1"/>
        </wcs:rangeSubset>
        <ows:interpolationMethod>nearest
```

```
neighbor</ows:interpolationMethod>
  <wcs:output crs="urn:opengis:def:crs:OGC:2:84"
    format="application/GeoTIFF" store="TRUE"/>
</wcs:GetCoverage>
</env:Body>
</env:Envelope>
```

GetCoverage Request

3.4.2 Purpose of the proposed change:

This change aims to add support for SOAP document/literal binding.

3.4.3 Reason for change:

The W3C standard way of doing web services is WSDL/SOAP/UDDI. Adding SOAP support will increase the potential user base of this specification.

3.4.4 Summary of change:

This change add support for SOAP.

3.4.5 Consequences of the change

The WCS interface will support SOAP binding.

3.4.6 Consequences if not approved

The industry standard for web services is WSDL/SOAP/UDDI, leaving the WCS without support for these is a thread for its mid-term adoption.