Open Geospatial Consortium Inc.

Date: 2009-03-25

Reference number of this OGC® Project Document: **OGC 08-059r3**

Version: 1.0.0

Category: OpenGIS® Interface Standard

Editor: Peter Baumann

Web Coverage Service (WCS) — ProcessCoverages Extension

Document type:	OpenGIS® Interface Standard
Document subtype:	Extension Package
Document stage:	Approved
Document language:	English

Warning

OGC official documents use a triple decimal-dot notation (i.e. MM.xx.ss). This document may be identified as MM.xx (Major.minor) and may include increments to the third dot series (schema changes) without any modification to this document, or the version displayed on the document. This means, for example, that a document labelled with versions 1.1.0 and 1.1.1 or even 1.1.9 are exactly the same except for modifications to the official schemas that are maintained and perpetually located at: http://schemas.opengis.net/. Note that corrections to the document are registered via corrigendums. A corrigendum will change the base document and notice will be given by appending a c# to the version (where # specifies the corrigendum number). In corrigendums that correct both the schemas and the base document, the third triplet of the document version will increment and the 'c1' or subsequent identifier will be appended, however the schemas will only increase the third triplet of the version.

This document is 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.

Contents

Figures

Page

Tables

Page

i. Preface

The OGC Web Coverage Service Processing Extension is an extension of the Web Coverage Service (WCS) Standard [OGC 07-067r5], based on the Web Coverage Processing Service (WCPS) Language Interface Standard [OGC 08-068r2]. This version of the extension applies to WCS version 1.1.2; with small changes, however, it is expected to also apply to the next version of WCS.

Suggested additions, changes, and comments on this draft specification are welcome and encouraged. Such suggestions may be submitted by email message or by making suggested changes in an edited copy of this document.

ii. Submitting organizations

The following organizations have submitted this Interface Specification to the Open Geospatial Consortium, Inc.

- Jacobs University Bremen

iii. Document Contributor Contact Points

Name	Organization
Peter Baumann	Jacobs University Bremen, rasdaman GmbH

iv. Revision history

Date	Release	Author	Paragraph modified	Description
2005-06-06	0.0.1	Peter Baumann, Trimita Chakma		Initial draft
2005-12-15	0.0.2	Peter Baumann, Georgi Chulkov		Reworked based on WCS progress
2006-11-24	0.0.3	Peter Baumann		Editorial changes and corrections Definition of a name EBNF syntax adapted to WCS/IETF Added DescribeCoverage request Adapted to WCS 1.1 (06-083r8), in particular: its coverage model Generalized domain model
2006-12-14	0.0.4	Peter Baumann	4 Annex A	Revised dimension and domain definition Syntactical error corrections, slight syntax modifications for trim/extend/slice/scaleExpr,
2007-04-17	0.0.4_b	Peter Baumann	Header; slice; dimensionSet()	Revised slice op Brushed up for 06035r2
2007-11-20	0.0.6	Peter Baumann	Several	More examples, removed some syntactic inconsistencies

2008-03-25	0.0.7	Peter Baumann	several	Syntax fix in point/interval list; added char to Table 7; dropped "in" in coverage constructor; typo in coverage constructor; typo in scale() syntax; updated "Conformance" Annex organised as extension to WCS 1.1.2
2008-04-29	0.0.8	Peter Baumann	Several	Editorial changes for extension; completed UML; factored out WCPS language into separate specification document
2008-06-07	0.0.9	Peter Baumann	Several	Incorporated change requests by Arliss Whiteside
2009-01-10	1.0.0	Peter Baumann	editorial	Finalized for release as standard

v. 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.

vi. Future Work

This framework is planned to be enhanced and extended, among others, with the following features:

1) Consider further protocols, such as REST.

Foreword

This document is an extension of the Web Coverage Service (WCS) Standard [OGC 07-067r5], based on the Web Coverage Processing Service (WCPS) Language Interface Standard [OGC 08-068r2]. This version of the extension applies to WCS version 1.1.2; with small changes, however, it is expected to also apply to the next version of WCS.

Unlike previous versions, WCS 1.1 and 1.2 are divided into a basic standard plus multiple extensions. This extension is based on a Best Practice Paper in the versions [OGC 07-157r1] [OGC 08-059], and supersedes those documents. This document does not supersede any other previously approved OGC document.

This document includes three normative Annexes A, B, and C.

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 Web Coverage Service (WCS) [OGC 07-067r5] supports electronic retrieval of geo-spatial data as "coverages" – that is, digital geospatial information representing space-varying phenomena. A WCS provides client access to potentially detailed and rich sets of geospatial information, in forms that are useful for client-side rendering, multi-valued coverages, and input into scientific models and other clients. The WCS is currently limited to quadrilateral grid coverages, providing information at the grid points, usually with interpolation between these grid points.

This extension of the WCS standard specifies the service interface encoding to an additional Web Coverage Processing Service (WCPS), the ProcessCoverages operation, that may optionally be implemented by WCS servers. This extension allows retrieval and processing of geospatial coverage data based on the coverage model of the OGC Web Coverage Service (WCS) Standard [OGC 07-067r5].

WCPS provides access to original or derived sets of geospatial coverage information, in forms that are useful for client-side rendering, input into scientific models, and other client applications. As such, WCPS includes WCS functionality and extends it with an expression language to form requests of arbitrary complexity allowing, e.g., multi-valued coverage results.

To this end, this extension defines an additional request type *ProcessCoverages* beyond the request types *GetCapabilities*, *DescribeCoverage*, and *GetCoverage*, which are mandatorily required by WCS [OGC 07-067r5].

NOTE The expressive power of the GetCoverage operation is a proper subset of the *Process-Coverages* expressiveness. Hence, any GetCoverage request can be rephrased as a *ProcessCoverages* request.

Familiarity is assumed with the Web Coverage Service (WCS) standard [OGC 07-067r5] on whose concepts and terminology WCPS builds.

Open GeoSpatial Consortium Interface:
Web Coverage Service (WCS) — ProcessCoverages extension

1   Scope

This extension of the WCS standard specifies an additional processing operation that may optionally be implemented by WCS servers. This operation, the ProcessCoverages request type, allows a client to request processing of multi-dimensional grid coverage data on a WCS server by means of the Web Coverage Processing Service (WCPS) language and to retrieve the results over the World Wide Web. Result coverages can be transmitted directly or made available for download by URLs communicated to the client.

Example   The following WCPS expression retrieves the difference between red and green channels of coverages Modis1, Modis2, and Modis3, encoded in NetCDF (the format name may vary, depending on the name specified in the NetCDF format encoding extension specification):

```
for $c in ( Modis1, Modis2, Modis3 )
return
    encode( abs( $c.red – $c.green ), "netcdf" )
```

The syntax used is the one defined in [OGC 08-068r2]. For sending it as a *ProcessCoverages* request to a WCS server this expression must be encoded as specified in Section 7.1.

The Web Coverage Service (WCS) supports electronic retrieval of geospatial data as "coverages" – that is, digital geospatial information representing space-varying phenomena. A WCS provides client access to potentially detailed and rich sets of geospatial information, in forms that are useful for client-side rendering, multi-valued coverages, and input into scientific models and other clients. The WCS is currently limited to quadrilateral grid coverages, providing information at the grid points, usually with interpolation between these grid points.

This document is an extension of the Web Coverage Service (WCS) 1.1 Corrigendum 2 (version 1.1.2) Interface Standard [OGC 07-067r5] and based on the Web Coverage Processing Service (WCPS) 1.0 Interface Standard [OGC 08-068r2].

Familiarity is assumed with the Web Coverage Service (WCS) standard [OGC 07-067r5] and WCPS [08-068r2] on whose concepts and terminology this extension builds.

2   Compliance

Compliance with this extension Standard **shall** be checked for WCS servers claiming to implement this Standard, using all the relevant tests specified in Annex A (normative).

1

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.

OGC 06-121r3, *OpenGIS® Web Services Common Specification*, version 1.1.0

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

OGC 07-067r5, *OpenGIS® Web Coverage Service Implementation Standard*, version 1.1.2

NOTE   This WCS Standard contains a list of normative references that are also applicable to this extension Standard.

OGC 07-092r1, *Definition identifier URNs in OGC namespace*, version 1.1.2

OGC 08-068r2, *Web Coverage Processing Service (WCPS) Interface Standard*, version 1.0.0

In addition to this document, this standard includes the normative XML Schema Document file specified in Annex B.

4   Terms and definitions

For the purposes of this document, the terms and definitions given in the above references (in particular: WCS [OGC 07-067r5] and WCPS [OGC 08-068r2]) apply.

5   Conventions

5.1   Symbols (and abbreviated terms)

Most of the abbreviated terms listed in Subclause 5.1 of the OWS Common Specification [OGC 05-008] also apply to this document.

Further, this document uses the terms and concepts of WCS [OGC 07-067r5] and WCPS [08-068r2].

5.2   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 the OGC Web Services Common Specification [OGC 06-121r3].

5.3   Platform-neutral and platform-specific specifications

As specified in Clause 10 of OGC Abstract Specification Topic 12 "OpenGIS Service Architecture" (which contains ISO 19119), this document includes both Distributed Computing Platform-neutral and platform-specific specifications. This document first specifies each operation request and response in platform-neutral fashion. This is done using a table for each data structure, which lists and defines the parameters and other data structures contained. These tables serve as data dictionaries for the UML model in Annex C, and thus specify the UML model data type and multiplicity of each listed item.

Example   Platform-neutral specifications are contained in Clause 6.

The specified platform-neutral data could be encoded in many alternative ways, each appropriate to one or more specific DCPs. This document specifies encodings appropriate for use of HTTP GET transfer of operations requests (using KVP encoding), and for use of HTTP POST transfer of operations requests (using KVP or XML or SOAP encoding). However, the same operation requests and responses (and other data) could be encoded for other specific computing platforms.

Example   Platform-specific specifications for XML and KVP encoding are contained in Clause 7.

6   ProcessCoverages extension implementation model

6.1   Overview

This extension specifies an additional WCS request type, *ProcessCoverages*, in addition to the WCS request types *GetCapabilities*, *DescribeCoverage*, and *GetCoverage*.

The *ProcessCoverages* operation defines how to request processing of multi-dimensional grid coverage data sets on a WCS server and to retrieve the results – which can be coverages or coverage metadata – over the World Wide Web. To this end, operation requests use a formally defined processing language which supports coverage expressions of unlimited complexity. Result coverages can be transmitted directly back to the client or made available for download by URLs communicated to the client.

Example   The following WCPS expression retrieves the difference between red and green channels of coverages Modis1, Modis2, and Modis3, encoded in NetCDF (the format name may vary, depending on the name specified in the NetCDF format encoding extension specification):

```
for $c in ( Modis1, Modis2, Modis3 )
return
    encode( abs( $c.red – $c.green ), "netcdf" )
```

The syntax used is the one defined in [OGC 08-068], which is not suitable for immediate shipping. For sending it as a ProcessCoverage request to a WCS server this expression must be encoded as specified in Section **Fehler! Verweisquelle konnte nicht gefunden werden.**.

The above expression immediately returns the encoded coverages as ProcessCoverages response. Alternatively, the store() function may be invoked to store the results server-side. In this case, the response contains references to the result files for subsequent download by the client:

```
        for $c in ( Modis1, Modis2, Modis3 )
        return
            store( encode( abs( $c.red – $c.green ), "netcdf" ) )
```

Coverages advertised by a service can be stored on the corresponding server, but the service may well itself rely on external data sources to substantiate the portfolio. In any case, the appearance towards the service clients always is one homogeneously accessible set of offered coverages.

Figure 1 is a simplified UML diagram summarizing the WCPS interface. This class diagram shows that the *WebCoverageServer* interface class inherits the *getCapabilities* operation from the *OGCWebService* interface class, and adds the *getCoverage* and *describeCoverage* operations[1]. This diagram also shows the new *WebCoverageProcessingServer* interface class, added by this extension, which inherits the *getCapabilities*, *getCoverage*, and *describeCoverage* operations from the *WebCoverageServer* interface class, and adds the *processCoverages* operation. See Annex C for a complete presentation of the WCS Processing package class diagram.

6.2   ProcessCoverages operation

The *ProcessCoverages* operation allows WCS clients to request processing of one or more coverages available on a WCS server. A WCPS server evaluates a *ProcessCoverages* request and returns an appropriate response to the client. The result returned to the client upon a successful request consists of an ordered sequence of one or more coverages or scalar values. This operation is required implementation by WCS servers that implement this extension Standard.

Each WCPS server **shall** implement the *ProcessCoverages* operation.

While the WCS *GetCoverage* operation allows retrieval of a coverage from a coverage offering, possibly modified through operations like spatial, temporal, and band subsetting and coordinate transformation, the *ProcessCoverages* operation extends this functionality through more powerful processing capabilities. This includes, on the one hand, further coverage processing primitives and, on the other hand, nesting of function application, thereby allowing for arbitrarily complex processing requests.

NOTE 1   WCPS has been designed so as to be "safe in evaluation" – i.e., implementations are possible where any valid WCPS request can be evaluated in a finite number of steps, based on the operation primitives. Hence, WCPS implementations can be constructed in a way that no single request can render the service permanently unavailable. Notwithstanding, it still is possible to send requests that will impose extremely high workload on a server.

NOTE 2   Data items within a WCPS response list can be heterogeneous in size and structure. In particular, the coverages within a response list can have different dimensions, domains, range types, etc. However, a response always consists of either coverages or scalar values.

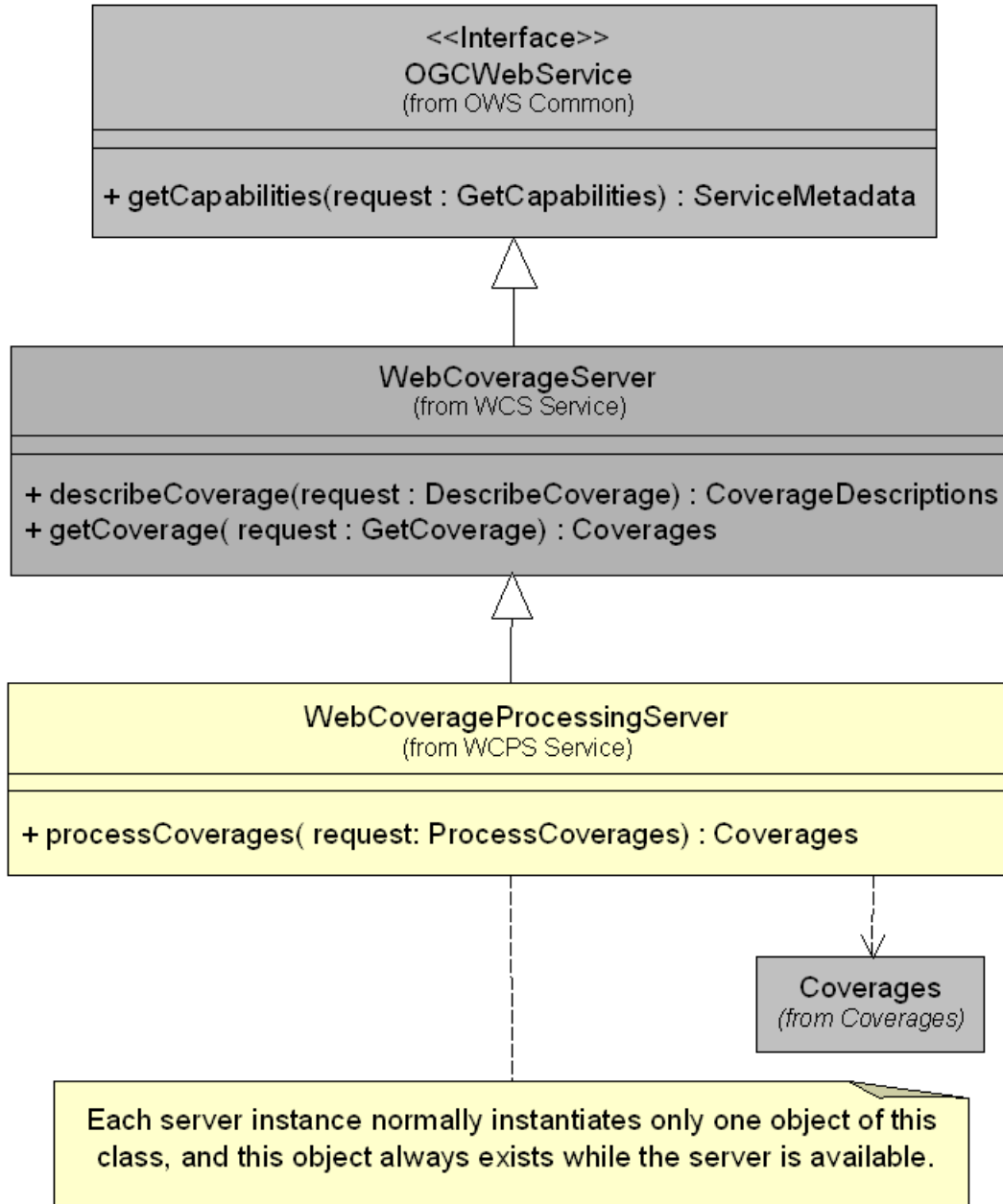[1] This capitalization of names uses the OGC/ISO profile of UML.

Figure 1 — Simplified WCPS interface UML diagram

NOTE 3   As the functionality of WCPS centers around coverage processing, metadata are considered only to the extent necessary for a coherent service. This way WCPS keeps orthogonal to other OGC standards.

6.2.1   Processing request contents

A request to perform the *ProcessCoverages* operation **shall** include the parameters and data structure shown graphically in the UML diagram in Figure 2 and specified in [OGC 08-068r2].

Table 1 — Parameters in *ProcessCoverages* operation request

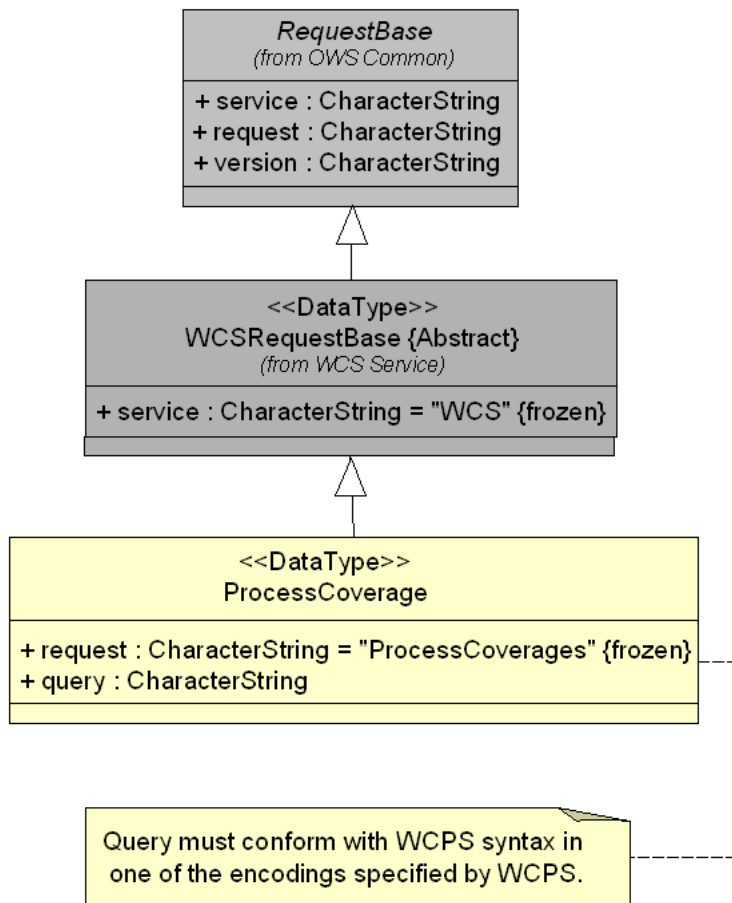| Names	Definition	Data type and value	Multiplicity
Service service	Service type identifier	Character String type **Must** be "WCS"	One (mandatory)
Request request	Operation name	Character String type **Must** be "ProcessCoverages"	One (mandatory)
Version version	Specification version for operation	Character String type, not empty. Value is specified by each Specification version	One (mandatory)
Query query	Operation(s) to be executed	Character String type. **Must** conform with WCPS syntax in chosen, supported encoding	One (mandatory)

Figure 2 — ProcessCoverages operation request UML class diagram

6.2.2 ProcessCoverages response

The *ProcessCoverages* operation response **shall** consist of one of the following alternatives:

- ▪ If the result of expression evaluation is of type coverage: a coverage, encoded in a particular data format, or a sequence of encoded coverages;

- ▪ If the result of expression evaluation is not of type coverage: A single value, or record of values, or a sequence of such single values or value records.

6.2.3 Evaluation exceptions

Whenever a coverage expression cannot be evaluated according to the rules specified in document [08-068r2], the Web Coverage Processing Server **shall** respond with an exception.

Example   The following request fragments will lead to an exception when used in a *ProcessCoverages* request (reasons: division by zero; square root of a negative number):

```
C / 0

sqrt( - abs( C ) )
```

7   WCPS extension request encodings

A *ProcessCoverages* request **shall** contain exactly one valid WCPS expression, encoded in one of the structures as described in Subclause 7.1.

The server **shall** answer with a response as described in Subclause 7.2.

7.1   Request encodings

A WCPS server **shall** support at least one of the KVP and SOAP request encodings as specified in this subclause. If SOAP request encoding is supported then the server **shall** also support SOAP response encoding.

7.1.1   KVP request encoding

When employing the key-value pair encoding clients **shall** use the HTTP GET method for transmitting *ProcessCoverages* requests.

Table 2 specifies the complete *ProcessCoverages* Request.

7.1.1.1   SERVICE=WCPS / VERSION=*version*

The SERVICE parameters **shall** be "WCS" ; any upper/lower case combination **may** be used. The VERSION parameter **shall** refer to the WCPS protocol version the server implements.

7.1.1.2 REQUEST=ProcessCoverages

The value **shall** be "ProcessCoverages"; any upper/lower case combination **may** be used.

7.1.1.3 QUERY=*expr*

The QUERY argument is a WCPS expression which **shall** conform with the syntax specified in WCPS [OGC 08-068] and with all characters not allowed in URLs properly escaped. For the URL encoding the pertaining IETF rules [IETF RFC 2616] **shall** be used.

Table 2 — The ProcessCoverages request expressed as Key-Value Pairs

URL Component	Description	Multiplicity
http://server_address/path/script?	URL of WCS server.	One (mandatory)
SERVICE=WCS	Service name. **Must** be "WCS".	One (mandatory)
VERSION=*m.n.p*	Request protocol version, *m*, *n*, *p* being nonnegative integer numbers.	One (mandatory)
REQUEST=ProcessCoverages	Name of the request. **Must** be "ProcessCoverages".	One (mandtory)
QUERY=*expr*	The expression describing the result coverage(s) derived from the coverage offering. **Shall** conform with Subclause 7.1.1.3.	One (mandatory)
(Vendor-specific parameters)	**Shall not** influence semantics of the request, but can lead to further exceptions. Default: none	*Optional*

Example   Vendor-specific parameters might carry authentication information which, in case of unsuccessful authentication, will lead to a specific exception not specified by WCS nor WCPS.

7.1.2 SOAP request encoding

When using the SOAP encoding, requests **shall** use the SOAP protocol for communication as specified Annex E in [OGC 07-067r5], based on the XML schema indicated in Annex B of this document.

7.2 Response encodings

7.2.1 Response structure

Depending on its result type, the normal response to a valid *ProcessCoverages* request **shall** consist of one of the following alternatives:

- If the result of expression evaluation is of type coverage, then the response **shall** be encoded in the Coverages response structure specified in WCS [OGC 07-067r5] Annex H.2.

If function `store()` as defined in Subclause 7.1.3 of document [08-068r2] is not used in the request, then a response **shall** be generated as specified in [OGC 07-067r5] Subclause 10.3.11 for request parameter "store=false". If function `store()` is used in the request, then an XML response **shall** be generated by the server and the response coverage data files **shall** be made accessible through the URLs communicated by the server as specified in [OGC 07-067r5] Subclause 10.3.11 for request parameter "store=true".

NOTE   Using the store() function is equivalent to indicating "store=true" in WCS GetCoverage request (see [OGC 07-067r5] Table 24).

Each coverage of the processing result set **shall** be encoded in one file.

NOTE   This restriction over WCS allows to distinguish the result coverages from each other. In future versions of this standard a variant of the manifest may be introduced which allows logical bundling of separately encoded parts of a coverage by other means.

▪ If the result of expression evaluation is not of type coverage, then the response **shall** be an ordered sequence of strings separated by newline characters. For a newline character, a server may use either the ASCII character CR (0x0D), or LF (0x0A), or CR immediately followed by LF. In this case, the HTTP Content type **shall** be "text/plain".

NOTE   In future versions of this standard this plain text encoding may get replaced by some other data structure, such as XML or JSON.

In an HTTP environment, the returned value **shall** have a Content-type entity header that matches the format of the return value.

When using the SOAP encoding, responses **shall** use the SOAP protocol for communication as specified Annex E in [OGC 07-067r5].

7.2.2   Exceptions

An invalid *ProcessCoverages* request **shall** yield an error output, either as a WCS exception reported in the OWS Common [OGC 06-121r3] Exceptions format (in case of a KVP or XML request), or as a SOAP Fault message (in case of a SOAP request), or as a network protocol error response.

An exception report message **shall** be returned as specified in Clause 8 of [OGC 06-121r3]. An exception code returned **shall** be a standard exception code, either as as defined in [OGC 06-121r3] Table 25 or as listed in Table 3. For each exceptionCode listed in Table 3, the contents of the "locator" parameter value **shall** be as specified in the right column of Table 3.

Table 3 — Exception codes for ProcessCoverages operation[a]

exceptionCode value	Meaning of code	"locator" value
SyntaxError	Operation request is syntactically malformed.	Offending token, optionally expected token and position of error
IllegalRequest	Operation request is semantically wrong and, hence, cannot be processed.	Further details on error reason
ProcessingError	Operation request is syntactically and semantically correct, but cannot be answered due to server-side problems.	Further details on error reason (optional)

Annex A
(normative)

Abstract Test Suite

The Abstract Test Suite for WCPS is provided in [OGC 08-053r2].

Annex B
(**normative**)

WCPS XML Schemas

B.1 ProcessCoverages Request and Response Schema

The request and response schemas are specified in files wcpsAll.xsd and wcpsProcess-Coverages.xsd.

Annex C
(normative)

UML Diagram

This annex provides a UML model of the WCPS interface, using the OGC/ISO profile of UML summarized in Subclause 5.2 of the OWS Common [OGC 05-008]. The UML model of the WCS processing package is based on the model of WCS [OGC 07-067r5].
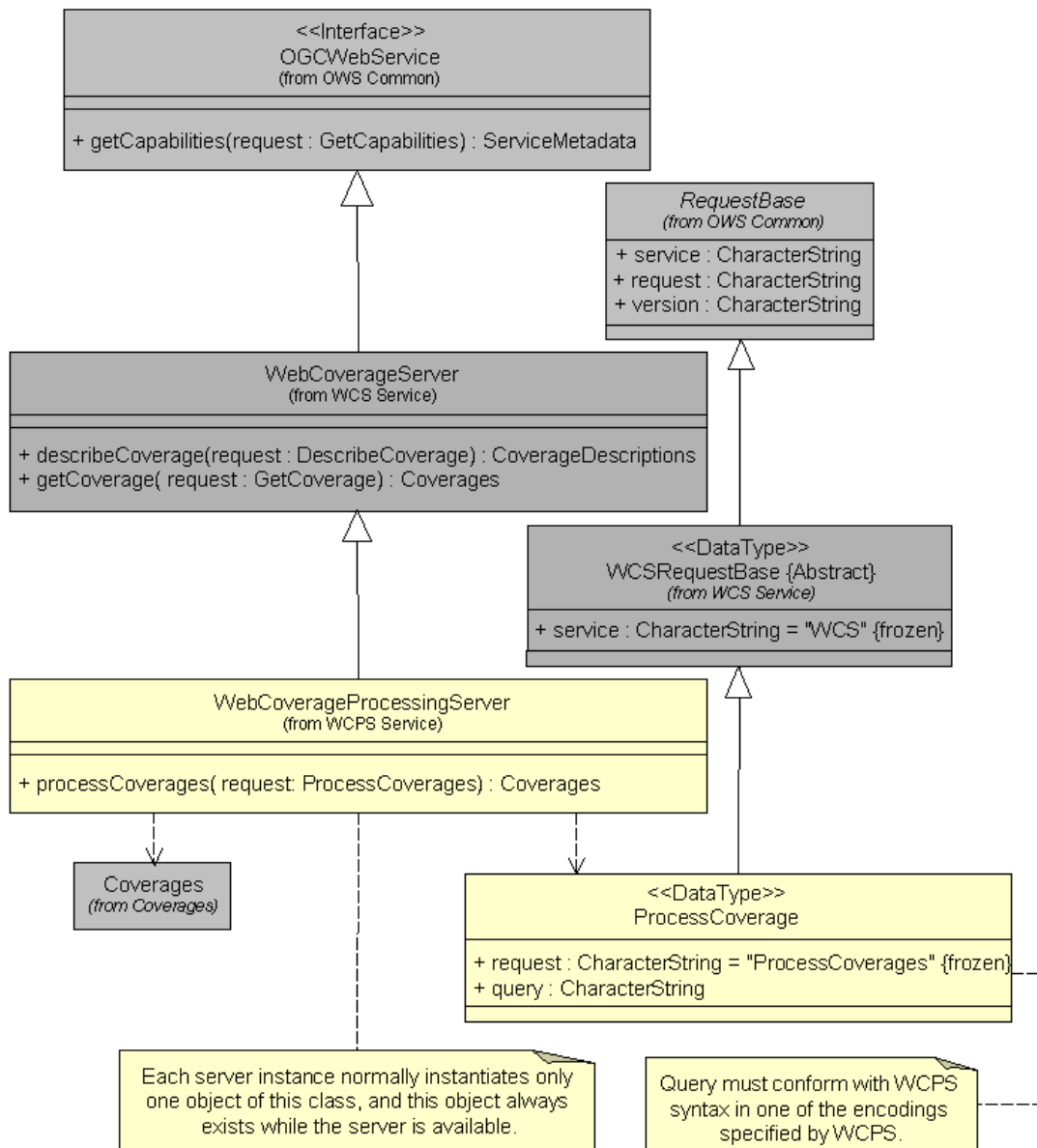

Figure C.1 — WCS Processing package class diagram

13