

**OGC® DOCUMENT: 14-065R2**

External identifier of this OGC® document: <http://www.opengis.net/doc/IS/wps/2.0.2>

**OGC®**  
Making location count.

# OGC® WPS 2.0.2 INTERFACE STANDARD CORRIGENDUM 2

---

**STANDARD**  
Implementation

**APPROVED**

**Version:** 2.0.2

**Submission Date:** 2014-11-25

**Approval Date:** 2015-01-27

**Publication Date:** 2015-03-05

**Editor:** Matthias Mueller, Benjamin Pross

**Notice:** This document is an OGC Member approved international standard. This document is available on a royalty free, non-discriminatory basis. 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.

## 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.

Suggested additions, changes and comments on this document are welcome and encouraged. Such suggestions may be submitted using the online change request form on OGC web site: [http://portal.opengeospatial.org/public\\_ogc/change\\_request.php](http://portal.opengeospatial.org/public_ogc/change_request.php)

## Copyright notice

Copyright © 2015 Open Geospatial Consortium  
To obtain additional rights of use, visit <http://www.ogc.org/legal/>

## Note

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.

# CONTENTS

I.	ABSTRACT .....	vi
II.	KEYWORDS .....	vi
III.	PREFACE .....	vii
IV.	SECURITY CONSIDERATIONS .....	viii
V.	SUBMITTING ORGANIZATIONS .....	ix
VI.	SUBMITTERS .....	ix
VII.	INTRODUCTION .....	ix
1.	SCOPE .....	2
2.	CONFORMANCE .....	4
3.	NORMATIVE REFERENCES .....	6
4.	TERMS AND DEFINITIONS .....	8
5.	CONVENTIONS .....	11
5.1.	Abbreviated terms .....	11
5.2.	Use of the Term “Process” .....	11
5.3.	UML Notation .....	12
5.4.	Namespace Conventions .....	12
6.	WPS CONCEPTUAL MODEL .....	14
6.1.	Service Discovery .....	15
6.2.	Service Capabilities .....	15
7.	WPS NATIVE PROCESS MODEL .....	18
7.1.	Common Description Type .....	18
7.2.	Data Description Structure .....	20
8.	WPS NATIVE PROCESS MODEL ENCODING .....	23
8.1.	XML Schema Implementation .....	23
8.2.	Plain Text Encoding for LiteralData and BoundingBoxData Values .....	25
9.	COMMON WPS SERVICE MODEL .....	28

9.1. Overview of WPS Core Operations .....	29
9.2. Data Transmission .....	30
<b>ANNEX A (NORMATIVE) ABSTRACT TEST SUITE .....</b>	<b>34</b>
A.1. Basic WPS (Conformance Class) .....	34
<b>ANNEX B (INFORMATIVE) XML EXAMPLES .....</b>	<b>36</b>
B.1. Data Types .....	36
<b>BIBLIOGRAPHY .....</b>	<b>39</b>

## LIST OF TABLES

---

Table 1 – Conformance classes related to WPS 2.0 .....	4
Table 2 – Properties of the DescriptionType structure .....	19
Table 3 – Properties of the Metadata structure .....	20
Table 4 – Format properties .....	21
Table 5 – Parts of the inline Data structure .....	31
Table 6 – Properties of the DataEncodingAttributes structure .....	31
Table 7 – Parts of the Reference structure .....	32
Table 8 – Parts of the RequestBody structure .....	32
Table 9 – Parts of the BodyReference structure .....	32

## LIST OF FIGURES

---

Figure 1 – Artifacts of the WPS service model .....	14
Figure 2 – DescriptionType for processes, process inputs and process outputs UML class diagram .....	19
Figure 3 – DataDescription and supported formats UML class diagram .....	21
Figure 4 – Common sequence of WPS operations UML sequence diagram .....	30
Figure 5 – Input and output data transmission structures UML class diagram .....	31

## LIST OF RECOMMENDATIONS

---

REQUIREMENTS CLASS 1 .....	14
REQUIREMENTS CLASS 2 .....	15

REQUIREMENTS CLASS 3 .....	15
REQUIREMENTS CLASS 4 .....	18
REQUIREMENTS CLASS 5 .....	18
REQUIREMENTS CLASS 6 .....	20
REQUIREMENTS CLASS 7 .....	23
REQUIREMENTS CLASS 8 .....	23
REQUIREMENTS CLASS 9 .....	24
REQUIREMENTS CLASS 10 .....	24
REQUIREMENTS CLASS 11 .....	25
REQUIREMENTS CLASS 12 .....	28
REQUIREMENTS CLASS 13 .....	30
REQUIREMENT A.1 .....	34
RECOMMENDATION A.1 .....	34
REQUIREMENT A.2 .....	34



## ABSTRACT

---

In many cases geospatial or location data, including data from sensors, must be processed before the information can be used effectively. The OGC Web Processing Service (WPS) Interface Standard provides a standard interface that simplifies the task of making simple or complex computational processing services accessible via web services. Such services include well-known processes found in GIS software as well as specialized processes for spatio-temporal modeling and simulation. While the OGC WPS standard was designed with spatial processing in mind, it can also be used to readily insert non-spatial processing tasks into a web services environment.



## KEYWORDS

---

The following are keywords to be used by search engines and document catalogues.

TBD, geoprocessing, ogcdoc, OGC document, processes, processing, WPS



## PREFACE

---

This standard is a continuation of WPS 1.0, a standard for web-based processing of geospatial data. It incorporates a range of change requests that have been submitted since the release of WPS 1.0 and further follows the OGC standard for modular specifications [OGC 08-131r3].

In contrast to the prior version, WPS 2.0 provides a core conceptual model that may be used to specify a WPS in different architectures such as REST or SOAP.

The WPS process model has been encapsulated into separate requirements and conformance classes, so it may be used independently from WPS servers in process catalogs and metadata records. The expressive power of process descriptions has been enhanced by permitting structured (or nested) inputs and outputs. The concept of process profiles has been clarified and extended to support process descriptions at different levels of abstraction.

Conversely, the process model itself has been largely decoupled from the WPS protocol, allowing the use of other domain-specific descriptions of processes, e.g. those defined in SensorML, and to execute them on a WPS server.

This specification also provides a Basic WPS conformance class that comprises the synchronous and asynchronous execution protocol, the WPS process model, and implements HTTP/POST+XML and HTTP/GET+KVP encodings.

Future work will target the definition of process interfaces for common processes based on the process model conformance class. Such profiles will encourage the development of well-defined, reliable, interoperable and exchangeable process implementations.

If OGC baseline and related specifications should further progress towards REST-oriented interfaces, the development of a REST-oriented WPS interface standard should be considered.



## SECURITY CONSIDERATIONS

---

No security considerations have been made for this document.



## V

# SUBMITTING ORGANIZATIONS

---

The following organizations submitted this Document to the Open Geospatial Consortium (OGC):

- TU Dresden
- 52°North
- Intergraph Corporation
- Image Matters LLC
- CRP Henri Tudor
- Airbus Defence and Space

## VI

# SUBMITTERS

---

NAME	REPRESENTING	OGC MEMBER
Matthias Müller	TU Dresden	Yes
Benjamin Pross	52°North	Yes
Stan Tillman	Intergraph Corporation	Yes
Jeff Yutzler	Image Matters LLC	Yes
Luís de Sousa	CRP Henri Tudor	Yes
Arnaud Cauchy	Airbus Defence & Space	Yes

## VII

# INTRODUCTION

---

The WPS standard provides a robust, interoperable, and versatile protocol for process execution on web services. It supports both immediate processing for computational tasks that take little time and asynchronous processing for more complex and time consuming tasks. Moreover, the

WPS standard defines a general process model that is designed to provide an interoperable description of processing functions. It is intended to support process cataloguing and discovery in a distributed environment.

1

# SCOPE

---

This document specifies the interface to a general purpose Web Processing Service (WPS). A WPS is a web service that enables the execution of computing processes and the retrieval of metadata describing their purpose and functionality. Typically, these processes combine raster, vector, and/or coverage data with well-defined algorithms to produce new raster, vector, and/or coverage information.

The WPS protocol supports both synchronous and asynchronous execution of processes. Synchronous execution may be used in simple and quick computation scenarios, where the data processing takes little to almost no time. Asynchronous processing is particularly well suited for complex computation scenarios which may take significant time.

The specification uses a core and extensions model to organize its features:

1. A core conceptual model, defining basic requirements for a web based processing service,
2. A process model to support the description and discovery of processes on the web,
3. A basic data model that supports arbitrary (standard or non-standard) data formats for inputs and outputs,
4. A WPS service model and encoding based on OGC baseline standards, and
5. A Dismiss extension to allow clients to terminate asynchronous processing jobs.

2

# CONFORMANCE

---

## CONFORMANCE

Conformance with this standard shall be checked using all the relevant tests specified in Annex A (normative) of this document. The framework, concepts, and methodology for testing, and the criteria to be achieved to claim conformance are specified in the OGC Compliance Testing Policies and Procedures and the OGC Compliance Testing web site<sup>1</sup>.

**Table 1** – Conformance classes related to WPS 2.0

CONFORMANCE CLASS	DESCRIPTION	CLAUSE
<a href="http://www.opengis.net/spec/WPS/2.0/conf/service/profile/basic-wps">http://www.opengis.net/spec/WPS/2.0/conf/service/profile/basic-wps</a>	Basic WPS service profile	Annex A.1
<a href="http://www.opengis.net/spec/WPS/2.0/conf/service/synchronous-wps">http://www.opengis.net/spec/WPS/2.0/conf/service/synchronous-wps</a>	Synchronous WPS	
<a href="http://www.opengis.net/spec/WPS/2.0/conf/service/asynchronous-wps">http://www.opengis.net/spec/WPS/2.0/conf/service/asynchronous-wps</a>	Asynchronous WPS	
<a href="http://www.opengis.net/spec/WPS/2.0/conf/process-model-encoding">http://www.opengis.net/spec/WPS/2.0/conf/process-model-encoding</a>	WPS process model encoding	
<a href="http://www.opengis.net/spec/WPS/2.0/conf/service/dismiss-extension">http://www.opengis.net/spec/WPS/2.0/conf/service/dismiss-extension</a>	Dismiss extension	

<sup>1</sup>[www.opengeospatial.org/cite](http://www.opengeospatial.org/cite)

3

# NORMATIVE REFERENCES

---

## NORMATIVE REFERENCES

---

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

Arliss Whiteside Jim Greenwood: OGC 06-121r9, *OGC Web Service Common Implementation Specification*. Open Geospatial Consortium (2010). [https://portal.ogc.org/files/?artifact\\_id=38867](https://portal.ogc.org/files/?artifact_id=38867).

Policy SWG: OGC 08-131r3, *The Specification Model – Standard for Modular specifications*. Open Geospatial Consortium (2009). [https://portal.ogc.org/files/?artifact\\_id=34762&version=2](https://portal.ogc.org/files/?artifact_id=34762&version=2).

A. Phillips, M. Davis: IETF RFC 4646, *Tags for Identifying Languages*. (2006). <https://www.rfc-editor.org/info/rfc4646>.

T. Berners-Lee, R. Fielding, L. Masinter: IETF RFC 3986, *Uniform Resource Identifier (URI): Generic Syntax*. (2005). <https://www.rfc-editor.org/info/rfc3986>.

ISO: ISO 8601:2004, *Data elements and interchange formats – Information interchange – Representation of dates and times*. International Organization for Standardization, Geneva (2004). <https://www.iso.org/standard/40874.html>.

W3C xmlschema-2, *XML Schema Part 2: Datatypes Second Edition*. <https://www.w3.org/TR/xmlschema-2/>.





4

# TERMS AND DEFINITIONS

---

## TERMS AND DEFINITIONS

---

This document uses the terms defined in [OGC Policy Directive 49](#), 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 document and OGC documents do not use the equivalent phrases in the ISO/IEC Directives, Part 2.

This document also uses terms defined in the OGC Standard for Modular specifications ([OGC 08-131r3](#)), also known as the ‘ModSpec’. The definitions of terms such as standard, specification, requirement, and conformance test are provided in the ModSpec.

For the purposes of this document, the following additional terms and definitions apply.

This document uses the terms defined in Sub-clause 5.3 of [[OGC 06-121r9](#)], 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.

For the purposes of this document, the following additional terms and definitions apply.

### 4.1. dataset

---

collection of data, published or curated by a single agent, and available for access or download in one or more formats

**Note 1 to entry:** The use of ‘collection’ in the definition from [DCAT] is broader than the use of the term collection in this specification.

[SOURCE: W3C vocab-dcat]

### 4.2. distribution

---

represents an accessible form of a **dataset**

Example      a downloadable file, an RSS feed or an API.

[SOURCE: W3C vocab-dcat]

## 4.3. feature

characteristic ADMITTED

---

DEPRECATED: character  
abstraction of real world phenomena

**Note 1 to entry:** For those unfamiliar with the term 'feature', the explanations on [Spatial Things, Features and Geometry](#) in the W3C/OGC Spatial Data on the Web Best Practice document provide more detail.

[**SOURCE:** ISO 19101-1:2014, modified – added alternative term for “characteristic” and deprecated term “character”.]

## 4.4. Process

---

A **process**  $p$  is a function that for each input returns a corresponding output

$$\mu: X \rightarrow Y$$

where  $X$  denotes the domain of arguments  $x$  and  $Y$  denotes the co-domain of values  $y$ . Within this specification, process arguments are referred to as process inputs and result values are referred to as process outputs. Processes that have no process inputs represent value generators that deliver constant or random process outputs.

## 4.5. Process description

---

A process description is an information model that specifies the interface of a process. A process description is used for a machine-readable description of the process itself but also provides some basic information about the process inputs and outputs.



5

# CONVENTIONS

---

## CONVENTIONS

---

This section provides details and examples for any conventions used in the document. Examples of conventions are symbols, abbreviations, use of XML schema, or special notes regarding how to read the document.

### 5.1. Abbreviated terms

---

GML	Geography Markup Language
GRS	Coordinate Reference System
HTTP	Hypertext Transfer Protocol
ISO	International Organization for Standardization
KVP	Keyword Value Pair
MIME	Multipurpose Internet Mail Extensions
OGC	Open Geospatial Consortium
UML	Unified Modeling Language
URI	Universal Resource Identifier
URL	Uniform Resource Locator
WPS	Web Processing Service
XML	Extensible Markup Language

### 5.2. Use of the Term “Process”

---

The term process is one of the most used terms both in the information and geosciences domain. If not stated otherwise, this specification uses the term process as an umbrella term for any algorithm, calculation or model that either generates new data or transforms some input data into output data as defined in Clause 4.4.

## 5.3. UML Notation

---

Unified Modeling Language (UML) static structure diagrams appearing in this specification are used as described in section 5.2 of OGC06-121r9. Further, the following conventions hold:

- UML elements having a package name of “OWS Common” are those defined in the UML model of OWS Common [OGC 06-121r9].
- UML data type Any is used here as an equivalence to XML’s xsd:any.
- UML elements not qualified with a package name are those defined in this standard.

The UML model data dictionary is specified herein in a series of tables. The contents of the columns in these tables are described in section 5.5 of [OGC 06-121r9]. The contents of these data dictionary tables are normative, including any table footnotes.

## 5.4. Namespace Conventions

---

The following namespaces are used in this document. The prefix abbreviations used constitute conventions used here, but are not normative. The namespaces to which the prefixes refer are normative, however.

PREFIX	NAMESPACE URI	DESCRIPTION
ows	<a href="http://www.opengis.net/ows/2.0">http://www.opengis.net/ows/2.0</a>	OWS Common 2.0 XML Schema
xlink	<a href="http://www.w3.org/1999/xlink">http://www.w3.org/1999/xlink</a>	Definitions for XLINK
xml	<a href="http://www.w3.org/XML/1998/namespace">http://www.w3.org/XML/1998/namespace</a>	XML (required for xml:lang)
xs	<a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a>	XML Schema

6

# WPS CONCEPTUAL MODEL

---

## WPS CONCEPTUAL MODEL

The WPS service model defines basic properties of any WPS server. A WPS server is a web service that provides access to pre-defined processes and provides job control operations to instantiate, control and monitor processing jobs (Figure 1).

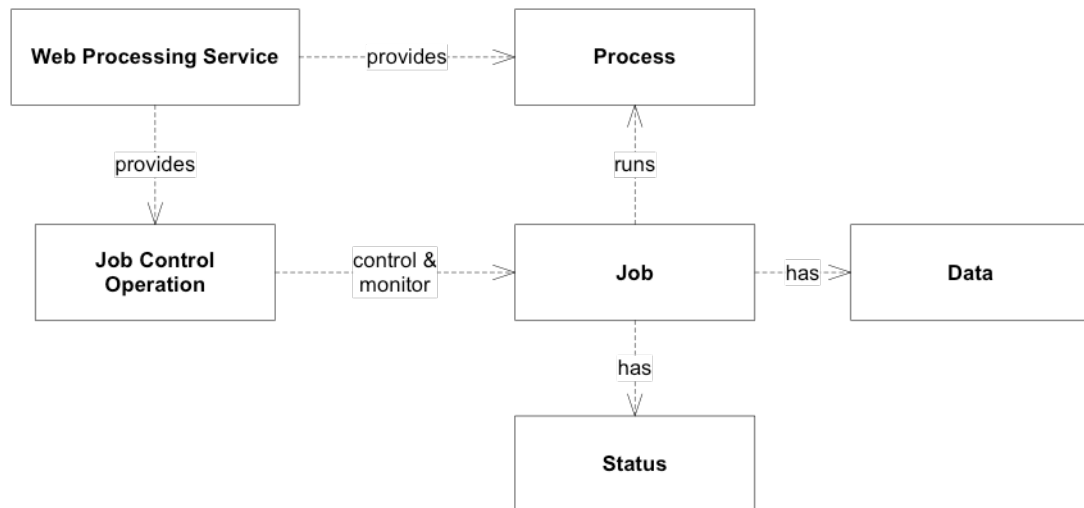


Figure 1 – Artifacts of the WPS service model

### REQUIREMENTS CLASS 1

Obligation requirement

Target type Derived information model, encoding, and software implementation

Requirement Requirement 1-1

Requirement Requirement 1-2

Recommendation Recommendation 1-1

Requirement Requirement 1-3

Requirement Requirement 1-4

Requirement Requirement 1-5

Requirement Requirement 1-6



## 6.1. Service Discovery

---

Any WPS server shall be self-contained, i.e. provide an initial endpoint that can be used by a WPS client to determine the server's capabilities.

### REQUIREMENTS CLASS 2

Obligation requirement

Target type Derived information model, encoding, and software implementation

Requirement Requirement 2-1

Requirement Requirement 2-2

Requirement Requirement 2-3

## 6.2. Service Capabilities

---

The basic capabilities of any WPS server fall into two categories: The first category comprises capabilities for process discovery and retrieval of process descriptions. The second category comprises capabilities to manage and monitor processing jobs.

Since the processes provided by a WPS server may have different degrees of complexity, the server shall indicate the allowed job control capabilities mode per process offering.

Further service capabilities, i.e. for secure communication and user authentication may be provided with the service but are neither covered nor restricted by this specification as long as they do not alter or change the semantics of other job control capabilities.

### REQUIREMENTS CLASS 3

Obligation requirement

Target type Derived information model, encoding, and software implementation

Dependency <http://www.opengis.net/spec/WPS/2.0/req/conceptual-model/process>

Requirement Requirement 3-1

## REQUIREMENTS CLASS 3

Requirement      Requirement 3-2

Requirement      Requirement 3-3

Requirement      Requirement 3-4

7

# WPS NATIVE PROCESS MODEL

---

## WPS NATIVE PROCESS MODEL

This section describes the information model of requirements. The corresponding XML and plain text encodings are specified in Clause 8.

### REQUIREMENTS CLASS 4

Obligation	requirement
Target type	Derived encoding and software implementation
Dependency	<a href="http://www.opengis.net/spec/WPS/2.0/req/conceptual-model/process">http://www.opengis.net/spec/WPS/2.0/req/conceptual-model/process</a>
Requirement	Requirement 4-1
Requirement	Requirement 4-2
Requirement	Requirement 4-3
Requirement	Requirement 4-4
Requirement	Requirement 4-5

### 7.1. Common Description Type

Descriptive elements of processes, inputs and outputs are derived from the BasicIdentificationType provided by OWS Common (Figure 2). Other descriptive information shall be recorded in the Metadata element in the form of simple links with an appropriate role identifier.

### REQUIREMENTS CLASS 5

Obligation	requirement
Target type	Derived encoding and software implementation
Dependency	<a href="http://www.opengis.net/spec/WPS/2.0/req/conceptual-model/process">http://www.opengis.net/spec/WPS/2.0/req/conceptual-model/process</a>
Dependency	OWS Common 2.0 – BasicDescriptionType

## REQUIREMENTS CLASS 5

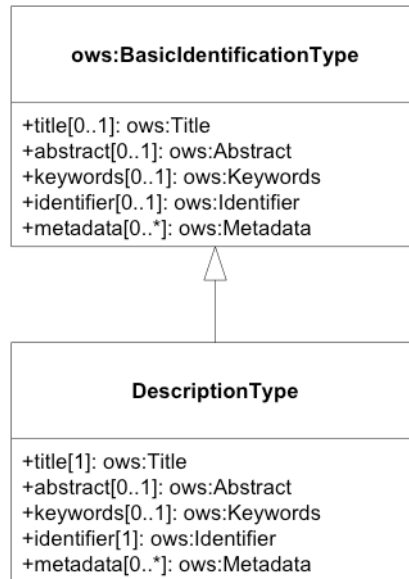
Requirement      Requirement 5-1

Requirement      Requirement 5-2

Requirement      Requirement 5-3

Requirement      Requirement 5-4

Requirement      Requirement 5-5



**Figure 2** – DescriptionType for processes, process inputs and process outputs UML class diagram

**Table 2** – Properties of the DescriptionType structure

NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
Title	Title of the process, input, and output. Normally available for display to a human.	ows:Title	One (mandatory)
Abstract	Brief narrative description of a process, input, and output. Normally available for display to a human.	ows:Abstract	Zero or one (optional) Include when available and useful.
Keywords	Keywords that characterize a process, its inputs, and outputs.	ows:Keywords	Zero or more (optional) Include when available and useful.
Identifier	Unambiguous identifier of a process, input, and output.	ows:Identifier Value is a URI or HTTP-URI <sup>a</sup>	One (mandatory)

NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
Metadata	Reference to additional metadata about this item.	ows:Metadata Allowed values are specified in Table 3.	Zero or more (optional)

<sup>a</sup> Additional content such as separate code space and version attributes in the Identifier element are not allowed.

**Table 3** – Properties of the Metadata structure

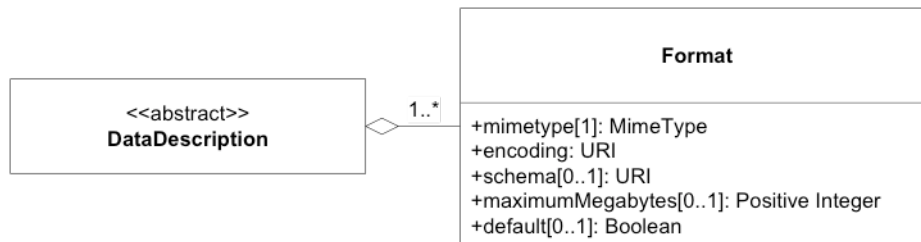
NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
Title	Title of the documentation. Normally available for display to a human.	Character String	One (mandatory)
Link type	Type of the xlink, fixed to simple.	Character String, fixed to “simple”.	One (mandatory)
Role	Role identifier, indicating the role of the linked document.	HTTP-URI	One (mandatory)
href	Reference to a documentation site for a process, input, or output.	HTTP-URI	One (mandatory)

## 7.2. Data Description Structure

The DataDescription structure contains basic properties for defining data inputs and outputs, including mimetype, encoding and schema. These properties specify supported formats for input and output data of computing processes. Any input or output item may support multiple formats, one of which is the default format. Processes may require that an input or output data set does not exceed a certain data volume.

### REQUIREMENTS CLASS 6

Obligation	requirement
Target type	Derived information model, encoding, and software implementation
Dependency	<a href="http://www.opengis.net/spec/WPS/2.0/req/conceptual-model/process">http://www.opengis.net/spec/WPS/2.0/req/conceptual-model/process</a>
Requirement	Requirement 6-1
Requirement	Requirement 6-2



**Figure 3** – DataDescription and supported formats UML class diagram

**Table 4** – Format properties

NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
mimetype	Media type of the data.	Character String	One (mandatory)
encoding	Encoding procedure or character set of the data (e.g. raw or base64)	Character String, fixed to “simple”.	One (mandatory)
schema	Identification of the data schema.	HTTP-URI	One (mandatory)
maximum Megabytes	The maximum size of the input data, in megabytes.	Integer	Zero or one (optional)
default	Indicates that this format is the default format. <sup>a</sup>	Boolean	Zero or one (conditional) <sup>a, b</sup>

<sup>a</sup> Defaults to FALSE if omitted.

<sup>b</sup> One of the formats included in the DataDescription structure shall have the attribute “default” set to “true”.

8

# WPS NATIVE PROCESS MODEL ENCODING

---



## 8.1. XML Schema Implementation

This section specifies the XML encoding of the elements of the WPS native process model. The referred XML schema elements are provided by the associated schema package delivered with this standard and located at <http://schemas.opengis.net/wps/2.0/>.

### REQUIREMENTS CLASS 7

Obligation requirement

Target type Derived encoding and software implementation

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/model>

Dependency OWS Common 2.0

Requirement Requirement 7-1

Requirement Requirement 7-2

Requirement Requirement 7-3

### 8.1.1. Data Types

The XML encoding of data types defines encoding rules for ComplexData, LiteralData, BoundingBoxData as well as their values.

Examples for data type encodings are listed in Annex B.1.

### REQUIREMENTS CLASS 8

Obligation requirement

Target type Derived encoding and software implementation

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/model/datatypes/complex-data>

## REQUIREMENTS CLASS 8

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/model/datatypes/literal-data>

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/model/datatypes/bounding-box-data>

Dependency OWS Common 2.0

Requirement Requirement 8-1

Requirement Requirement 8-2

### 8.1.2. Process Description

This clause specifies the XML encoding for the Process description.

## REQUIREMENTS CLASS 9

Obligation requirement

Target type Derived software implementation

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/model/description>

Dependency OWS Common 2.0

Requirement Requirement 9-1

Requirement Requirement 9-2

### 8.1.3. Generic Process

This clause specifies the XML encoding for the GenericProcess.

## REQUIREMENTS CLASS 10

Obligation requirement

Target type Derived software implementation

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/profile/generic>

## REQUIREMENTS CLASS 10

Dependency OWS Common 2.0

Requirement Requirement 10-1

Requirement Requirement 10-2

## 8.2. Plain Text Encoding for LiteralData and BoundingBoxData Values

This clause specifies the plain text encoding of data types for literal and bounding box data values.

## REQUIREMENTS CLASS 11

Obligation requirement

Target type Derived encoding and software implementation

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/model/datatypes/complex-data>

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/model/datatypes/literal-data>

Dependency <http://www.opengis.net/spec/WPS/2.0/req/native-process/model/datatypes/bounding-box-data>

Dependency OWS Common 2.0

Requirement Requirement 11-1

Requirement Requirement 11-2

```
Literal values - BNF schema:  
literalvalue = value *1("@datatype=" datatype) *1("@uom=" uom)  
value       = 1*VCHAR  
datatype    = URI  
uom         = URI
```

```
Literal values - Example:  
70@datatype=http://www.w3.org/2001/XMLSchema#integer@uom=meter
```

BoundingBox values - BNF schema<sup>2</sup>:  
bbox = lc\_coords "," uc\_coords ["," crs] ["," dimensions]  
lc\_coords = number ["," number]  
uc\_coords = number ["," number]  
number = 1\*DIGIT["." 1\*DIGIT]  
crs = 1\*VCHAR  
dimensions = 1\*DIGIT

BoundingBoxData values - Examples:

51.9,7.0,53.0,8.0,EPSG:4326

51.9,7.0,53.0,8.0,<http://www.opengis.net/def/crs/EPSG/0/4258>

---

<sup>2</sup>The dimensions attribute is included for compliance with the BoundingBox structure defined in OWS Common [OGC 06-121r9]. However, its use should be generally avoided since the number of dimensions is already part of the CRS definition and usually superfluous.



9

# COMMON WPS SERVICE MODEL

---

## COMMON WPS SERVICE MODEL

A Web Processing Service consists of processes and service operations. By definition, processes represent the computational functionality of a WPS, while service operations are used to interact with the WPS and in particular to use the service's process offerings.

### REQUIREMENTS CLASS 12

Obligation	requirement
Target type	Software implementation
Dependency	<a href="http://www.opengis.net/spec/WPS/2.0/req/conceptual-model">http://www.opengis.net/spec/WPS/2.0/req/conceptual-model</a>
Dependency	OWS Common 2.0
Requirement	Requirement 12-1
Requirement	Requirement 12-2
Requirement	Requirement 12-3
Requirement	Requirement 12-4
Requirement	Requirement 12-5
Requirement	Requirement 12-6
Requirement	Requirement 12-7
Requirement	Requirement 12-8
Requirement	Requirement 12-9
Requirement	Requirement 12-10

## 9.1. Overview of WPS Core Operations

---

The WPS interface specified in this section supports retrieval and execution of processes for geospatial computation. For that purpose, the WPS service model specifies the following operations that may be invoked by a WPS client and performed by a WPS server<sup>3</sup>:

**GetCapabilities** – This operation allows a client to request information about the server's capabilities and processes offered.

**DescribeProcess** – This operation allows a client to request detailed metadata on selected processes offered by a server.

**Execute** – This operation allows a client to execute a process comprised of a process identifier, the desired data inputs and the desired output formats.

**GetStatus** – This operation allows a client to query status information of a processing job (conditional).

**GetResult** – This operation allows a client to query the results of a processing job (conditional).

During a sequence of WPS requests, a client should first issue a **GetCapabilities** request to the server to obtain an up-to date listing of available processes. Then, it may issue a **DescribeProcess** request to find out more details about particular processes offered, including the supported data formats. To run a process with the desired input data, a client will issue an **Execute** request<sup>4</sup> (Figure 4).

The operations **GetStatus** and **GetResult** are used in conjunction with asynchronous execution. If a WPS server offers synchronous process execution only, these operations may not be implemented. Detailed guidance is provided by the corresponding profiles and conformance classes.

---

<sup>3</sup>NOTE: Future extensions of this specification may introduce additional operations.

<sup>4</sup>NOTE: A WPS server can change its offering at any time, in particular between a **GetCapabilities**, a subsequent **DescribeProcess**, and a subsequent **Execute** request. Any quality of service (QoS) guarantees are within the responsibilities of the service provider and not covered by this standard.

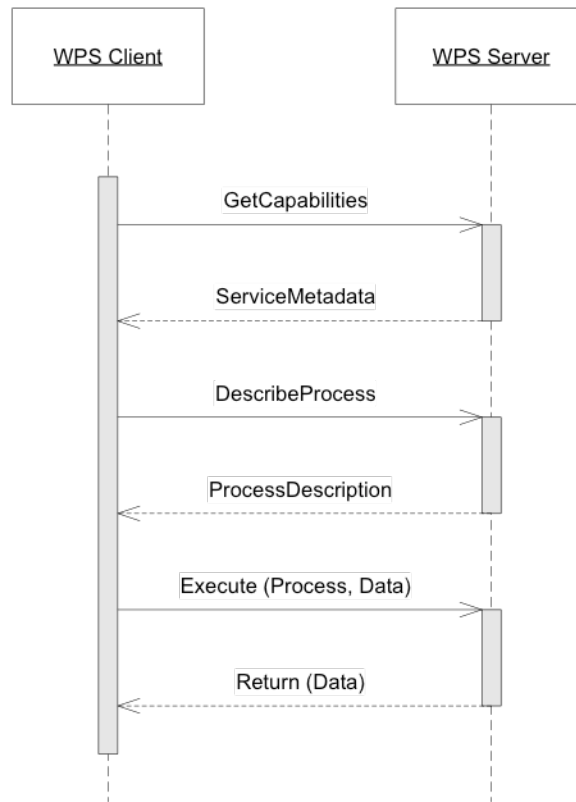


Figure 4 – Common sequence of WPS operations UML sequence diagram

## 9.2. Data Transmission

Data exchange between WPS clients and servers requires an agreement on the general data exchange patterns and suitable communication protocols. Data may be sent to (received from) a WPS in two distinct ways: (1) by reference (using HTTP/GET or HTTP/POST), and (2) by value. Clients may send input data in either fashion. Output data may be requested in any fashion declared by the data transmission options defined for the process offering. Typically, large data inputs and outputs are delivered by reference.

### REQUIREMENTS CLASS 13

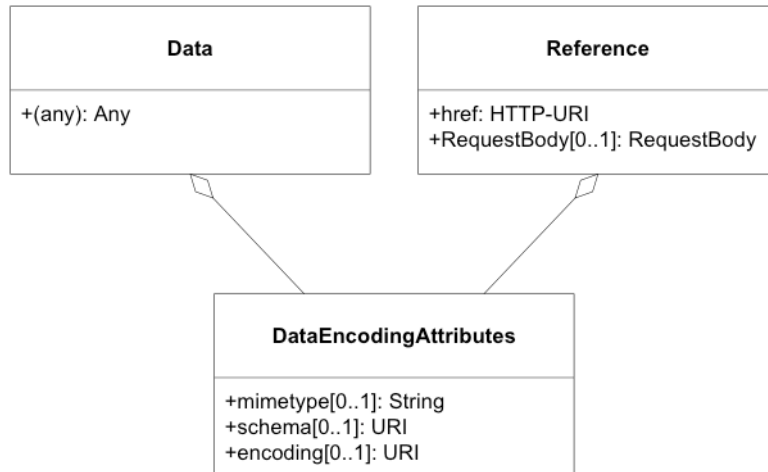
Obligation requirement

Target type Derived encoding and software implementation

Dependency <http://www.opengis.net/spec/WPS/2.0/req/conceptual-model>

Requirement Requirement 13-1





**Figure 5** – Input and output data transmission structures UML class diagram

**Table 5** – Parts of the inline Data structure

NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
mimetype			
encoding	See Table 6 – Properties of the DataEncodingAttributes structure.		
schema			
(any) <sup>a</sup>	The actual input or output data.	Any type and value	One (mandatory)

<sup>a</sup> The data is embedded here as part of the Data element, in the mimeType, encoding, and schema indicated by the first three parameters if they exist, or by the relevant defaults.

**Table 6** – Properties of the DataEncodingAttributes structure

NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
mimetype	MimeType of the data.	Character String	One (mandatory)
encoding	Well-known encoding or character set of the data.	URI “raw” shall be used for plain binary data “base64” shall be used for base64 encoded data Character set identifiers (e.g. “UTF-8”) shall be used for text or CSV data.	Zero or one (conditional) <sup>a</sup>
schema	Identification of the data schema.	URI	Zero or one (conditional) <sup>a</sup>

<sup>a</sup> This shall be provided if: 1) the process data item supports multiple encodings / schemas, and 2) the data is not of the default encoding / schema, and 3a) the schema / encoding cannot be retrieved from the data itself, or 3b) the

encoding / schema information is deeply buried inside the data (i.e. not part of some header) and requires significant parsing effort.

**Table 7 – Parts of the Reference structure**

NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
mimetype			
encoding	See Table 6 – Properties of the DataEncodingAttributes structure.		
schema			
href	HTTP URI that points to the remote resource where the data may be retrieved.	HTTP URI	One (mandatory)
Request Body	Request body element that is used for HTTP/POST requests to the above URL. If no request body is present, an HTTP/GET	Request should be used to retrieve the data. RequestBody structure, see Table 8.	Zero or one (optional)

**Table 8 – Parts of the RequestBody structure**

NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
Body	The contents of this element to be used as the body of the HTTP request message to be sent to the service identified in . ./Reference/@href. For example, it could be an XML encoded WFS request using HTTP/POST.	Any type	Zero or one (conditional) <sup>a</sup>
Body Reference	Reference to a remote document to be used as the body of an HTTP/POST request message to the service identified in the href element in the Reference structure (Table 7).	BodyReference, see Table 9.	Zero or one (conditional) <sup>a</sup>

<sup>a</sup> One and only one of these items shall be included.

**Table 9 – Parts of the BodyReference structure**

NAMES	DEFINITION	DATA TYPE AND VALUES	MULTIPLICITY AND USE
href	HTTP URI that points to the remote resource where the request body may be retrieved.	HTTP URI	One (mandatory)



# ANNEX A (NORMATIVE) ABSTRACT TEST SUITE

---



# ANNEX A (NORMATIVE) ABSTRACT TEST SUITE

---

Tests and requirement identifiers below are relative to <http://www.opengis.net/spec/WPS/2.0>

## A.1. Basic WPS (Conformance Class)

---

The OGC URI identifier of this conformance class is: <http://www.opengis.net/spec/WPS/2.0/conf/service/profile/basic-wps>

Example

### REQUIREMENT A.1

conf/service/profile/basic-wps

### RECOMMENDATION A.1

Verify that the server implements the Basic WPS conformance class.

### REQUIREMENT A.2

Verify that the server implements the Synchronous WPS and/or the Asynchronous WPS conformance class. Verify that the requests and responses to a supported operation are syntactically correct. Verify that the service supports the Synchronous WPS Conformance class, the Asynchronous WPS Conformance class or both. Verify that all process offerings implement the native process model.



# ANNEX B (INFORMATIVE) XML EXAMPLES

---

# B

## ANNEX B (INFORMATIVE) XML EXAMPLES

---

### B.1. Data Types

---

#### B.1.1. Complex Data Description

```
<wps:ComplexData>  
  <wps:Format mimeType="application/geotiff" encoding=  
  "raw"  
  default="true"/>  
  <wps:Format mimeType="application/geotiff" encoding="base64"/>  
</wps:ComplexData>
```

#### B.1.2. Literal Data Description

```
<wps:LiteralData>  
  <wps:Format mimeType="text/plain" default="true"/>  
  <wps:Format mimeType="text/xml"/>  
  <LiteralDataDomain default="true">  
    <ows:AllowedValues>  
      <ows:Range>  
        <ows:MinimumValue>1</ows:MinimumValue>  
        <ows:MaximumValue>1000</ows:MaximumValue>  
      </ows:Range>  
    </ows:AllowedValues>  
    <ows:  
  DataType  
  ows:reference="http://www.w3.org/2001/XMLSchema#float">float  
</ows:DataType>  
    <ows:UOM>meters</ows:UOM>  
    <ows:DefaultValue>100</ows:DefaultValue>  
  </LiteralDataDomain>  
<LiteralDataDomain>  
  <ows:AllowedValues>  
    <ows:Range>  
      <ows:MinimumValue>1</ows:MinimumValue>  
      <ows:MaximumValue>3000</ows:MaximumValue>  
    </ows:Range>  
  </ows:AllowedValues>  
  <ows:  
  DataType  
  ows:reference="http://www.w3.org/2001/XMLSchema#float">float
```

```
</ows:DataType>
  <ows:UOM>feet</ows:UOM>
</LiteralDataDomain>
</wps:LiteralData>
```

### B.1.3. Literal data values

```
<LiteralValue
  dataType=http://www.w3.org/2001/XMLSchema#double
  uom="meter">
  42.1
</LiteralValue>

<LiteralValue
  dataType="http://www.w3.org/2001/XMLSchema#string">
  ArableLand
</LiteralValue>
```

### B.1.4. BoundingBox Data Description

```
<wps:BoundingBoxData>
  <wps:Format mimeType="text/plain" default="true"/>
  <wps:Format mimeType="text/xml"/>
  <wps:SupportedCRS default="true">EPSG:4326</wps:SupportedCRS>
  <wps:SupportedCRS>
    http://www.opengis.net/def/crs/EPSSG/0/4258
  </wps:SupportedCRS>
</wps:BoundingBoxData>
```

### B.1.5. BoundingBox Data Values

```
<ows:BoundingBox crs="EPSG:4326">
  <ows:LowerCorner>51.9 7.0</ows:LowerCorner>
  <ows:UpperCorner>53.0 8.0</ows:UpperCorner>
</ows:BoundingBox>

<ows:BoundingBox
  crs="http://www.opengis.net/def/crs/EPSSG/0/4258">
  <ows:LowerCorner>51.9 7.0</ows:LowerCorner>
  <ows:UpperCorner>53.0 8.0</ows:UpperCorner>
</ows:BoundingBox>
```



# BIBLIOGRAPHY







## BIBLIOGRAPHY

---

1. ISO: ISO 19101, *Geographic information – Reference model*. International Organization for Standardization, Geneva <https://www.iso.org/standard/26002.html>.
2. W3C vocab-dcat, *Data Catalog Vocabulary (DCAT)*. <https://www.w3.org/TR/vocab-dcat/>.