Open Geospatial Consortium

Submission Date: 2020-12-07

Approval Date: 2020-11-30

Publication Date: 2021-02-26

External identifier of this OGC® document: http://www.opengis.net/doc/BP/DGIWG-WFS/2.0

Internal reference number of this OGC® document: 15-005r2

Version: 2.0.2

Category: OGC® Best Practice

Editor: DGIWG

Defence Profile of OGC Web Feature Service 2.0

Warning

This document defines an OGC Best Practice on a particular technology or approach related to an OGC standard. This document is not an OGC Standard and may not be referred to as an OGC Standard. It is subject to change without notice. However, this document is an official position of the OGC membership on this particular technology topic.

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® Best Practice
Document subtype:
Document stage: Approved
Document language: English

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.

DGIWG 122

Defence Profile of OGC's Web Feature Service 2.0

Document type:	Standard
Document subtype:	Implementation Profile
Profile of	WFS 2.0.2 (OGC 09-025r2)
Document date:	30 October 2019
Edition number:	2.0.2
Supersedes:	This document supersedes DGIWG – 122 Ed. 2.0.1, Defence Profile of OGC's Web Feature Service 2.0, dated 28 November 2017.
Responsible Party:	Defence Geospatial Information Working Group (DGIWG)
Audience:	This document is approved for public release and is available on the DGIWG website, http://www.dgiwg.org/dgiwg/.
Abstract:	This document defines the DGIWG profile for the ISO 19142:2010 - Web Feature Service (WFS) including changes made in the OpenGIS Web Feature Service 2.0 Interface Standard - Corrigendum. The Web Feature Service provides access to geospatial features in a manner independent of the underlying data store.
Copyright:	(C) Copyright DGIWG, some rights reserved - (CC) (By:) Attribution You are free: - to copy, distribute, display, and perform/execute the work - to make derivative works - to make commercial use of the work Under the following conditions: - (By:) Attribution. You must give the original author (DGIWG) credit. - For any reuse or distribution, you must make clear to others the license terms of this work. Any of these conditions can be waived if you get permission from the copyright holder DGIWG. Your fair use and other rights are in no way affected by the above. This is a human-readable summary of the Legal Code (the full license is available from Creative Commons <http://creativecommons.org/licenses/by/2.0/ >).

Table of Contents

List of Figures

List of Tables

Executive Summary

This document provides recommended implementation profiles for the ISO 19142:2010 Web Feature Service / Open Geospatial Consortium Web Feature Service Interface Standard (WFS) 2.0 – With Corrigendum. The WFS standard provides an interface allowing requests for geospatial features across the web using platform-independent mechanisms and is independent of the underlying data store. One can think of geospatial features as the "source code" behind a map. Whereas the OGC Web Map Service (WMS) interface or online mapping portals return only an image, which end-users cannot edit or spatially analyze, the WFS provides XML-based Geography Markup Language (GML) as the default payload-encoding for transporting geospatial features. In other words, rather than sharing geographic information at the file level using File Transfer Protocol (FTP), for example, the WFS offers direct fine-grained access to geographic information at the feature and feature property level. The WFS standard specifies discovery operations, query operations, locking operations, transaction operations and operations to manage stored parameterized query expressions. The WFS interface permits users to access and manipulate geospatial feature information from distributed network sources.

Technical specifications sometimes have optional features, such that two conforming implementations may not inter-operate completely due to choosing different sets of optional features to support. Even when no formal optional features exist within a standard, there is still a risk that vendors will not implement functionality that is most important to the military community. Also, some standards contain vague or ambiguous wording thus the development and use of profiles can enforce one possible interpretation. To limit the number of interpretations by implementers and improve interoperability it is possible to define profiles. In standardization, a profile consists of an agreed-upon subset and specific interpretation of a specification.

The intention of the DGIWG WFS 2.0 profiles is to minimize such interoperability issues with a specific view to a military context and to mandate a minimum set of service requirements necessary to ensure usability in an operational coalition environment. These profiles are designed to both increase interoperability between WFS servers and to improve the ease of implementation of the WFS standard.

A survey of DGIWG nations was conducted to determine implementation requirements for WFS. These profiles are in response to those survey results. Nations were asked to identify specific requirements for the type of WFS required (Simple, Basic, Transactional, Transactional with Locking, Manage Stored Queries). Based on the results of this survey the profiles define requirements for both a Basic WFS and a Transactional with Locking WFS. The survey also asked respondents to identify requirements for query filters, bindings, bandwidth constraints, output formats and quality of service. Analyses of the results of the survey response have directly influenced the development of this profile.

i. Submitting organizations

For the Defence Geospatial Information Working Group (DGIWG):

Nation	Parent organization
Germany	Bundeswehr Geoinformation Centre (BGIC)
France	Institut Géographique National (IGN)
United States	National Geospatial-Intelligence Agency (NGA)
United Kingdom	Defence Science and Technology Laboratory (DSTL)

ii. Future work

A revision of WFS 2.0 is currently undertaken at the OGC. The revision will produce a version 3.0 of the WFS standard. This DGIWG WFS profile is based on WFS version 2.0.2 and will be updated with future OGC WFS versions.

The WFS 3.0 standard will include several changes like for instance

- 11-087 New Operation: ImportSchema and DeleteFeatureType Operation for managing WFS server
- 12-038 Time slicing for Features: Goal is to add this new functionality directly in the projection clause (Filter Encoding)
- 12-015 Support Units of Measure (UoM): Assists users when requesting features in a certain UoM (e.g. meters, feet, …)

Additionally the DGIWG WFS project team will evaluate the use of compression formats like gzip, zip and others as a future work item.

Another important development with potential use within DGIWG is the OGC Geosynchronization Service standard. This standard allows for a managed update of features in a database and adds notification services. This is contingent upon change request 11-087.

All these issues will have to be considered in a future amendment to the DGIWG WFS 2.0 profiles document.

1. Introduction

The Web Feature Service provides access to geospatial features in a manner independent of the underlying data store. WFS can also provide the capability to perform operations to create, update and delete features from a data store.

This document defines the DGIWG profile for the ISO 19142:2010 - Web Feature Service (WFS) including changes made in the OpenGIS Web Feature Service 2.0 Interface Standard - Corrigendum.

The intention of the profile is to minimize interoperability issues specific to a military context and to mandate a minimum set of service requirements necessary to ensure usability in an operational coalition environment.

The following general categories of requirements for implementing a WFS 2.0 profile have been considered.

- Filter operators (spatial, temporal,…) to generate queries to access data
- Operations and parameters to create, read, update and delete data
- Adaptation to network, bandwidth constraints (response paging, compression)
- Platform and system independent output formats
- Quality of service (performance, capacity or availability)
- Pre-defined queries (use case dependent data set)

2. Scope

The DGIWG WFS profiles minimize interoperability issues specific to a military context and mandate a minimum set of service requirements necessary to ensure usability in an operational coalition environment.

This document defines the DGIWG profile for the OpenGIS Web Feature Service 2.0 Interface Standard – With Corrigendum based on the assessment of the WFS questionnaire.

Table 1: DGIWG WFS profiles overview

WFS Profiles	Operations	Supported Operators	Bindings	Additional
DGIWG Basic WFS	GetCapabilities DescribeFeatureType GetFeature GetPropertyValue ListStoredQueries DescribeStoredQueries	Standard Spatial Temporal	HTTP Get HTTP Post	Response Paging GML 3.2.1 CRS EPSG:4326 Operation constraints
DGIWG Locking (Transactional) WFS	includes all of DGIWG Basic WFS and			
	Transaction LockFeature			

3. Conformance

The WFS 2.0 International Standard specifies the behavior of a service that provides transactions on and access to geographic features. It specifies discovery operations, query operations, locking operations, transaction operations and operations to manage stored parameterized query expressions. The International Standard states that a server supporting a specific operation shall be capable of handling any well-formed request.

The DGIWG WFS profiles require conformance to OGC 09-025r2 and mandate specific operations to be supported by the server. The profiles also describe how the server shall respond, in other words it identifies the set of mandatory elements a server shall provide.

The conformance classes defined below by the DGIWG WFS profiles and lists the tests specified in Annex A that shall be satisfied in order to comply with each class.

The conformance classes also list which, if any, WFS 2.0 (see OGC 09-025r2, Clause 2) conformance tests need to be satisfied with each WFS conformance class. These WFS 2.0 conformance classes also reference the OGC 09-026R2 Filter Encoding Conformance Test(s).

Table 2: DGIWG WFS profiles – Conformance Classes

Conformance class name	**Operation or behavior ***	**OGC / ISO Conformance Test**	**DGIWG WFS Conformance Test**
DGIWG Basic WFS (Annex A 1.1) http://www.dgiwg.org/std/wfs/2.0/conf/basic	**OGC 09-025r2 Basic WFS**: "The server shall implement the Simple WFS conformance class and shall additionally implement the GetFeature operation with the Query action and the GetPropertyValue operation."	OGC 09-025r2, A.1.2	
	OGC 09-025r2 HTTP GET: The server shall implement the Key-Value Pair (HTTP GET/KVP) encoding.	OGC 09-025R2, A.1.5	
	OGC 09-025r2 HTTP POST: The server shall implement the XML (HTTP POST/XML) encoding for the operations that the server offers.	OGC 09-025R2, A.1.6	
	OGC 09-026R2 Minimum Standard Filter Implements the comparison operators: PropertyIsEqualTo, PropertyIsNotEqualTo, PropertyIsLessThan, PropertyIsGreaterThan, PropertyIsLessThanOrEqualTo, PropertyIsGreaterThanOrEqualTo.	OGC 09-026R2, A.5	

Conformance class name	Operation or behavior *	OGC / ISO Conformance Test	DGIWG WFS Conformance Test
	Implements the logical operators. Does not implement any additional functions.		
	OGC 09-026R2 Standard Filter Implements all the comparison and logical operators and may implement one or more additional functions.	OGC 09-026R2, A.6	
	OGC 09-026R2 Minimum Spatial Filter Implements only the BBOX spatial operator.	OGC 09-026R2, A.7	
	OGC 09-026R2 Spatial Filter Implements the BBOX spatial operator and one or more of the other spatial operators.	OGC 09-026R2, A.8	
	OGC 09-026R2 Minimum Temporal Filter Implements only the During temporal operator.	OGC 09-026R2, A.9	
	OGC 09-026R2 Temporal Filter Implements the During temporal operator and one or more of the other temporal operators.	OGC 09-026R2, A.10	
	DGIWG requirements The server shall implement additional DGIWG requirements.		Annex A 1.1 Annex A.2
DGIWG Locking (transactional) WFS (Annex A.1.2) http://www.dgiwg.org/std/wfs/2.0/conf/locking	**OGC 09-025R2 Locking WFS:** The server shall implement the Transactional WFS conformance class and shall implement at least one of the GetFeatureWithLock or LockFeature operations.	OGC 09-025R2, A.1.4	
	OGC 09-025R2 HTTP GET: The server shall implement the Key-Value Pair (HTTP GET/KVP) encoding.	OGC 09-025R2, A.1.5	
	OGC 09-025R2 HTTP POST: The server shall implement the XML (HTTP POST/XML) encoding for the operations that the server offers.	OGC 09-025R2, A.1.6	
	OGC 09-026R2 Minimum Standard Filter	OGC 09-026R2, A.5	

Conformance class name	Operation or behavior *	OGC / ISO Conformance Test	DGIWG WFS Conformance Test
	Implements the comparison operators: PropertyIsEqualTo, PropertyIsNotEqualTo, PropertyIsLessThan, PropertyIsGreaterThan, PropertyIsLessThanOrEqualTo, PropertyIsGreaterThanOrEqualTo. Implements the logical operators. Does not implement any additional functions.		
	OGC 09-026R2 Standard Filter Implements all the comparison and logical operators and may implement one or more additional functions.	OGC 09-026R2, A.6	
	OGC 09-026R2 Minimum Spatial Filter Implements only the BBOX spatial operator.	OGC 09-026R2, A.7	
	OGC 09-026R2 Spatial Filter Implements the BBOX spatial operator and one or more of the other spatial operators.	OGC 09-026R2, A.8	
	OGC 09-026R2 Minimum Temporal Filter Implements only the During temporal operator.	OGC 09-026R2, A.9	
	OGC 09-026R2 Temporal Filter Implements the During temporal operator and one or more of the other temporal operators.	OGC 09-026R2, A.10	
	DGIWG requirements The server shall implement additional DGIWG requirements		AnnexA.1.2 Annex A.3

*OGC 09-025r2 = Web Feature Service 2.0 Interface Standard – With Corrigendum
OGC 09-026r2 = Filter Encoding with Corrigendum

ISO 19142 Web Feature Service 2.0 and ISO 19143 Filter Encoding 2.0 are currently revised in ISO/TC 211 to be inline with the changes made in the OGC Corrigenda.

4. References

4.1. Normative references

ID	Title	Reference	Version
[1]	Web Feature Service 2.0 Interface Standard – With Corrigendum (same as ISO 19142:2010)	OGC 09-025r2	2.0
[2]	Filter Encoding 2.0 Encoding Standard (same as ISO 19143:2010)	OGC 09-026r2	2.0
[3]	OGC Web Services Common Specification 1.1		1.1

4.2. Informative References

Title	Reference	Version
GML Performance Investigation by CubeWerx	OGC 05-050	1.0

5. Terms, definitions, and abbreviations

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

5.1. Definitions

For the purposes of this document, terms and definitions found in WFS 2.0, **ISO 19142** and **[OGC 09-025r2]** apply.

5.2. Abbreviations

CRS	Coordinate Reference System
BBox	Bounding Box
DGIWG	Defence Geospatial Information Working Group
EPSG	European Petroleum Survey Group
FES	Filter Encoding Specification
GML	Geography Markup Language
GZIP	GNU Zip File format
HTTP	Hypertext Transfer Protocol
IETF	Internet Engineering Task Force
ISO	International Organization for Standardization

ISO/DIS	ISO Draft International Standard
KVP	Keyword Value Pairs
MIME	Multipurpose Internet Mail Extension
OGC	Open Geospatial Consortium
OWS	OGC Web Service
SOAP	Simple Object Access Protocol
SPARQL	Simple Protocol And RDF Query Language
SQL	Structured Query Language
UML	Unified Modelling Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
VSP	Vendor Specific Parameter
WFS	Web Feature Service
WMS	Web Map Service
WSDL	Web Services Description Language
XML	Extensible Markup Language
XPath	XML Path Language
XQuery	XML Query

6. OGC WFS 2.0 (informative)

This section is informative and gives an overview of the different WFS conformance classes that can be used to build profiles of the standard.

6.1. OGC WFS 2.0 Conformance Classes

The Web Feature Service standard defines several conformance classes for service implementations. There are 5 conformance classes, which define general WFS implementations. They are **WFS Simple, WFS Basic, Transactional WFS, WFS Locking** and one additional supporting **Manage Stored Queries**. These WFS implementations are hierarchical structured, meaning all capabilities within Simple are captured within Basic and so forth. Figure 1 shows the dependencies between the WFS implementations.

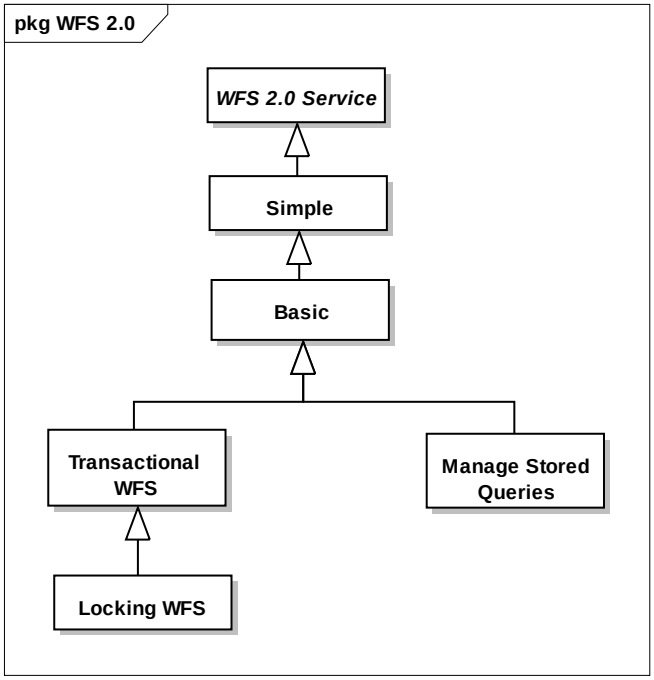

Figure 1: OGC WFS Conformance Classes for general implementations

Operations and capabilities for each WFS implementation are described in the standard (1) as follows:

- **Simple WFS**
  "The server shall implement the following operations: GetCapabilities, DescribeFeatureType, ListStoredQueries, DescribeStoredQueries, GetFeature operation with only the StoredQuery action. One stored query, that fetches a feature using its id, shall be available but the server may also offer additional stored queries. Additionally the server shall conform to at least one of the HTTP GET, HTTP POST or SOAP conformance classes."

  NOTE: This WFS implementation supports only limited/restricted access via pre-defined and stored queries. Users are not able or allowed to define their own queries using Filter Encoding (2).

- **Basic WFS**
  "The server shall implement the Simple WFS conformance class and shall additionally implement the GetFeature operation with the Query action and the GetPropertyValue operation."

  NOTE: A Basic WFS supports at least a minimum set of filter sets to create ad hoc queries

- **Transactional WFS**
  "The server shall implement the Basic WFS conformance class and shall also implement the Transaction operation."

  NOTE: A Transactional WFS supports operations to create, update and delete features on the server side.

- **Locking WFS**
  "The server shall implement the Transactional WFS conformance class and shall implement at least one of the GetFeatureWithLock or LockFeature operations."

  NOTE: A Locking WFS supports the locking of features on the server side, for instance while one transaction accesses a data item, no other transaction may modify the same data item. This may be accomplished by using locks that control access to the data.

- **Manage stored queries**

  "The server shall implement the CreateStoredQuery and the DropStoredQuery operations."

  NOTE: A Web Feature Service that supports the management of stored queries via the Web allows clients to create, drop, list and described parameterized query expressions that are stored by the server and can be repeatedly invoked using different parameter values.

Apart from the WFS Service Level conformance classes there are several other conformance classes related to bindings, filter requirements and additional components. These are shown in Figure 2 below. Figure 2 also shows requirements about output formats and multilingual aspects which could be possible requirements or recommendations that need to be defined as part of the DGIWG profile.

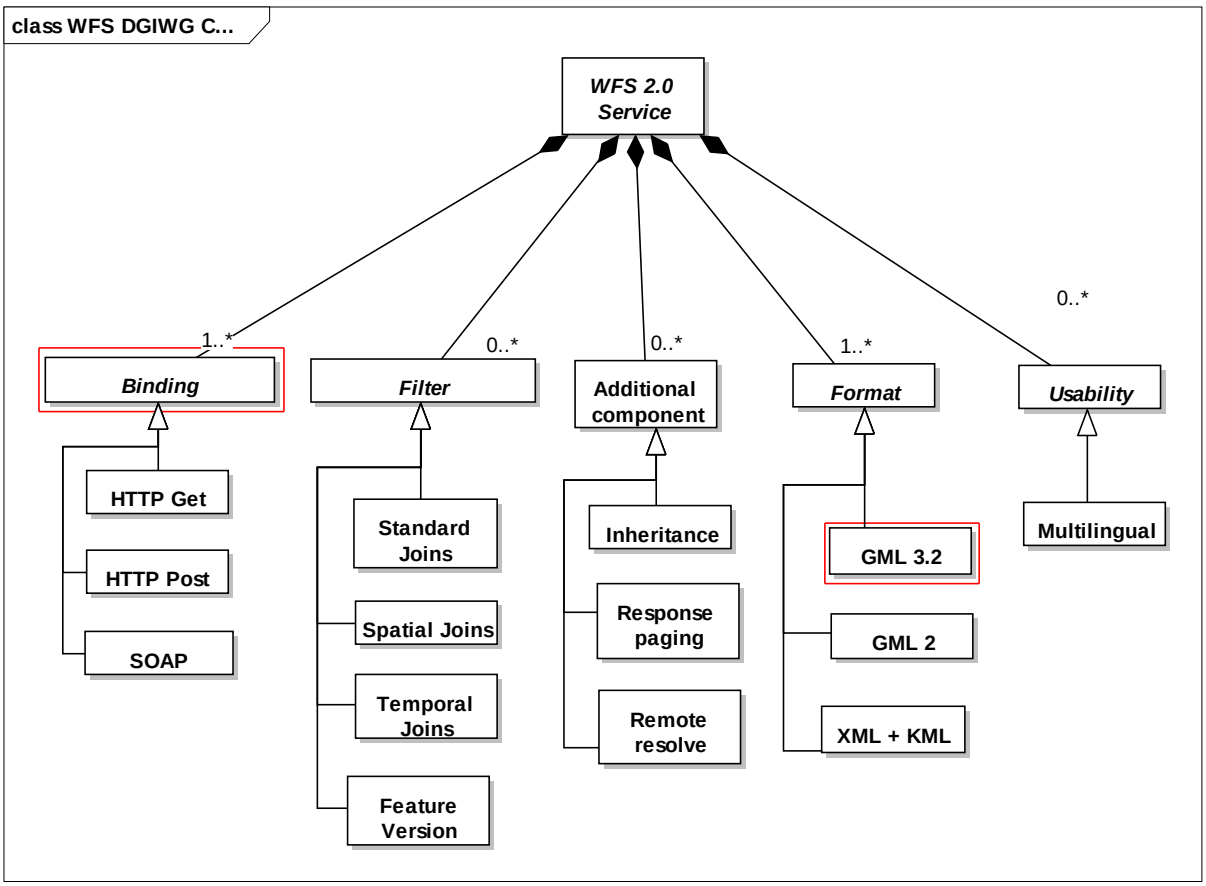

Figure 2: WFS Conformance Classes Overview

The red boxes highlight mandatory requirements, meaning that a WFS 2.0 compliant service must implement at least GML version 3.2 as output format and one of the following bindings listed below:

Use cases for the listed bindings are for example:

- HTTP GET with KVP encoding
  - simple to implement, adapted for low bandwidth (short request)
  - less adapted for complicated set of parameters especially when using filter expressions
- HTTP POST with XML encoding,
  - more adapted for complex requests (with filter expression for example)
- SOAP over HTTP POST
  - Supports the same requests as HTTP POST with XML
  - SOAP provides a mechanism to transmit additional information; these can be used for security and authentication purposes.

The issue of bindings is an important one to consider. If the client and server do not use the same binding, they will not interoperate. A service will define which binding it supports as part of its GetCapabilities document. If a client requests information from the server via a different binding than the server supports, the server typically returns an error message that

would not necessarily be useful to the client. If a WFS server for example provides for a HTTP POST binding and a client makes the request using a SOAP binding an interoperability issue exists. It is possible however, to require a developer to implement a "façade" on the server side which would be able to read a request and interpret it or translate it to a binding it does support to solve such interoperability problems. The other possibility would be to build clients that support more than one binding.

6.2. Filter operators

The DGIWG WFS profiles do require support for Filter operators.

OGC 09-026R2 specifies standard, spatial and temporal filters. These implement the following operators:

- Standard operators support comparison (PropertyIsEqualTo, PropertyIsNotEqualTo, PropertyIsLessThan, PropertyIsGreaterThan, PropertyIsLessThanOrEqualTo, PropertyIsGreaterThanOrEqualTo) and logical operators (AND, OR, NOT) and may implement one or more additional functions.
- Spatial operators are BBOX and Equals, Disjoint, Intersects, Touches, Crosses, Within, Contains, Overlaps
- Temporal operators are After, Before, Begins, BegunBy, TContains, During, TEquals, TOverlaps, Meets, OverlappedBy, MetBy, EndedBy, AnyInteracts

6.3. Joins

The DGIWG WFS profiles do not require the support for Joins. Nevertheless there might use cases where Joins are required. Therefore the support of Joins is optional.

Join queries are categorized as standard, spatial and temporal joins based on the operators that are used in the join predicates (filter expression). Filter operators implemented within standard, temporal and spatial joins are not dependent upon each other, meaning if a service implements spatial joins it is not necessarily true that it supports either standard or temporal joins. A service will define which set of joins it supports as part of its GetCapabilities document. If all the Filter operators, except the spatial and temporal operators are used in a join predicate, the server implements “standard join” queries. This kind of query includes comparison (PropertyIsEqualTo/NotEqualTo, PropertyIsLessThan/GreaterThan, ...) and logical operators (AND, OR, NOT).

Example: The following query expression uses a join to find the spouse of the person whose Identifier is "12345". (OGC 09-025r1 and ISO/DIS 19142 section 7.9.2.5.3)

```
<wfs:Query typeNames="myns:Person myns:Person" aliases="a b">
    <fes:Filter>
        <fes:And>
            <fes:PropertyIsEqualTo>
                <fes:ValueReference>a/Identifier</fes:ValueReference>
                <fes:Literal>12345</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsEqualTo>
                <fes:ValueReference>a/spouse</fes:ValueReference>
                <fes:ValueReference>b/Identifier</fes:ValueReference>
            </fes:PropertyIsEqualTo>
        </fes:And>
    </fes:Filter>
</wfs:Query>
```

In this example a join predicate between "a/spouse" and "b/Identifier" is used to locate the spouse of the person whose identifier is "12345". This is also an example of a self-join since the myns:Person feature type is being joined to itself in order to identify the spouse.

If spatial operators are used in join predicates, the server implements “spatial join” queries. This conformance class requires the support of BBOX operator + at least an additional spatial operator among a list detailed in OGC 09-026R2 (Equals, Disjoint, Intersects, Touches, Overlaps, Contains, …).

Example: The following query expression uses a spatial join to find all park features that contain lakes:

```
<wfs:Query typeNames="myns:Parks myns:Lakes">
    <fes:Filter>
        <fes:Contains>
            <fes:ValueReference>ns1:Parks/geometry</fes:ValueReference>
            <fes:ValueReference>ns1:Lakes/geometry</fes:ValueReference>
        </fes:Contains>
    </fes:Filter>
</wfs:Query>
```

The list of feature types to join is specified using the typeNames attribute (i.e. typeName="myns:Parks myns:Lakes") on the wfs:Query element. The join predicate is specified using the fes:Filter element and finds all pairs of ns1:Park and ns1:Lake features whose geometries satisfy the spatial operator fes:Contains.

If **temporal** operators are used in join predicates, the server implement “**temporal join**” queries. This conformance class requires the support of During operator + at least an additional spatial operator among a list detailed in OGC 09-026R2 (After/Before, Begins, OverlapsBy, EndedBy, TContains, …)

The following table shows the filter operations defined in *OGC 09-026r2 Filter Encoding* in relation to the particular joining capability defined in *OGC 09-025r2 Web Feature Service*. For more information regarding the differences of the particular filters see (4) (5).

Table 3: Links between OGC 09-025r2 conformance classes (WFS) and OGC 09-026r2 conformance classes (FE)

	Conformance classes(OGC 09-025r2 - WFS 2.0)			
OGC 09-026r2 Conformance classes	Standard joins	Spatial joins	Temporal joins	Feature version
Minimum Standard Filter	X			
Standard Filter	X			
Minimum Spatial Filter		X		
Spatial Filter		X		
Minimum Temporal Filter			X	
Temporal Filter			X	
Version navigation				X

6.4. Versioning

The DGIWG WFS profiles do not require the support for versioning. Nevertheless there might use cases where versioning is required. Therefore the support of versioning is optional.

Versioning provides the ability to access versions of one feature or historical reference. Within filter expressions, specific feature instances can be identified. If the server supports versioning, specific versions of a feature can be referenced. The “Feature version” conformance class requires the implementation of ResourceId operator with the parameters that allow versions of resources to be queried (version, startTime, endTime). This allows for navigation between different versions of the same feature.

The version attribute may be an integer (accessing the version of the resource), a date (accessing the version of the resource closest to the specified date) or a string (FIRST, LATEST, PREVIOUS, NEXT and ALL).

The attributes startTime and endTime may be used to specify a predicate that selects all versions of a resource between the specified start date and end date. The startTime and endTime attributes shall always be specified together. If the startTime and endTime are specified, the version attribute shall not be specified.

6.5. Output Format

As shown in Figure 2, a service, claiming compliance with WFS 2.0, shall support at least GML version 3.2. However the operations in the OGC 09-025R2 International Standard are defined in a manner that allows them to work with previous and future versions of GML.

Servers may also support additional non-GML feature encodings that shall also be listed in the server's capabilities document. This may include any other string or MIME types that the server supports. DGIWG WFS 2.0 profiles will not make any recommendations for the support of other output formats like for instance compressed GML (e.g. zip, tar, gzip) at this time. Nevertheless the following example shows a possible excerpt of a GetCapabilities document for possibly supported formats:

```
<ows:Parameter name="outputFormat">
    <ows:Value>application/gml+xml;version=3.2</ows:Value>
    <ows:Value>GML2</ows:Value>
    <ows:Value>GML2-GZIP</ows:Value>
    <ows:Value>SHAPE-ZIP</ows:Value>
    <ows:Value>csv</ows:Value>
    <ows:Value>excel</ows:Value>
    <ows:Value>gml3</ows:Value>
    <ows:Value>json</ows:Value>
    <ows:Value>text/xml; subtype=gml/2.1.2</ows:Value>
</ows:Parameter>
```

Compression formats like GZIP, ZIP and others do an excellent job of compressing XML data. Compression rates of 80% and more have been achieved in the past OGC testbed OWS-3 (see (2)). In the referenced document it was mentioned that the encoding and decoding of GZIP and BZIP2 formats is CPU intensive, so that the “compaction” of the raw information before the “compression” step can have an large impact on overall system performance.

7. DGIWG Basic WFS (normative)

This section is normative and defines the DGIWG WFS Basic conformance class.

7.1. Normative Requirements

The Normative requirements requested by this conformance class are summarized in Table 4.

Table 4: DGIWG Basic WFS Normative Server Requirements

No.	Requirement	Compliance
1	A DGIWG Basic WFS server shall implement the OGC 09-025R2 Basic WFS conformance class.	M
2	A DGIWG Basic WFS server shall include the following information in the abstract element of the service metadata: "This service implements the DGIWG WFS 2.0 profile version 2.0, DGIWG Basic WFS conformance class (http://www.dgiwg.org/std/wfs/2.0/conf/basic)."	M
3	A DGIWG Basic WFS server shall provide a minimum keywords list based on the DGIM (DGIWG Geospatial Information Model) groups.	M
4	If content provided by a WFS server is classified, a DGIWG Basic WFS server shall identify the highest classification level of the content accessible through the WFS service by populating the <ows:AccessConstraints> element.	C
5	A DGIWG Basic WFS server shall provide wfs:FeatureTypeList elements from a successful GetCapabilities request according to Table 54.	M
6	A DGIWG Basic WFS server shall create and store stored queries according to Table 76.	M
7	A DGIWG WFS Basic server shall support OGC 09-026R2 Standard Filter conformance class.	M
8	A DGIWG WFS Basic server shall support OGC 09-026R2 Spatial Filter conformance class.	M
9	In addition, a DGIWG WFS Basic server shall support Equals, Disjoint, Intersects, Touches, Overlaps, Contains, Within, Crosses, DWithin, Beyond	M
10	A DGIWG WFS Basic server shall support OGC 09-026R2 Temporal Filter conformance class.	M
11	In addition, a DGIWG WFS Basic server shall support at least After, Before.	M
12	A DGIWG WFS Basic server shall implement HTTP GET conformance class.	M
13	A DGIWG WFS Basic server shall implement HTTP POST conformance class.	M
20	A service implementing the DGIWG WFS profiles shall provide their data in - CRS:84 WGS84 geographic longitude, then latitude, expressed in decimal degrees - EPSG:4326 WGS84 geographic latitude, then longitude, expressed in decimal degrees - Optionally data may be provided in additional CRSs, depending on national requirements.	M
21	A service implementing the DGIWG WFS profiles shall define constraints on operation as required by Table 12.	M

7.2. Non-Normative Recommendations for Implementation

The non-normative requirements requested by this profile are summarized in Table 5.

Table 5: DGIWG WFS Profile Non-normative Recommendations for DGIWG Basic WFS Server Implementation

No.	Recommendation	Compliance
1	A DGIWG Basic WFS server should provide the updateSequence attribute.	O
2	A DGIWG Basic WFS server should provide following URL designation: <ows:Profile>http://www.dgiwg.org/std/wfs/2.0/conf/basic</ows:Profile>	O
3	A DGIWG WFS Basic server should support OGC 09-025R2 Standard joins	O
4	A DGIWG WFS Basic server should support OGC 09-025R2 Spatial joins.	O
5	A DGIWG WFS Basic server should support OGC 09-025R2 Temporal joins.	O
6	A DGIWG WFS server should implement Response paging conformance class.	O
9	The followings CRSs should be supported as well: - World Mercator projection, known as EPSG:3395. - All projections for which validity zone overlaps data published by the service - UTM projections over WGS84 (north zones), EPSG:32601 to EPSG:32660 - UTM projections over WGS84 (south zones), EPSG:32701 to EPSG:32760 - UPS projection over WGS84 (north zone), EPSG:5041 - UPS projection over WGS84 (south zone), EPSG:5042	0

7.3. Introduction

The DGIWG Basic WFS Profile is based on the OGC 09-025R2 Basic WFS.

Requirement 1: A DGIWG Basic WFS server shall implement the OGC 09-025R2 Basic WFS conformance class.

The relevant operations and parameters for the DGIWG Basic WFS profile are discussed in the following sections.

7.4. Supported operations

Based on the **mandatory** OGC 09-025r2 conformance classes the DGIWG WFS Basic Profile shall implement the following operations: **GetCapabilities, DescribeFeatureType, ListStoredQueries, DescribeStoredQueries, GetFeature, and GetPropertyValue**. The sections below provide additional information/requirements on operations supported.

7.4.1. GetCapabilities

The GetCapabilities operation generates a service metadata document describing a WFS service provided by a server. This metadata document holds information about:

- Service identification, service provider and operation metadata, which lists service operations offered by the server. These elements are inherited from the OWS-Common 1.1 specification.

  Recommendation 1: A DGIWG Basic WFS server should provide the updateSequence attribute.

  Requirement 2: A DGIWG Basic WFS server shall include the following information in the abstract element of the service metadata: "This service implements the DGIWG WFS 2.0 profile version 2.0, DGIWG Basic WFS conformance class (http://www.dgiwg.org/std/wfs/2.0/conf/basic)."

  Recommendation 2: A DGIWG Basic WFS server should provide following URL designation:

  <ows:Profile>http://www.dgiwg.org/std/wfs/2.0/conf/basic</ows:Profile>

  Requirement 3: A DGIWG Basic WFS server shall provide a minimum keywords list based on the DGIM (DGIWG Geospatial Information Model) groups.

  NOTE 1: Annex F presents groups and subgroups for DGIM 2016-2.0.

  NOTE 2: Additional keywords may be added to the list as appropriate to support data discovery.

  Requirement 4: If content provided by a WFS server is classified, a DGIWG Basic WFS server shall identify the highest classification level of the content accessible through the WFS service by populating the <ows:AccessConstraints> element.

  NOTE: In an NATO environment, the common NATO classification scheme has to be applied (see DMF chapter 5.6.1.4 Classification Level Codelist).

- WFS version(s) supported by the server (mandatory "2.0.2" and possibly other versions "1.1.0", "1.0.0"). This information allows for version negotiation between a WFS service and a WFS client.

- FeatureTypes available on the server.

  For each feature type the name is the minimum information required. It is also possible to provide additional optional information like output format, title, keywords or abstracts describing the feature types in more detail (see Table 11 in OGC 09-025r2).

  Table 6 defines further requirements and recommendations (extending OGC 09-025r2) for the description of feature types.

- filterCapabilities supported by the WFS.

  This section of the GetCapabilities document lists filter expressions that are supported by the server and then used for the GetFeature and GetPropertyValue operation. Filters which are supported by the DGIWG Basic WFS profile are presented in section 7.5.

The following table lists the elements that are used to describe each feature type listed within the wfs:FeatureTypeList element in the DGIWG profile compared to the base standard.

Table 6: Elements to describe feature types in a GetCapabilities document

Element name	**O/M[1] OGC 09-025r2**	**O/R/M[1] DGIWG Basic profile**	**Description**
Name	M	M	Name of the feature type
Title	O	M / C	One human readable title is mandatory for all cases. For coalition interoperability an English title is mandatory. NOTE: For national use the title can be provided in the national language.
Abstract	O	R / C	One abstract element is recommended for all cases. If the abstract element is provided in a coalition environment an English abstract is mandatory. NOTE: For national use the abstract can be provided in the national language.
Keywords	O	M	At least one keyword is required for each feature type to facilitate data discovery in catalogues. Keywords can be duplicated in multiple languages. Using DFDD codes and names as keywords are recommended.
DefaultCRS	M	M	The wfs:DefaultCRS element indicates which coordinate reference system shall be used by a WFS. See section 9.2 for more details.
OtherCRS	O	R	See section 9.2 for more details.
NoCRS	O	O	The wfs:NoCRS element shall be used for feature types that have no spatial properties.
OutputFormats	O	O	If this optional element is not specified, then all the result formats listed for the GetFeature operation are assumed to be supported.
WGS84BoundingBox	O	M	This knowledge aids client applications by letting them know where they should query in order to have a high probability of finding feature data.
MetadataURL	O	R	When MetadataURL is used, the xlink:href element shall be used to reference any metadata. In an SDI like architecture, a link to the metadata resource available on a CSW server should be provided.
ExtendedDescription	O	O	A WFS may add elements to the description of a feature type, without having to redefine the capabilities schema, using the wfs:ExtendedDescription element.
[1] O = Optional, R = Recommended, M = Mandatory, C = Conditional			

Requirement 5: A DGIWG Basic WFS server shall provide wfs:FeatureTypeList elements from a successful GetCapabilities request according to Table 6.

7.4.2. DescribeFeatureType

The DescribeFeatureType operation returns a schema description of feature types offered by a WFS instance. The schema descriptions define how a WFS expects feature instances to be encoded on output (in response to a GetPropertyValue, GetFeature or GetFeatureWithLock operation). These descriptions are at a minimum provided as GML 3.2 application schemas. Other versions of GML may be supported.

Table 7: DescribeFeatureType

Element name	O/M[1] OGC 09-025r2	O/R/M[1] DGIWG Basic profile	Description
TYPENAME	O	O	A comma separated list of feature types to describe. If no value is specified, the complete application schema offered by the server shall be described.
OUTPUTFORMAT	O	O	Shall support the value "application/gml+xml; version=3.2" indicating that a GML (see ISO19136:2007) application schema shall be generated. A server may support other values to which this International Standard does not assign any meaning.
[1] O = Optiona, R = Recommended, M = Mandatory			

The DGIWG WFS Basic profile does not define any further requirements for the DescribeFeatureType operation.

7.4.3. GetFeature

The GetFeature operation returns a selection of features from a data store. The GetFeature operation returns a selection of features that satisfies the query expressions specified in the request.

7.4.4. GetPropertyValue

The GetPropertyValue operation allows the value of a feature property or part of the value of a complex feature property to be retrieved from the data store for a set of features identified using a query expression.

7.4.5. Stored queries

To be compliant with the DGIWG Basic profile (Requirement 1), servers support two operations for stored queries (ListStoredQueries, DescribeStoredQueries) and implement the GetFeatureById stored query.

NOTE: The management of stored queries, i.e. the implementation of CreateStoredQuery and the DropStoredQuery operations, is optional.

7.4.5.1. Stored queries definition

Servers can be pre-configured to offer any number of stored queries implemented in any number of ways. However the DGIWG requirements mandates stored queries follow the rules defined by Table 8.

Requirement 6: A DGIWG Basic WFS server shall create and store stored queries according to Table 8.

Table 8: StoredQuery definition

Element name		**O/M[1] OGC 09-025r2**	**O/R/M[1] DGIWG Basic profile**	**Description**
Id		M	M	Unique identifier that can be used to repeatedly invoke a stored query
Title		O	M / C	Human-readable name to a stored query. One human readable title is mandatory. If the Title element is provided in a coalition environment an English abstract is mandatory. NOTE: For national use the abstract can be provided in the national language.
Abstract		O	R/C	Descriptive human-readable to a stored query. If the abstract element is provided in a coalition environment an English abstract is mandatory. NOTE: For national use the abstract can be provided in the national language.
Metadata		O	O	Inline or reference more detailed metadata about a stored query (from OWS-Common 1.1)
Parameter[2]		O	O	Argument of a stored query
	Name	M	M	Name of each argument
	Type	M	M	Type of each argument
	Title	O	M	Human-readable name to an argument
	Abstract	O	R	Descriptive human-readable to an argument
	Metadata	O	O	Inline or reference more detailed metadata about an argument
QueryExpressionText		M	M	The wfs:QueryExpressionText element shall be used to enumerate one or more component query expressions that a stored query executes when invoked.
	returnFeatureTypes	M	M	Specifies the feature type(s) that each component query expression returns.

Element name		O/M[1] OGC 09-025r2	O/R/M[1] DGIWG Basic profile	Description
	language	M	M	Specifies the implementation language of a component query expression. Servers shall support the value "urn:ogc:def:queryLanguage:OGC-WFS::WFSQueryExpression". Optionally, the server may support languages such as SQL, XQuery, XPath, SPARQL, and Java.
	isPrivate	O	O	Private means that the implementation text is visible only to the creator of the stored query.
[1] O = Optional, R = Recommended, M = Mandatory, C = Conditional [2]NOTE: when Parameter exists name, type, Title are mandatory.				

The QueryExpressionText element allows to enumerate one or more component query expressions that a stored query executes when invoked. Each QueryExpressionText element may contain a wfs:Query element, or a wfs:StoredQuery element or some other implementation specific content for specifying a component query expression in another implementation language.

7.4.5.2. ListStoredQueries

The ListStoredQueries operation lists all the stored queries available at a server. For each stored query returned, Id, Title and a list of returnFeaturetype are provided.

7.4.5.3. DescribeStoredQueries

The DescribeStoredQueries operation provides detailed information about stored queries that a server offers. This is achieved through the use of StoredQueryId within the request.

7.5. Supported filters

7.5.1. Standard filter

Implements all the comparison and logical operators and may implement one or more additional functions. The DGIWG profile requires the support of all the comparison and logical operators and does not exclude the use of additional functions. If a WFS server provides additional functions then it must follow the requirements defined in OGC 09-026R2.

Requirement 7: A DGIWG WFS Basic server shall support OGC 09-026R2 Standard Filter conformance class.

7.5.2. Spatial filter

The OGC 09-026R2 standard defines spatial joins as implementation of the BBOX operator and at least an additional spatial operator (Equals, Disjoint, Intersects, Touches, Overlaps, Contains, Within, Crosses, DWithin, Beyond).

The DGIWG WFS Basic profile requires the support of all of these spatial operators.

Requirement 8: A DGIWG WFS Basic server shall support OGC 09-026R2 Spatial Filter conformance class.

Requirement 9: In addition, a DGIWG WFS Basic server shall support Equals, Disjoint, Intersects, Touches, Overlaps, Contains, Within, Crosses, DWithin, Beyond.

7.5.3. Temporal filter

The OGC 09-026R2 standard defines **temporal** joins as implementation of the During operator and at least an additional temporal operator among a list detailed in OGC 09-026R2 (After, Before, Begins, BegunBy, TContains, TEquals, TOverlaps, Meets, OverlappedBy, MetBy, EndedBy, AnyInteracts).

Requirement 10: A DGIWG WFS Basic server shall support OGC 09-026R2 Temporal Filter conformance class.

Requirement 11: In addition, a DGIWG WFS Basic server shall support at least After, Before.

7.6. Joins

A server that claims conformance with the DGIWG WFS Basic profile should implement the following OGC 09-025r2 conformance classes.

- OGC 09-025R2 Standard joins
- OGC 09-025R2 Spatial joins
- OGC 09-025R2 Temporal joins

The requirement for joins is strongly dependent on the underlying architecture of the WFS. If the WFS server is set up on a spatial database sophisticated joins are possible. The WFS request will be mapped to a SQL statement, which will then be processed by the database. Joining of features from same or different feature-types requires server side processing and caching.

If the WFS server operates on a file based structure the possibility of joins is much more complicated. Additionally joins are currently not widely supported by WFS implementations.

Therefore the requirements for joins are conditional and dependent upon the underlying database structure.

NOTE: The standard, spatial and temporal operators to select specific features from one feature-type still work, only the capability to do joins between different feature types are limited by this change.

Standard joins allow executing a query that fetches data from at least two feature types (this is also true for self-joins with aliases) and includes a join predicate that does not involve spatial or temporal operators. The response generates a valid response using the wfs:Tuple element to contain each feature tuple

Recommendation 3: A DGIWG WFS Basic server should support OGC 09-025R2 Standard joins

Spatial joins allow executing a query that fetches data from at least two feature types (this is also true for self-joins with aliases) and includes a join predicate that uses a spatial operator. The response generates a valid response using the wfs:Tuple element to contain each feature tuple.

Recommendation 4: A DGIWG WFS Basic server should support OGC 09-025R2 Spatial joins.

Temporal joins allow for executing a query that fetches data from at least two feature types (this is also true for self-joins with aliases) and includes a join predicate that uses a temporal operator. The response generates a valid response using the wfs:Tuple element to contain each feature tuple.

Recommendation 5: A DGIWG WFS Basic server should support OGC 09-025R2 Temporal joins.

The following example shows a GetFeature request that uses a self-join with aliases to find all road segments that cross within some specified area of interest.

```
<GetFeature service="WFS" version="2.0.2" xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0/wfs.xsd http://www.opengis.net/gml/3.2
http://schemas.opengis.net/gml/3.2.1/gml.xsd">
    <Query typeNames="myns:RoadSegments myns:RoadSegments" aliases="RS1 RS2">
        <fes:Filter>
            <fes:And>
                <fes:BBOX>
                    <fes:ValueReference>/RS1/geometry</fes:ValueReference>
                    <gml:Envelope srsName="urn:ogc:def:crs:EPSG::1234">
                        <gml:lowerCorner>10 10</gml:lowerCorner>
                        <gml:upperCorner>20 20</gml:upperCorner>
                    </gml:Envelope>
                </fes:BBOX>
                <fes:BBOX>
                    <fes:ValueReference>/RS2/geometry</fes:ValueReference>
                    <gml:Envelope srsName="urn:ogc:def:crs:EPSG::1234">
                        <gml:lowerCorner>10 10</gml:lowerCorner>
                        <gml:upperCorner>20 20</gml:upperCorner>
                    </gml:Envelope>
                </fes:BBOX>
                <fes:Crosses>
                    <fes:ValueReference>/RS1/geometry</fes:ValueReference>
                    <fes:ValueReference>/RS2/geometry</fes:ValueReference>
                </fes:Crosses>
            </fes:And>
        </fes:Filter>
    </Query>
</GetFeature>
```

7.7. Bindings

Conformance classes HTTP GET and HTTP POST require the support of requests according to these two bindings.

Requirement 12: A DGIWG WFS Basic server shall implement HTTP GET conformance class.

Requirement 13: A DGIWG WFS Basic server shall implement HTTP POST conformance class.

According to the previous binding requirements, a service that claims to be compliant with the DGIWG WFS Basic profile shall implement the following bindings for WFS operations.

Table 9: Supported bindings for DGIWG WFS Basic profile

Operation	Request Encoding
GetCapabilities	XML & KVP
DescribeFeatureType	XML & KVP
GetPropertyValue	XML & KVP
GetFeature	XML & KVP
ListStoredQueries	XML & KVP
DescribeStoredQueries	XML & KVP

7.8. Response paging

Response paging is the ability of a client to scroll through a set of response features or values, N-features or values at-a-time much like one scrolls through the response from a search engine one page at a time. For example, a server that supports response paging responds with a wfs:FeatureCollection element containing the first 100 records in the result set. The next attribute is set so that the client can retrieve the next 100 features. This can continue until all features are retrieved. Response paging is likely to provide for faster response time and more consistent client functions.

However, response paging is currently not widely supported by implementations of WFS. Therefore DGIWG nations should consider the benefit of response paging versus the additional implementation cost.

Recommendation 6: A DGIWG WFS server should implement Response paging conformance class.

8. DGIWG Locking (Transactional) WFS Profile (normative)

This section is normative and defines the DGIWG WFS Locking (Transactional) conformance class.

8.1. Normative Requirements

The Normative requirements requested by this conformance class are summarized in Table 4.

Table 10: DGIWG Locking (Transactional) WFS Normative Server Requirements

No.	Requirement	Compliance
14	A DGIWG Locking (Transactional) WFS server shall implement the OGC 09-025R2 Locking WFS conformance class.	M
15	A DGIWG Locking (Transactional) WFS server shall satisfy following requirements: Requirement 1, Requirement 3, Requirement 4, Requirement 5 and Requirement 6.	M
16	A DGIWG Locking (Transactional) WFS server shall include the following information in the abstract element of the service metadata: "This service implements the DGIWG WFS 2.0 profile version 2.0, DGIWG Locking (Transactional) WFS conformance class (http://www.dgiwg.org/std/wfs/2.0/conf/locking).	M
17	A DGIWG Locking (Transactional) WFS server shall implement at least the LockFeature operation.	M
18	A DGIWG Locking (Transactional) WFS server shall satisfy the requirements in 7.5 (requirements 7 to requirement 11).	M
19	A WFS server that claims to be conformant with the DGIWG Locking (Transactional) WFS profile shall satisfy the requirements in 7.7 (requirements 12 and 13).	M
20	A service implementing the DGIWG WFS profiles shall provide their data in - CRS:84 WGS84 geographic longitude, then latitude, expressed in decimal degrees - EPSG:4326 WGS84 geographic latitude, then longitude, expressed in decimal degrees - Optionally data may be provided in additional CRSs, depending on national requirements.	M
21	A service implementing the DGIWG WFS profiles shall define constraints on operation as required by Table 12.	M

8.2. Non-Normative Recommendations for Implementation

The non-normative requirements requested by this profile are summarized in Table 5.

Table 11: DGIWG WFS Profile Non-normative Recommendations for DGIWG Locking (Transactional) WFS Server Implementation

No.	Recommendation	Compliance
7	A DGIWG Locking (Transactional) WFS server should provide following URL designation: <ows:Profile>http://www.dgiwg.org/std/wfs/2.0/conf/locking</ows:Profile>	O
8	A DGIWG WFS Locking server should implement Response paging conformance class.	O
9	The followings CRSs should be supported as well: - World Mercator projection, known as EPSG:3395. - All projections for which validity zone overlaps data published by the service - UTM projections over WGS84 (north zones), EPSG:32601 to EPSG:32660 - UTM projections over WGS84 (south zones), EPSG:32701 to EPSG:32760 - UPS projection over WGS84 (north zone), EPSG:5041 - UPS projection over WGS84 (south zone), EPSG:5042	O

8.3. Introduction

The DGIWG Locking (Transactional) WFS conformance class is based on the DGIWG Basic WFS conformance class and also implements transactional and locking functionalities.

The DGIWG Locking (Transactional) WFS is based on the OGC 09-025R2 Locking WFS conformance class.

Requirement 14: A DGIWG Locking (Transactional) WFS server shall implement the OGC 09-025R2 Locking WFS conformance class.

NOTE: The Locking WFS conformance class contains all capabilities of the Basic WFS.

In the following the relevant operations and parameters for the DGIWG Transactional (Locking) WFS profile are discussed.

8.4. Supported operations

The DGIWG WFS 2.0 Locking (Transactional) profile requires the following operations: GetCapabilities, DescribeFeatureType, ListStoredQueries, DescribeStoredQueries, GetFeature, GetPropertyValue, Transaction and at least one of the GetFeatureWithLock or LockFeature operations. The sections below provide additional information on operations supported.

8.4.1. Common operations

All operations 7.4.1 through 7.4.5.3 are applicable to DGIWG WFS Locking (Transactional) profile.

Requirement 15: A DGIWG Locking (Transactional) WFS server shall satisfy following requirements: Requirement 1, Requirement 3, Requirement 4, Requirement 5 and Requirement 6.

NOTE: Requirement 2 in DGIWG Basic WFS is adpated here in requirement 16 for DGIWG Locking WFS.

Requirement 16: A DGIWG Locking (Transactional) WFS server shall include the following information in the abstract element of the service metadata: "This service implements the DGIWG WFS 2.0 profile version 2.0, DGIWG Locking (Transactional) WFS conformance class (http://www.dgiwg.org/std/wfs/2.0/conf/locking).

Recommendation 7: A DGIWG Locking (Transactional) WFS server should provide following URL designation:

<ows:Profile>http://www.dgiwg.org/std/wfs/2.0/conf/locking</ows:Profile>

8.4.2. Transaction

Using the Transaction operation clients can create, modify, replace and delete features in the Web Feature Service's data store.

8.4.3. Locking operations

Locking can be supported by two types of servers:

- Servers that support automatic data locking, i.e. the transaction operation automatically locks data in order to maintain consistency thus alleviating the client from having to use the LockFeature or GetFeatureWithLock operations to lock the features to be modified. This is advertised with AutomaticDataLocking constraint = TRUE.
- Servers that do not support automatic data locking are required to lock features prior to operating on them. Locking can be done through LockFeature or GetFeatureWithLock operations.

It is recommended to implement the DGIWG Locking (Transactional) profile with automatic data locking servers.

Requirement 17: A DGIWG Locking (Transactional) WFS server shall implement at least the LockFeature operation.

8.5. Supported filters

A server that claims conformance with the DGIWG WFS Locking profile does not have any further requirement than for the DGIWG WFS Basic profile.

Requirement 18: A DGIWG Locking (Transactional) WFS server shall satisfy the requirements in 7.5 (requirements 7 to requirement 11).

8.6. Bindings

A server that claims conformance with the DGIWG WFS Locking (Transactional) profile does not have any further requirement than for the DGIWG WFS Basic profile. It then implements HTTP GET and HTTP POST.

Requirement 19: A WFS server that claims to be conformant with the DGIWG Locking (Transactional) WFS profile shall satisfy the requirements in 7.7 (Requirement 12 and Requirement 13).

According to the previous binding requirements, a service that claims to be compliant with the DGIWG WFS Locking profile shall implement the following bindings for WFS operations.

Operation	Request Encoding
GetCapabilities	XML & KVP
DescribeFeatureType	XML & KVP
GetPropertyValue	XML & KVP
GetFeature	XML & KVP
ListStoredQueries	XML & KVP
DescribeStoredQueries	XML & KVP
Transaction	XML
LockFeature	XML & KVP
GetFeatureWithLock[1]	XML & KVP

[1]NOTE GetFeatureWithLock operation is optional.

8.7. Response paging

A server that claims conformance with the DGIWG WFS Locking (Transactional) profile does not have any further requirement than for the DGIWG WFS Basic profile. It then implements Response Paging.

Recommendation 8: A DGIWG WFS Locking server should implement Response paging conformance class.

9. Common requirements (normative)

9.1. Output formats

OGC 09-025R2 requires GML 3.2 as output format for DescribeFeatureType, GetFeature (GetFeatureWIthLock) and GetPropertyValue operations. This requirement is just a minimum; this means that additional output formats may be supported by the server, e.g. previous version of GML, KML or Shape.

9.2. CRS

There is no requirement to support a specific CRS in OGC 09-025R2. But within the goal to be more interoperable (more interoperable between WFS servers and clients and more interoperable between the different types of OGC/DGIWG services), a service implementing the DGIWG WFS profiles shall provide their data at a minimum in the following CRSs

Requirement 20: A service implementing the DGIWG WFS profiles shall provide their data in

- ***CRS:84 WGS84 geographic longitude, then latitude, expressed in decimal degrees***
- ***EPSG:4326 WGS84 geographic latitude, then longitude, expressed in decimal degrees***
- ***Optionally data may be provided in additional CRSs, depending on national requirements.***

If the internal storage CRS is different from EPSG:4326, then the WFS shall support a transformation between the internal storage CRS and EPSG:4326. The effects of such a transformation shall be considered when determining and declaring the guaranteed data accuracy.

Recommendation 9: The followings CRSs should be supported as well:

- ***World Mercator projection, known as EPSG:3395.***
- ***All projections for which validity zone overlaps data published by the service***
  - ***UTM projections over WGS84 (north zones), EPSG:32601 to EPSG:32660***
  - ***UTM projections over WGS84 (south zones), EPSG:32701 to EPSG:32760***
  - ***UPS projection over WGS84 (north zone), EPSG:5041***
  - ***UPS projection over WGS84 (south zone), EPSG:5042***

NOTE *The CRSs above are mandatory in the DGWIG WMS 1.3 profile.*

The supported CRS can be advertised through different sections/elements in the GetCapabilities document:

- through the operation metadata section
- through the supportedCRS element (see 7.4.1) declared for each feature type.

NOTE: *OGC Change Request 11-152 to WFS 2.0.2 allows for the* following schema changes of the form:

Before: srsName="urn:ogc:def:crs:EPSG::4326"

After: srsName="http://www.opengis.net/def/crs/epsg/0/4326"

This CR specified is related to the OGC Policy for moving from :urn" specifications to http uri (URL) specifications for coordinate reference systems.

9.3. Operation constraints

According to OGC 09-025r2 (Table 14), a server may optionally specify one or more of the constraints defined in its capabilities document.

Table 12 below defines the capabilities document requirements for DGIWG as well as provides recommendations on default values constraints.

Requirement 21: A service implementing the DGIWG WFS profiles shall define constraints on operation as required by Table 12.

Table 12: Operation constraints
source (OGC 09-025r2- Table 14 — Operation Constraint)

Constraint Name	Possible Values and/or Value Types	Default Value OGC 09-025R2	DGIWG M/R/O	DGIWG values constraints	WFS Operation	Description
AutomaticDataLocking	Boolean value; either "TRUE" or "FALSE"	FALSE	M	TRUE (recommended)	Transaction	Indicates that the transaction operation automatically locks data in order to maintain consistency thus alleviating the client from having to use the LockFeature or GetFeatureWithLock operations to lock the features to be modified.
PreservesSiblingOrder	Boolean value; either "TRUE" or "FALSE".	FALSE	O		Transaction	Specifies whether the server preserves sibling order for properties with cardinality greater than 1. If the value is true, the server shall preserve sibling order. Otherwise sibling order is not guaranteed to be preserved.
PagingIsTransactionSafe	Boolean value; either "TRUE" or "FALSE"	FALSE	O	FALSE (recommended)	GetFeature GetFeature WithLock GetProperty Value	Specifies whether the server maintains transactional consistency between paging iterations. Servers that declare transactional consistency shall maintain transactional consistency between paging iterations. Thus, the view

Constraint Name	Possible Values and/or Value Types	Default Value OGC 09-025R2	DGIWG M/R/O	DGIWG values constraints	WFS Operation	Description
						of the data that a client sees while paging through the response set shall be consistent with respect to the time that the originating request was executed. See NOTE 2, below the table.
CountDefault	Integer value greater than or equal to zero.		M		GetFeature GetFeature WithLock GetProperty Value	Specifies the default value for the count parameter. Servers shall advertise a value for CountDefault as a means of self-defence, so that a request may not clog a server
ResolveTimeoutDefault	Integer greater than zero.		M	300s (default)	GetFeature GetFeature WithLock GetProperty Value	Defines the maximum number of seconds a server shall wait before receiving a response while resolving resource references. If the constraint is not specified the server shall wait indefinitely in order to resolves a remote reference.
SortLevelLimit	Integer value greater than zero.		O		GetFeature GetFeature WithLock	The SortLevelLimit constraint defines the maximum number of properties that may be simultaneously sorted. In the event that a request contains too many fes:SortProperty elements for a particular service (i.e.

Constraint Name	Possible Values and/or Value Types	Default Value OGC 09-025R2	DGIWG M/R/O	DGIWG values constraints	WFS Operation	Description
						exceeds the SortLevelLimit constraint), the service shall respond with an exception as specified in 7.5. If the constraint is not specified then there is no limit to the number of sort properties that may be specified.
ResolveLocalScope	Integer greater than zero OR the character "*".		M	1..3(min/max) (recommended)	GetFeature GetFeature WithLock GetProperty Value	Defines the minimum and maximum number of levels, when resolving references to resources that are part of the server's local data store. The value * means to as many as all levels. If the constraint is not specified the default value of "*" shall be assumed.
ResolveRemoteScope	Integer greater than zero OR the character "*"	*	M (if Remote resolve supported)	1..3(min/max) (recommended)	GetFeature GetFeature WithLock GetProperty Value	Defines the minimum and maximum number of levels, when resolving remote references. The value * means to as many as all levels. The default value of "*" shall be assumed.
ResponseCacheTimeout	Integer greater than zero.	*	O		GetFeature GetFeature WithLock GetProperty	Define the length of time in seconds that responses shall be cached to support paging (see 7.7.4.4).

Constraint Name	Possible Values and/or Value Types	Default Value OGC 09-025R2	DGIWG M/R/O	DGIWG values constraints	WFS Operation	Description
					Value	If the constraint is not specified then the response cache never times out.
QueryExpressions	QName; one of wfs:Query of wfs:StoredQuery		O		GetFeature GetFeature WithLock GetProperty Value LockFeature	The names of the supported query expression elements.

NOTE 1: The constraints may be specified on the indicated operation. If more than one operation is listed, the constraint may be specified on each operation individually (perhaps with different values for each operation) or at the service level indicating that the constraint applies for all listed operations.

NOTE 2: Transaction safety mechanism can require a lot of resources on the server side, because all requested data are cached at the time of the request until all these data being distributed to the client. It prevents from inconsistencies between data received by the client within different pages. A use case for supporting this mechanism is when you have a GetFeature request with paging and a transaction on the same features (or linked features) at the same time. For example if a user requests a “powerline” feature and the server executes an update transaction on pylons at the same time, this could bring some inconsistencies (missing, pylons, wrong references, …) in the result downloaded dataset.

9.4. Remote resolve

Resolve is the retrieval of a referenced resource and its insertion into a server-generated response document. The insertion may be accomplished by either replacing the reference in-line with a copy of the resource or by relocating the reference to point to a copy of the resource that has been placed in the response document.

Local resolve allows retrieving only features/properties that are stored in the same data base via their reference.

To be consistent with the requirements stated in OGC 09-025R2, servers that are compliant with the DGIWG WFS profiles shall implement the mandatory ability to resolve local resource references.

Support for resolving remote resources is not recommended as at this time there is limited commercial software support and therefore optional.

9.5. Inheritance

When the feature content that a WFS offers is encoded using XML, a WFS shall use XPath (see W3C XML Path Language) expressions for referencing the properties and components of the value of properties of a feature. The minimum mandatory subset of XPath that servers shall support is described in (1).

A WFS server can support additional XPath operators and/or functions such as the schema-element(). The support of this function is defined in the **inheritance** conformance class. If the schema-element() XPath function (see (1)- XPath schema-element()) is implemented, it can be used to trigger a sequence of queries on a specified resource type and any resource type whose object elements are in the substitution group of this specified resource type.

Example (1):

Consider the examples XPath expression:

/Person/mailAddress/USAddress/streetName

Person is the feature type name, mailAddress is a property of the type Person, USAddress is the property type of the mailAddress, streetName is a property of the Address type. Now consider the XPath expression:

/Person/mailAddress/schema-element(Address)/streetName

In this case, the WFS shall consider the property streetName of all subtypes of Address, i.e. USAddress, EUAddress, AUAddress.

While helpful for specific use cases, the DGIWG WFS profiles do not require support for the Inheritance conformance class.

Annex A
Abstract Test Suite
(Normative)

A.1 Conformance classes

A.1.1 DGIWG Basic WFS

a) Test Purpose: Verify that the server implements the DGIWG Basic WFS profile (Requirement 1, Requirement 7, Requirement 8, Requirement 10, Requirement 12 and Requirement 13).

b) Test Method: Verify that the server implements the followings conformance classes from OGC 09-025r2: Basic, HTTP Get, HTTP Post, Standard Filter, Spatial Filter and Temporal Filter. Verify A.2 conformance tests.

c) References: Clauses 6 and 8

d) Test Type: Capability

A.1.2 DGIWG Locking WFS

a) Test Purpose: Verify that the server implements the DGIWG Locking (Transactional) WFS profile.

b) Test Method: Verify that the server implements the followings conformance classes from OGC 09-025r2: Locking, HTTP Get, HTTP Post, Standard Filter, Spatial Filter and Temporal Filter. Verify A.2.2, A.2.3, A.2.4, A.2.5, A.3 conformance tests.

c) References: Clauses 7 and 8

d) Test Type: Capability

A.2 Basic tests for DGIWG profiles

This section describes additional basic tests for DGIWG profiles.

A.2.1 GetCapabilities for DGIWG Basic WFS

a) Test Purpose: Verify that the server implements the DGIWG additional requirements (Requirement 2 and Requirement 5).

b) Test Method: verify that the XML response to a valid GetCapabilities request contains required elements as in Table 6.

c) References:

- Requirement 2: A DGIWG Basic WFS server shall include the following information in the abstract element of the service metadata: "This service implements the DGIWG WFS 2.0 profile version 2.0, DGIWG Basic WFS conformance class (http://www.dgiwg.org/std/wfs/2.0/conf/basic)."

- Requirement 5: A DGIWG Basic WFS server shall provide wfs:FeatureTypeList elements from a successful GetCapabilities request according to Table 6.

d) Test Type: Capability

A.2.2 Stored queries

a) Test Purpose: Verify that the server implements the DGIWG additional Requirement 6.

b) Test Method: verify that all stored queries available on the server (described by a successful DescribeStoredQueries request) contain elements required by Table 8.

c) References: Requirement 6.

d) Test Type: Capability

A.2.3 Filters

a) Test Purpose: Verify that the server implements the DGIWG additional requirements for filters (Requirement 9 and Requirement 11).

b) Test Method: verify that all standard, spatial and temporal operators required are supported by the server. They shall be advertised in the GetCapabilities document and shall generate a successful response when used with a GetFeature request.

c) References: Requirement 9 and Requirement 11.

d) Test Type: Capability

A.2.4 CRS

a) Test Purpose: Verify that the server implements the DGIWG additional Requirement 20.

b) Test Method: verify that CRS EPSG:4326 is advertised for all available data in the XML response (Capabilities document) to a valid GetCapabilities request. Check that data in a successful GetFeature response with EPSG:4326 are well georeferenced.

c) References: Requirement 20.

d) Test Type: Capability

A.2.5 Operations constraints

a) Test Purpose: Verify that the server implements the DGIWG additional Requirement 21.

b) Test Method: verify that a WFS server advertises and satisfies all constraints listed in Table 12.

c) References: Requirement 21

d) Test Type: Capability

A.3 Locking tests for DGIWG profiles

A.3.1 GetCapabilities for DGIWG Locking WFS

a) Test Purpose: Verify that the server implements the DGIWG additional requirements (Requirement 16 and Requirement 5).

b) Test Method: verify that the XML response to a valid GetCapabilities request contains required elements as in Table 6

c) References:

- Requirement 16: A DGIWG Locking (Transactional) WFS server shall include the following information in the abstract element of the service metadata: "This service implements the DGIWG WFS 2.0 profile version 2.0, DGIWG Locking (Transactional) WFS conformance class (http://www.dgiwg.org/std/wfs/2.0/conf/locking).
- Requirement 5: A DGIWG Basic WFS server shall provide wfs:FeatureTypeList elements from a successful GetCapabilities request according to Table 6.

d) Test Type: Capability

Annex B
Recommended practices
(Informative)

The following OGC publicly available engineering reports give an overview of WFS practices:

- OGC® Testbed 10 CCI Profile Interoperability Engineering Report (OGC 14-021r2)
- OWS 9: OWS-9_Aviation_Architecture_ER_Final (OGC 12-147)
- OWS-9 Engineering Report - CCI - Single Point of Entry Global Gazetteer (OGC 12-104r1)
- OWS-8 Aviation: Guidance for Retrieving AIXM 5.1 data via an OGC WFS 2.0 (OGC 11-073r2)
- OWS-8 Aviation Thread - Authoritative AIXM Data Source Engineering Report (OGC 11-086r1)

Annex C
Examples
(Informative)

C.1 GetCapabilities examples

This GetCapabilities response document contains required parameters for DGIWG WFS Basic profile.

```
<WFS_Capabilities version="2.0.2" xmlns="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:fes="http://www.opengis.net/fes/2.0"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0/wfs.xsd http://www.opengis.net/ows/1.1
http://schemas.opengis.net/ows/1.1.0/owsAll.xsd">
    <ows:ServiceIdentification>
        <ows:Title xml:lang="fr">Profil DGIWG WFS</ows:Title>
        <ows:Title xml:lang="en">DGIWG Profile WFS</ows:Title>
        <ows:Abstract xml:lang="fr">Ce service web prÃ©sente un exemple d'implÃ©mentation
du profil basic DGIWG WFS 2.0</ows:Abstract>
        <ows:Abstract xml:lang="en">This webService is an implementation of the DGIWG WFS
2.0 Basic Profile. </ows:Abstract>
        <ows:Keywords>
            <ows:Keyword>DGIWG</ows:Keyword>
            <ows:Keyword>WFS</ows:Keyword>
            <ows:Keyword>Cropland</ows:Keyword>
            <ows:Keyword>Hedgerow</ows:Keyword>
            <ows:Keyword>Lake/pound</ows:Keyword>
            <ows:Type>String</ows:Type>
        </ows:Keywords>
        <ows:ServiceType>WFS</ows:ServiceType>
        <ows:ServiceTypeVersion>2.0.2</ows:ServiceTypeVersion>
    </ows:ServiceIdentification>
    <ows:ServiceProvider>
        <ows:ProviderName>BlueOx Inc.</ows:ProviderName>
        <ows:ServiceContact>
            <ows:IndividualName>Paul Bunyon</ows:IndividualName>
            <ows:PositionName>Mythology Manager</ows:PositionName>
            <ows:ContactInfo>
                <ows:Address>

    <ows:ElectronicMailAddress>Paul.Bunyon@BlueOx.org</ows:ElectronicMailAddress>
                </ows:Address>
                <ows:OnlineResource xlink:href="http://www.BlueOx.org/contactUs"/>
            </ows:ContactInfo>
            <ows:Role>PointOfContact</ows:Role>
        </ows:ServiceContact>
    </ows:ServiceProvider>
    <ows:OperationsMetadata>
        <ows:Operation name="GetCapabilities">
```

```
            <ows:DCP>
                <ows:HTTP>
                    <ows:Get xlink:href="http://www.BlueOx.org/wfs/wfs.cgi?"/>
                    <ows:Post xlink:href="http://www.BlueOx.org/wfs/wfs.cgi"/>
                </ows:HTTP>
            </ows:DCP>
            <ows:Parameter name="AcceptVersions">
                <ows:AllowedValues>
                    <ows:Value>2.0.0</ows:Value>
                </ows:AllowedValues>
            </ows:Parameter>
        </ows:Operation>
        <ows:Operation name="DescribeFeatureType">
            <ows:DCP>
                <ows:HTTP>
                    <ows:Get xlink:href="http://www.BlueOx.org/wfs/wfs.cgi?"/>
                    <ows:Post xlink:href="http://www.BlueOx.org/wfs/wfs.cgi"/>
                </ows:HTTP>
            </ows:DCP>
        </ows:Operation>
        <ows:Operation name="ListStoredQueries">
            <ows:DCP>
                <ows:HTTP>
                    <ows:Get xlink:href="http://www.BlueOx.org/wfs/wfs.cgi?"/>
                    <ows:Post xlink:href="http://www.BlueOx.org/wfs/wfs.cgi"/>
                </ows:HTTP>
            </ows:DCP>
        </ows:Operation>
        <ows:Operation name="DescribeStoredQueries">
            <ows:DCP>
                <ows:HTTP>
                    <ows:Get xlink:href="http://www.BlueOx.org/wfs/wfs.cgi?"/>
                    <ows:Post xlink:href="http://www.BlueOx.org/wfs/wfs.cgi"/>
                </ows:HTTP>
            </ows:DCP>
        </ows:Operation>
        <ows:Operation name="GetPropertyValue">
            <ows:DCP>
                <ows:HTTP>
                    <ows:Get xlink:href="http://www.BlueOx.org/wfs/wfs.cgi?"/>
                    <ows:Post xlink:href="http://www.BlueOx.org/wfs/wfs.cgi"/>
                </ows:HTTP>
            </ows:DCP>
        </ows:Operation>
        <ows:Operation name="GetFeature">
            <ows:DCP>
                <ows:HTTP>
                    <ows:Get xlink:href="http://www.BlueOx.org/wfs/wfs.cgi?"/>
                    <ows:Post xlink:href="http://www.BlueOx.org/wfs/wfs.cgi"/>
                </ows:HTTP>
            </ows:DCP>
        </ows:Operation>
```

```
<ows:Parameter name="version">
        <ows:AllowedValues>
                <ows:Value>2.0.0</ows:Value>
        </ows:AllowedValues>
</ows:Parameter>
<!-- ************************************************** -->
<!-- *   CONFORMANCE DECLARATION                      * -->
<!-- ************************************************** -->
<ows:Constraint name="ImplementsBasicWFS">
        <ows:NoValues/>
        <ows:DefaultValue>TRUE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="ImplementsTransactionalWFS">
        <ows:NoValues/>
        <ows:DefaultValue>FALSE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="ImplementsLockingWFS">
        <ows:NoValues/>
        <ows:DefaultValue>FALSE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="KVPEncoding">
        <ows:NoValues/>
        <ows:DefaultValue>TRUE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="XMLEncoding">
        <ows:NoValues/>
        <ows:DefaultValue>TRUE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="SOAPEncoding">
        <ows:NoValues/>
        <ows:DefaultValue>FALSE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="ImplementsInheritance">
        <ows:NoValues/>
        <ows:DefaultValue>FALSE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="ImplementsRemoteResolve">
        <ows:NoValues/>
        <ows:DefaultValue>FALSE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="ImplementsResultPaging">
        <ows:NoValues/>
        <ows:DefaultValue>TRUE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="ImplementsStandardJoins">
        <ows:NoValues/>
        <ows:DefaultValue>TRUE</ows:DefaultValue>
</ows:Constraint>
<ows:Constraint name="ImplementsSpatialJoins">
        <ows:NoValues/>
        <ows:DefaultValue>TRUE</ows:DefaultValue>
</ows:Constraint>
```

```
            <ows:Constraint name="ImplementsTemporalJoins">
                <ows:NoValues/>
                <ows:DefaultValue>TRUE</ows:DefaultValue>
            </ows:Constraint>
            <ows:Constraint name="ImplementsFeatureVersioning">
                <ows:NoValues/>
                <ows:DefaultValue>FALSE</ows:DefaultValue>
            </ows:Constraint>
            <ows:Constraint name="ManageStoredQueries">
                <ows:NoValues/>
                <ows:DefaultValue>FALSE</ows:DefaultValue>
            </ows:Constraint>
            <!-- ************************************************** -->
            <!-- *   CAPACITY CONSTRAINTS                         * -->
            <!-- ************************************************** -->
            <ows:Constraint name="PagingIsTransactionSafe">
                <ows:NoValues/>
                <ows:DefaultValue>TRUE</ows:DefaultValue>
            </ows:Constraint>
            <ows:Constraint name="CountDefault">
                <ows:NoValues/>
                <ows:DefaultValue>1000</ows:DefaultValue>
            </ows:Constraint>
            <ows:Constraint name="ResolveTimeoutDefault">
                <ows:NoValues/>
                <ows:DefaultValue>300</ows:DefaultValue>
            </ows:Constraint>
            <ows:Constraint name="SortLevelLimit">
                <ows:NoValues/>
                <ows:DefaultValue>1</ows:DefaultValue>
            </ows:Constraint>
            <ows:Constraint name="ResolveLocalScope">
                <ows:NoValues/>
                <ows:DefaultValue>1..3</ows:DefaultValue>
            </ows:Constraint>
            <ows:Constraint name="ResolveRemoteScope">
                <ows:NoValues/>
                <ows:DefaultValue>5</ows:DefaultValue>
            </ows:Constraint>
            <ows:Constraint name="QueryExpressions">
                <ows:AllowedValues>
                    <ows:Value>wfs:Query</ows:Value>
                    <ows:Value>wfs:StoredQuery</ows:Value>
                </ows:AllowedValues>
            </ows:Constraint>
            <!-- ************************************************** -->
        </ows:OperationsMetadata>
        <FeatureTypeList>
            <FeatureType xmlns:veg="http://www.dgiwg.org/veg">
                <Name>veg:EA010</Name>
                <Title xml:lang="fr">Champs (cultivÃ©s)</Title>
                <Title xml:lang="en">Cropland</Title>
```

```
                <Abstract xml:lang="en">An area that has been tilled for the planting
crops.</Abstract>
                <ows:Keywords>
                    <ows:Keyword>E - Vegetation</ows:Keyword>
                    <ows:Keyword>EA - Vegetation - Cropland</ows:Keyword>
                    <ows:Keyword>VMap1</ows:Keyword>
                    <ows:Keyword xml:lang="fr">E- VÃ©gÃ©tation</ows:Keyword>
                    <ows:Keyword xml:lang="fr">EA- VÃ©gÃ©tation - Champs</ows:Keyword>
                </ows:Keywords>
                <DefaultCRS>urn:ogc:def:crs:EPSG::4326</DefaultCRS>
                <OtherCRS>urn:ogc:def:crs:EPSG::3395</OtherCRS>
                <OtherCRS>urn:ogc:def:crs:CRS:84</OtherCRS>
                <ows:WGS84BoundingBox>
                    <ows:LowerCorner>-180 -90</ows:LowerCorner>
                    <ows:UpperCorner>180 90</ows:UpperCorner>
                </ows:WGS84BoundingBox>
            </FeatureType>
            <FeatureType xmlns:veg="http://www.dgiwg.org/veg">
                <Name>veg:EA020</Name>
                <Title xml:lang="fr">Haies</Title>
                <Title xml:lang="en">Hedgerow</Title>
                <Abstract xml:lang="en">A continuous growth of shrubbery planted as fence, a
boundary, or a wind break.</Abstract>
                <ows:Keywords>
                    <ows:Keyword>E - Vegetation</ows:Keyword>
                    <ows:Keyword>EA - Vegetation - Cropland</ows:Keyword>
                    <ows:Keyword>VMap1</ows:Keyword>
                    <ows:Keyword xml:lang="fr">E- VÃ©gÃ©tation</ows:Keyword>
                    <ows:Keyword xml:lang="fr">EA- VÃ©gÃ©tation - Champs</ows:Keyword>
                </ows:Keywords>
                <DefaultCRS>urn:ogc:def:crs:EPSG::4326</DefaultCRS>
                <OtherCRS>urn:ogc:def:crs:EPSG::3395</OtherCRS>
                <OtherCRS>urn:ogc:def:crs:CRS:84</OtherCRS>
                <ows:WGS84BoundingBox>
                    <ows:LowerCorner>-180 -90</ows:LowerCorner>
                    <ows:UpperCorner>180 90</ows:UpperCorner>
                </ows:WGS84BoundingBox>
            </FeatureType>
            <FeatureType xmlns:hydro="http://www.dgiwg.org/hydro">
                <Name>hydro:BH080</Name>
                <Title xml:lang="fr">Lac/Etang</Title>
                <Title xml:lang="en">Lake/pond</Title>
                <Abstract xml:lang="en">A body of water surrounded by land.</Abstract>
                <ows:Keywords>
                    <ows:Keyword>B - Hydrography</ows:Keyword>
                    <ows:Keyword>BH - Hydrography - InlandWater</ows:Keyword>
                    <ows:Keyword>VMap1</ows:Keyword>
                    <ows:Keyword xml:lang="fr">B - Hydrographie</ows:Keyword>
                    <ows:Keyword xml:lang="fr">B - Hydrographie - Eaux
continentales</ows:Keyword>
                </ows:Keywords>
                <DefaultCRS>urn:ogc:def:crs:EPSG::4326</DefaultCRS>
```

```
                <OtherCRS>urn:ogc:def:crs:EPSG::3395</OtherCRS>
                <OtherCRS>urn:ogc:def:crs:CRS:84</OtherCRS>
                <ows:WGS84BoundingBox>
                    <ows:LowerCorner>-180 -90</ows:LowerCorner>
                    <ows:UpperCorner>180 90</ows:UpperCorner>
                </ows:WGS84BoundingBox>
            </FeatureType>
        </FeatureTypeList>
        <fes:Filter_Capabilities>
            <fes:Conformance>
                <fes:Constraint name="ImplementsQuery">
                    <ows:NoValues/>
                    <ows:DefaultValue>TRUE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsAdHocQuery">
                    <ows:NoValues/>
                    <ows:DefaultValue>TRUE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsFunctions">
                    <ows:NoValues/>
                    <ows:DefaultValue>FALSE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsMinStandardFilter">
                    <ows:NoValues/>
                    <ows:DefaultValue>TRUE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsStandardFilter">
                    <ows:NoValues/>
                    <ows:DefaultValue>TRUE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsMinSpatialFilter">
                    <ows:NoValues/>
                    <ows:DefaultValue>TRUE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsSpatialFilter">
                    <ows:NoValues/>
                    <ows:DefaultValue>TRUE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsMinTemporalFilter">
                    <ows:NoValues/>
                    <ows:DefaultValue>TRUE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsTemporalFilter">
                    <ows:NoValues/>
                    <ows:DefaultValue>TRUE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsVersionNav">
                    <ows:NoValues/>
                    <ows:DefaultValue>FALSE</ows:DefaultValue>
                </fes:Constraint>
                <fes:Constraint name="ImplementsSorting">
                    <ows:NoValues/>
```

```
                <ows:DefaultValue>FALSE</ows:DefaultValue>
            </fes:Constraint>
            <fes:Constraint name="ImplementsExtendedOperators">
                <ows:NoValues/>
                <ows:DefaultValue>FALSE</ows:DefaultValue>
            </fes:Constraint>
        </fes:Conformance>
        <fes:Id_Capabilities>
            <fes:ResourceIdentifier name="fes:ResourceId"/>
        </fes:Id_Capabilities>
        <fes:Scalar_Capabilities>
            <fes:LogicalOperators/>
            <fes:ComparisonOperators>
                <fes:ComparisonOperator name="PropertyIsLessThan"/>
                <fes:ComparisonOperator name="PropertyIsGreaterThan"/>
                <fes:ComparisonOperator name="PropertyIsLessThanOrEqualTo"/>
                <fes:ComparisonOperator name="PropertyIsGreaterThanOrEqualTo"/>
                <fes:ComparisonOperator name="PropertyIsEqualTo"/>
                <fes:ComparisonOperator name="PropertyIsNotEqualTo"/>
                <fes:ComparisonOperator name="PropertyIsLike"/>
                <fes:ComparisonOperator name="PropertyIsBetween"/>
                <fes:ComparisonOperator name="PropertyIsNull"/>
                <fes:ComparisonOperator name="PropertyIsNil"/>
            </fes:ComparisonOperators>
        </fes:Scalar_Capabilities>
        <fes:Spatial_Capabilities>
            <fes:GeometryOperands>
                <fes:GeometryOperand name="gml:Point"/>
                <fes:GeometryOperand name="gml:MultiPoint"/>
                <fes:GeometryOperand name="gml:LineString"/>
                <fes:GeometryOperand name="gml:MultiLineString"/>
                <fes:GeometryOperand name="gml:Curve"/>
                <fes:GeometryOperand name="gml:MultiCurve"/>
                <fes:GeometryOperand name="gml:Polygon"/>
                <fes:GeometryOperand name="gml:MultiPolygon"/>
                <fes:GeometryOperand name="gml:Surface"/>
                <fes:GeometryOperand name="gml:MultiSurface"/>
                <fes:GeometryOperand name="gml:MultiGeometry"/>
                <fes:GeometryOperand name="gml:Box"/>
                <fes:GeometryOperand name="gml:Envelope"/>
            </fes:GeometryOperands>
            <fes:SpatialOperators>
                <fes:SpatialOperator name="BBOX"/>
                <fes:SpatialOperator name="Equals"/>
                <fes:SpatialOperator name="Disjoint"/>
                <fes:SpatialOperator name="Intersects"/>
                <fes:SpatialOperator name="Touches"/>
                <fes:SpatialOperator name="Crosses"/>
                <fes:SpatialOperator name="Within"/>
                <fes:SpatialOperator name="Contains"/>
                <fes:SpatialOperator name="Overlaps"/>
                <fes:SpatialOperator name="Beyond"/>
```

```
                        <fes:SpatialOperator name="DWithin"/>
                    </fes:SpatialOperators>
                </fes:Spatial_Capabilities>
                <fes:Temporal_Capabilities>
                    <fes:TemporalOperands>
                        <fes:TemporalOperand name="gml:validTime"/>
                        <fes:TemporalOperand name="gml:TimeInstant"/>
                        <fes:TemporalOperand name="gml:TimePeriod"/>
                        <fes:TemporalOperand name="gml:timePosition"/>
                        <fes:TemporalOperand name="gml:timeInterval"/>
                        <fes:TemporalOperand name="gml:duration"/>
                    </fes:TemporalOperands>
                    <fes:TemporalOperators>
                        <fes:TemporalOperator name="During"/>
                        <fes:TemporalOperator name="After"/>
                        <fes:TemporalOperator name="Before"/>
                    </fes:TemporalOperators>
                </fes:Temporal_Capabilities>
            </fes:Filter_Capabilities>
        </WFS_Capabilities>
```

GetCapabilities document based on WFS 2.0 implementation by GeoServer :

```
<?xml version="1.0" encoding="UTF-8" ?>
-<wfs:WFS_Capabilities
    version="2.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://www.opengis.net/wfs/2.0" xmlns:wfs="http://www.opengis.net/wfs/2.0"
    xmlns:ows="http://www.opengis.net/ows/1.1" xmlns:gml="http://www.opengis.net/gml/3.2"
    xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xsi:schemaLocation="http://www.opengis.net/wfs/2.0
    http://localhost:8080/geoserver/schemas/wfs/2.0/wfs.xsd"
    xmlns:it.geosolutions="http://www.geo-solutions.it"
    xmlns:cite="http://www.opengeospatial.net/cite" xmlns:tiger="http://www.census.gov"
    xmlns:sde="http://geoserver.sf.net" xmlns:topp="http://www.openplans.org/topp"
    xmlns:sf="http://www.openplans.org/spearfish" xmlns:nurc="http://www.nurc.nato.int"
    updateSequence="125">
- <ows:ServiceIdentification>
- <ows:Operation name="GetCapabilities">
- <ows:DCP>
- <ows:HTTP>
  <ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
  <ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
    </ows:HTTP>
    </ows:DCP>
- <ows:Parameter name="AcceptVersions">
- <ows:AllowedValues>
  <ows:Value>1.0.0</ows:Value>
  <ows:Value>1.1.0</ows:Value>
  <ows:Value>2.0.0</ows:Value>
    </ows:AllowedValues>
```

```
    </ows:Parameter>
- <ows:Parameter name="AcceptFormats">
- <ows:AllowedValues>
  <ows:Value>text/xml</ows:Value>
    </ows:AllowedValues>
    </ows:Parameter>
    </ows:Operation>
- <ows:Operation name="DescribeFeatureType">
- <ows:DCP>
- <ows:HTTP>
  <ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
  <ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
    </ows:HTTP>
    </ows:DCP>
- <ows:Parameter name="outputFormat">
- <ows:AllowedValues>
  <ows:Value>text/xml; subtype=gml/3.2</ows:Value>
    </ows:AllowedValues>
    </ows:Parameter>
    </ows:Operation>
- <ows:Operation name="GetFeature">
- <ows:DCP>
- <ows:HTTP>
  <ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
  <ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
    </ows:HTTP>
    </ows:DCP>
- <ows:Parameter name="resultType">
- <ows:AllowedValues>
  <ows:Value>results</ows:Value>
  <ows:Value>hits</ows:Value>
    </ows:AllowedValues>
    </ows:Parameter>
- <ows:Parameter name="outputFormat">
- <ows:AllowedValues>
  <ows:Value>text/xml; subtype=gml/3.2</ows:Value>
  <ows:Value>GML2</ows:Value>
  <ows:Value>GML2-GZIP</ows:Value>
  <ows:Value>SHAPE-ZIP</ows:Value>
  <ows:Value>application/gml+xml; version=3.2</ows:Value>
  <ows:Value>csv</ows:Value>
  <ows:Value>gml3</ows:Value>
  <ows:Value>gml32</ows:Value>
  <ows:Value>json</ows:Value>
  <ows:Value>text/xml; subtype=gml/2.1.2</ows:Value>
  <ows:Value>text/xml; subtype=gml/3.1.1</ows:Value>
    </ows:AllowedValues>
    </ows:Parameter>
- <ows:Constraint name="PagingIsTransactionSafe">
  <ows:NoValues />
  <ows:DefaultValue>FALSE</ows:DefaultValue>
    </ows:Constraint>
```

```
- <ows:Constraint name="CountDefault">
  <ows:NoValues />
  <ows:DefaultValue>1000000</ows:DefaultValue>
    </ows:Constraint>
    </ows:Operation>
- <ows:Operation name="GetPropertyValue">
- <ows:DCP>
- <ows:HTTP>
  <ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
  <ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
    </ows:HTTP>
    </ows:DCP>
- <ows:Parameter name="resolve">
- <ows:AllowedValues>
  <ows:Value>none</ows:Value>
    </ows:AllowedValues>
    </ows:Parameter>
    </ows:Operation>
- <ows:Operation name="ListStoredQueries">
- <ows:DCP>
- <ows:HTTP>
  <ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
  <ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
    </ows:HTTP>
    </ows:DCP>
    </ows:Operation>
- <ows:Operation name="DescribeStoredQueries">
- <ows:DCP>
- <ows:HTTP>
  <ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
  <ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
    </ows:HTTP>
    </ows:DCP>
    </ows:Operation>
- <ows:Operation name="DropStoredQuery">
- <ows:DCP>
- <ows:HTTP>
  <ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
  <ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
    </ows:HTTP>
    </ows:DCP>
    </ows:Operation>
- <ows:Operation name="LockFeature">
- <ows:DCP>
- <ows:HTTP>
  <ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
  <ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
    </ows:HTTP>
    </ows:DCP>
- <ows:Operation name="GetFeatureWithLock">
- <ows:DCP>
- <ows:HTTP>
```

```
<ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
<ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
</ows:HTTP>
</ows:DCP>
- <ows:Parameter name="resultType">
- <ows:AllowedValues>
<ows:Value>results</ows:Value>
<ows:Value>hits</ows:Value>
</ows:AllowedValues>
</ows:Parameter>
- <ows:Parameter name="outputFormat">
- <ows:AllowedValues>
<ows:Value>text/xml; subtype=gml/3.2</ows:Value>
<ows:Value>GML2</ows:Value>
<ows:Value>GML2-GZIP</ows:Value>
<ows:Value>SHAPE-ZIP</ows:Value>
<ows:Value>application/gml+xml; version=3.2</ows:Value>
<ows:Value>csv</ows:Value>
<ows:Value>gml3</ows:Value>
<ows:Value>gml32</ows:Value>
<ows:Value>json</ows:Value>
<ows:Value>text/xml; subtype=gml/2.1.2</ows:Value>
<ows:Value>text/xml; subtype=gml/3.1.1</ows:Value>
</ows:AllowedValues>
</ows:Parameter>
- <ows:Constraint name="PagingIsTransactionSafe">
<ows:NoValues />
<ows:DefaultValue>FALSE</ows:DefaultValue>
</ows:Constraint>
- <ows:Constraint name="CountDefault">
<ows:NoValues />
<ows:DefaultValue>1000000</ows:DefaultValue>
</ows:Constraint>
</ows:Operation>
- <ows:Operation name="Transaction">
- <ows:DCP>
- <ows:HTTP>
<ows:Get xlink:href="http://localhost:8080/geoserver/wfs" />
<ows:Post xlink:href="http://localhost:8080/geoserver/wfs" />
</ows:HTTP>
</ows:DCP>
- <ows:Parameter name="inputFormat">
- <ows:AllowedValues>
<ows:Value>text/xml; subtype=gml/3.2</ows:Value>
</ows:AllowedValues>
</ows:Parameter>
- <ows:Parameter name="releaseAction">
- <ows:AllowedValues>
<ows:Value>ALL</ows:Value>
<ows:Value>SOME</ows:Value>
</ows:AllowedValues>
</ows:Parameter>
```

```
    </ows:Operation>
- <ows:Constraint name="ImplementsBasicWFS">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsTransactionalWFS">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsLockingWFS">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="KVPEncoding">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="XMLEncoding">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="SOAPEncoding">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsInheritance">
  <ows:NoValues />
  <ows:DefaultValue>FALSE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsRemoteResolve">
  <ows:NoValues />
  <ows:DefaultValue>FALSE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsResultPaging">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsStandardJoins">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsSpatialJoins">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsTemporalJoins">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="ImplementsFeatureVersioning">
  <ows:NoValues />
  <ows:DefaultValue>FALSE</ows:DefaultValue>
```

```
    </ows:Constraint>
- <ows:Constraint name="ManageStoredQueries">
  <ows:NoValues />
  <ows:DefaultValue>TRUE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="PagingIsTransactionSafe">
  <ows:NoValues />
  <ows:DefaultValue>FALSE</ows:DefaultValue>
    </ows:Constraint>
- <ows:Constraint name="QueryExpressions">
- <ows:AllowedValues>
  <ows:Value>wfs:Query</ows:Value>
  <ows:Value>wfs:StoredQuery</ows:Value>
    </ows:AllowedValues>
    </ows:Constraint>
    </ows:OperationsMetadata>
- <FeatureTypeList>
- <FeatureType xmlns:tiger="http://www.census.gov">
  <Name>tiger:poi</Name>
  <Title>Manhattan (NY) points of interest</Title>
  <Abstract>Points of interest in New York, New York (on Manhattan). One of the attributes contains
      the name of a file with a picture of the point of interest.
      </Abstract>
- <ows:Keywords>
  <ows:Keyword>poi</ows:Keyword>
  <ows:Keyword>Manhattan</ows:Keyword>
  <ows:Keyword>DS_poi</ows:Keyword>
  <ows:Keyword>points_of_interest</ows:Keyword>
    </ows:Keywords>
  <DefaultCRS>urn:ogc:def:crs:EPSG::4326</DefaultCRS>
- <ows:WGS84BoundingBox>
  <ows:LowerCorner>-74.0118315772888 40.70754683896324</ows:LowerCorner>
  <ows:UpperCorner>-74.00857344353275 40.711945649065406</ows:UpperCorner>
    </ows:WGS84BoundingBox>
    </FeatureType>
     </wfs:WFS_Capabilities>
```

C.2 GetPropertyValue examples

In response to the GetPropertyValue, which retrieves the value of a feature property, or part of the value of a complex feature property, from a data source for a given set of features identified by a query, request:

This example retrieves the geographic content only for the features in the topp:states layer:

```
<wfs:ValueCollection
   xsi:schemaLocation="http://www.opengis.net/wfs/2.0
   http://localhost:8080/geoserver/schemas/wfs/2.0/wfs.xsd">
   <wfs:member>
   <topp:the_geom>
   <gml:MultiSurface srsDimension="2" srsName="urn:ogc:def:crs:EPSG::4326">
   <gml:surfaceMember>
```

```
<gml:Polygon srsDimension="2">
 <gml:exterior>
 <gml:LinearRing>
 <gml:posList>Coordinates</gml:posList>
        </gml:LinearRing>
        </gml:exterior>
        </gml:Polygon>
        </gml:surfaceMember>
        </gml:MultiSurface>
        </topp:the_geom>
        </wfs:member>
<gml:surfaceMember>
<gml:Polygon srsDimension="2">
<gml:exterior>
<gml:LinearRing>
<gml:posList>Coordinates</gml:posList>
        </gml:LinearRing>
        </gml:exterior>
        </gml:Polygon>
        </gml:surfaceMember>
        </gml:MultiSurface>
        </topp:the_geom>
        </wfs:member>
        </wfs:ValueCollection>
```

C.3 DescribeFeatureType examples

The DescribeFeatureType request will return the list of feature types, sorted by namespace:

```
<xsd:schema
    xmlns:tiger="http://www.census.gov" xmlns:topp="http://www.openplans.org/topp"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified" targetNamespace="http://www.census.gov">
 <xsd:import
    namespace="http://www.census.gov"
    schemaLocation="http://localhost:8080/geoserver/wfs?request=DescribeFeatureType&version=
    2.0.0&service=WFS&outputFormat=text%2Fxml%3B+subtype%3Dgml%2F3.2&typeName=tiger
    %3Apoi" />
 <xsd:import
    namespace="http://www.openplans.org/topp"
    schemaLocation="http://localhost:8080/geoserver/wfs?request=DescribeFeatureType&version=
    2.0.0&service=WFS&outputFormat=text%2Fxml%3B+subtype%3Dgml%2F3.2&typeName=topp
    %3Astates" />
    </xsd:schema>
```

C.4 GetFeature examples

The request will execute a GetFeature request for a given layer namespace:featuretype:

```
<?xml version="1.0" encoding="UTF-8" ?>
<wfs:FeatureCollection numberMatched="49" numberReturned="49" timeStamp="2013-01-
26T00:55:37.062Z" xsi:schemaLocation="http://www.opengis.net/gml/3.2
http://localhost:8080/geoserver/schemas/gml/3.2.1/gml.xsd http://www.openplans.org/topp
http://localhost:8080/geoserver/wfs?service=WFS&version=2.0.0&request=DescribeFeatureTy
pe&typeName=topp%3Astates http://www.opengis.net/wfs/2.0
http://localhost:8080/geoserver/schemas/wfs/2.0/wfs.xsd" xmlns:tiger="http://www.census.gov"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:cite="http://www.opengeospatial.net/cite"
xmlns:nurc="http://www.nurc.nato.int" xmlns:sde="http://geoserver.sf.net"
xmlns:topp="http://www.openplans.org/topp" xmlns:it.geosolutions="http://www.geo-solutions.it"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sf="http://www.openplans.org/spearfish" xmlns:wfs="http://www.opengis.net/wfs/2.0">
<wfs:boundedBy>
    <gml:Envelope>
        <gml:lowerCorner>24.955967 -124.73142200000001</gml:lowerCorner>
        <gml:upperCorner>49.371735 -66.969849</gml:upperCorner>
    </gml:Envelope>
  </wfs:boundedBy>
<wfs:member>
    <topp:states gml:id="states.1">
    <gml:boundedBy>
        <gml:Envelope srsDimension="2" srsName="urn:ogc:def:crs:EPSG::4326">
            <gml:lowerCorner>36.986771000000005 -91.516129</gml:lowerCorner>
            <gml:upperCorner>42.50936100000001 -87.507889</gml:upperCorner>
        </gml:Envelope>
    </gml:boundedBy>
  <topp:the_geom>
    <gml:MultiSurface srsDimension="2" srsName="urn:ogc:def:crs:EPSG::4326">
        <gml:surfaceMember>
            <gml:Polygon srsDimension="2">
                <gml:exterior>
                    <gml:LinearRing>
                        <gml:posList>Coordinates</gml:posList>
                    </gml:LinearRing>
                </gml:exterior>
            </gml:Polygon>
        </gml:surfaceMember>
    </gml:MultiSurface>
  </topp:the_geom>
    <topp:STATE_NAME>Washington</topp:STATE_NAME>
    <topp:STATE_FIPS>53</topp:STATE_FIPS>
     <topp:SUB_REGION>Pacific</topp:SUB_REGION>
     <topp:STATE_ABBR>WA</topp:STATE_ABBR>
     <topp:LAND_KM>172447.205</topp:LAND_KM>
     <topp:WATER_KM>12226.63</topp:WATER_KM>
     <topp:PERSONS>4866692.0</topp:PERSONS>
     <topp:FAMILIES>1264934.0</topp:FAMILIES>
     <topp:HOUSHOLD>1872431.0</topp:HOUSHOLD>
```

```
    <topp:MALE>2413747.0</topp:MALE>
    <topp:FEMALE>2452945.0</topp:FEMALE>
    <topp:WORKERS>1830031.0</topp:WORKERS>
    <topp:DRVALONE>1700872.0</topp:DRVALONE>
    <topp:CARPOOL>282240.0</topp:CARPOOL>
    <topp:PUBTRANS>104403.0</topp:PUBTRANS>
    <topp:EMPLOYED>2293961.0</topp:EMPLOYED>
    <topp:UNEMPLOY>139216.0</topp:UNEMPLOY>
    <topp:SERVICE>637487.0</topp:SERVICE>
    <topp:MANUAL>302635.0</topp:MANUAL>
    <topp:P_MALE>0.496</topp:P_MALE>
    <topp:P_FEMALE>0.504</topp:P_FEMALE>
    <topp:SAMP_POP>736744.0</topp:SAMP_POP>
  </topp:states>
 </wfs:member>
</wfs:FeatureCollection>
```

C.5 ListStoredQueries example

This URL gives an example for accessing the list of queries stored on a WFS2.0 server:

This XML fragment is the response returned by the server:

```
<wfs:ListStoredQueriesResponse
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:fes="http://www.opengis.net/fes/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:bw="http://www.geodaten.bayern.de"
xsi:schemaLocation="http://www.opengis.net/wfs/2.0 http://schemas.opengis.net/wfs/2.0/wfs.xsd ">
        <wfs:StoredQuery id="urn:ogc:def:query:OGC-WFS::GetFeatureById">
                <wfs:Title xml:lang="en">Get feature by identifier</wfs:Title>
                <wfs:ReturnFeatureType>gml:AbstractFeatureType</wfs:ReturnFeatureType>
        </wfs:StoredQuery>
        <wfs:StoredQuery id="DWithinQuery">
                <wfs:Title xml:lang="en">Perimeter search for cities</wfs:Title>
                <wfs:ReturnFeatureType
xmlns:bvv="http://www.geodaten.bayern.de">bw:gmd_ex</wfs:ReturnFeatureType>
        </wfs:StoredQuery>
        <wfs:StoredQuery id="CityById">
                <wfs:Title xml:lang="en">Request City by Id</wfs:Title>
                <wfs:ReturnFeatureType
xmlns:bvv="http://www.geodaten.bayern.de">bvv:gmd_ex</wfs:ReturnFeatureType>
        </wfs:StoredQuery>
        <wfs:StoredQuery id="urn:wei:def:query:OGC-WFS::InspireStoredQueryExample">
                <wfs:Title xml:lang="en">INSPIRE pre-defined WFS - Stored Query Example</wfs:Title>
                <wfs:ReturnFeatureType
xmlns:bvv="http://www.geodaten.bayern.de">bvv:gmd_ex</wfs:ReturnFeatureType>
        </wfs:StoredQuery>
        <wfs:StoredQuery id="CityMuenchenEpsg31468">
                <wfs:Title xml:lang="en">Request for City Munich</wfs:Title>
```

```
            <wfs:ReturnFeatureType
xmlns:bvv="http://www.geodaten.bayern.de">bvv:gmd_ex</wfs:ReturnFeatureType>
        </wfs:StoredQuery>
        <wfs:StoredQuery id="bboxQuery">
            <wfs:Title xml:lang="de">BoundingBox Abfrage für Gemeinden</wfs:Title>
            <wfs:ReturnFeatureType
xmlns:bvv="http://www.geodaten.bayern.de">bvv:gmd_ex</wfs:ReturnFeatureType>
        </wfs:StoredQuery>
</wfs:ListStoredQueriesResponse>
```

C.6 Describe Stored Queries example

This URL gives an example for accessing a description of the stored query “DWithinQuery” on a WFS2.0 server:

This XML fragment is the response returned by the server:

```
<wfs:DescribeStoredQueriesResponse
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:fes="http://www.opengis.net/fes/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:bw="http://www.geodaten.bayern.de"
xsi:schemaLocation="http://www.opengis.net/wfs/2.0 http://schemas.opengis.net/wfs/2.0/wfs.xsd">
        <wfs:StoredQueryDescription id="DWithinQuery">
            <wfs:Title xml:lang="en"> Perimeter search for cities </wfs:Title>
            <wfs:Parameter name="x" type="xsi:double"/>
            <wfs:Parameter name="y" type="xsi:double"/>
            <wfs:Parameter name="radius" type="xsi:double"/>
            <wfs:QueryExpressionText isPrivate="false"
language="urn:ogc:def:queryLanguage:OGC-WFS::WFS_QueryExpression"
xmlns:bw="http://www.geodaten.bayern.de" returnFeatureTypes="bw:gmd_ex">
                <wfs:Query xmlns:fes="http://www.opengis.net/fes/2.0"
xmlns:ows="http://www.opengis.net/ows/1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:bw="http://www.geodaten.bayern.de" xmlns:xlink="http://www.w3.org/1999/xlink" handle="Q01"
srsName="EPSG:31468" typeNames="bvv:gmd_ex">
                    <fes:Filter>
                        <fes:DWithin>
                            <fes:ValueReference>the_geom</fes:ValueReference>
                            <gml:Point gml:id="P1" srsName="EPSG:31468">
                                <gml:pos>${x} ${y}</gml:pos>
                            </gml:Point>
                            <fes:Distance uom="m">10</fes:Distance>
                        </fes:DWithin>
                    </fes:Filter>
                </wfs:Query>
            </wfs:QueryExpressionText>
        </wfs:StoredQueryDescription>
</wfs:DescribeStoredQueriesResponse>
```

Annex D
Stored Queries examples
(Informative)

This example finds all the features in a given polygon (AreaOfInterest)

```
<wfs:StoredQueryDefinition
        id="http://www.interactive-instruments.de/stored-queries/ows-10/tds-
dgiwg/FeaturesInPolygon">
        <wfs:Title>Features In Polygon</wfs:Title>
        <wfs:Abstract>Find all the features in a Polygon.</wfs:Abstract>
        <wfs:Parameter name="AreaOfInterest" type="gml:SurfacePropertyType"/>
        <wfs:QueryExpressionText
            returnFeatureTypes="tds:AerodromeBoundaryGeocurve
tds:AircraftHangarGeopoint tds:AircraftHangarGeosurface
            tds:BridgeGeocurve tds:BridgeGeocurve tds:BuildingGeopoint
tds:BuildingGeosurface
            tds:ControlTowerGeopoint tds:ControlTowerGeosurface
tds:EmbankmentGeocurve tds:EmbankmentGeosurface
            tds:HelipadGeopoint tds:LandAerodromeGeosurface
tds:LandSubjectToInundationGeosurface
            tds:LandWaterBoundaryGeocurve tds:MarshGeosurface
tds:RailwayGeocurve tds:RailwaySidetrackGeocurve
            tds:RailwayYardGeosurface tds:RoadGeocurve
tds:RoadInterchangeGeopoint tds:RunwayGeosurface
            tds:ShorelineConstructionGeocurve tds:ShorelineConstructionGeosurface
tds:StopwayGeosurface
            tds:TaxiwayGeosurface tds:TidalWaterGeosurface tds:TowerGeopoint
tds:TunnelGeocurve"
            language="urn:ogc:def:queryLanguage:OGC-WFS::WFS_QueryExpression"
            isPrivate="false">
            <wfs:Query typeNames="tds:AerodromeBoundaryGeocurve">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                    </fes:Within>
                </fes:Filter>
            </wfs:Query>
            <wfs:Query typeNames="tds:AircraftHangarGeopoint">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                    </fes:Within>
                </fes:Filter>
            </wfs:Query>
            <wfs:Query typeNames="tds:AircraftHangarGeosurface">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
```

```
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:BridgeGeocurve">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:BuildingGeopoint">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:BuildingGeosurface">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:ControlTowerGeopoint">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:ControlTowerGeosurface">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:EmbankmentGeocurve">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
```

```
<wfs:Query typeNames="tds:EmbankmentGeosurface">
        <fes:Filter>
                <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                </fes:Within>
        </fes:Filter>
</wfs:Query>
<wfs:Query typeNames="tds:HelipadGeopoint">
        <fes:Filter>
                <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                </fes:Within>
        </fes:Filter>
</wfs:Query>
<wfs:Query typeNames="tds:LandAerodromeGeosurface">
        <fes:Filter>
                <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                </fes:Within>
        </fes:Filter>
</wfs:Query>
<wfs:Query typeNames="tds:LandSubjectToInundationGeosurface">
        <fes:Filter>
                <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                </fes:Within>
        </fes:Filter>
</wfs:Query>
<wfs:Query typeNames="tds:LandWaterBoundaryGeocurve">
        <fes:Filter>
                <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                </fes:Within>
        </fes:Filter>
</wfs:Query>
<wfs:Query typeNames="tds:MarshGeosurface">
        <fes:Filter>
                <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                </fes:Within>
        </fes:Filter>
</wfs:Query>
<wfs:Query typeNames="tds:RailwayGeocurve">
        <fes:Filter>
                <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
```

```
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:RailwaySidetrackGeocurve">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:RailwayYardGeosurface">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:RoadGeocurve">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:RoadInterchangeGeopoint">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:RunwayGeosurface">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
    <wfs:Query typeNames="tds:ShorelineConstructionGeocurve">
        <fes:Filter>
            <fes:Within>
                <fes:ValueReference>tds:geometry</fes:ValueReference>
                ${AreaOfInterest}
            </fes:Within>
        </fes:Filter>
    </wfs:Query>
```

```
            <wfs:Query typeNames="tds:ShorelineConstructionGeosurface">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                    </fes:Within>
                </fes:Filter>
            </wfs:Query>
            <wfs:Query typeNames="tds:StopwayGeosurface">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                    </fes:Within>
                </fes:Filter>
            </wfs:Query>
            <wfs:Query typeNames="tds:TaxiwayGeosurface">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                    </fes:Within>
                </fes:Filter>
            </wfs:Query>
            <wfs:Query typeNames="tds:TidalWaterGeosurface">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                    </fes:Within>
                </fes:Filter>
            </wfs:Query>
            <wfs:Query typeNames="tds:TowerGeopoint">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                    </fes:Within>
                </fes:Filter>
            </wfs:Query>
            <wfs:Query typeNames="tds:TunnelGeocurve">
                <fes:Filter>
                    <fes:Within>
                        <fes:ValueReference>tds:geometry</fes:ValueReference>
                        ${AreaOfInterest}
                    </fes:Within>
                </fes:Filter>
            </wfs:Query>
        </wfs:QueryExpressionText>
    </wfs:StoredQueryDefinition>
```

Annex E
Client Requirements and Recommendations

The Normative client requirements requested by this profile are summarized in Table 13.

Table 13: Normative client requirements

No.	Requirement	Cardinality
1	A WFS client that claims to be conformant with the DGIWG Basic WFS profile shall support all capabilities as defined in the OGC 09-025r2 Basic WFS conformance class.	M
2	A WFS client that claims to be conformant with the DGIWG Basic WFS profile, shall support stored queries according to Table 6.	M
3	A WFS client that claims to be conformant with the DGIWG WFS Basic profile shall support OGC 09-026R2 Standard Filter conformance classes.	M
4	A WFS client that claims to be conformant with the DGIWG WFS Basic profile shall support OGC 09-026R2 Spatial Filter conformance classes.	M
5	In addition, a WFS client that claims to be conformant with the DGIWG WFS Basic profile shall support Equals, Disjoint, Intersects, Touches, Overlaps, Contains, Within, Crosses, DWithin, Beyond.	M
6	A WFS client that claims to be conformant with the DGIWG WFS Basic profile shall support OGC 09-026R2 Temporal Filter conformance classes.	M
7	A WFS client that claims to be conformant with the DGIWG WFS Basic profile shall support at least After, Before.	M
8	A WFS client that claims to be conformant with the DGIWG WFS Basic profile shall implement the HTTP GET conformance class.	M
9	A WFS client that claims to be conformant with the DGIWG WFS Basic profile shall implement the HTTP POST conformance class.	M
10	A WFS client that claims to be conformant with the DGIWG Transactional (Locking) WFS shall support all capabilities as defined in the OGC 09-025R2 Locking WFS conformance class.	M
11	A WFS client that claims to be conformant with the DGIWG Transactional (Locking) WFS profile shall satisfy all requirements defined for DGIWG WFS basic conformant clients (requirements 1 - 10)	M
12	A WFS client that claims to be conformant with the DGIWG Transactional (Locking) WFS profile shall support at least the LockFeature operation.	M
13	A WFS client that claims to be conformant with the DGIWG WFS profiles shall support at least. • CRS:84 WGS84 geographic longitude, then latitude, expressed in decimal degrees • EPSG:4326 WGS84 geographic latitude, then longitude, expressed in decimal degrees	M
14	A client implementing the DGIWG WFS profiles shall support constraints on operation as required by Table 8.	M

No.	Requirement	Cardinality
15	A WFS client that claims to be conformant with the DGIWG Basic WFS profile shall support the updateSequence attribute.	M
16	A WFS client that claims to be conformant with the DGIWG WFS profiles shall be able to percent-encode angle brackets using %3C for '<' and %3E for '>' for Filter requests.	M
17	A WFS client shall support service exceptions in English language.	M
18	A WFS client shall be able to support all mandatory and optional service metadata elements.	M
19	A WFS client shall be able to report classification levels provided in the <ows:AccessConstraints> element.	M

The non-normative client recommendations requested by this profile are summarized in Table 14.

Table 14: DGIWG WFS Profile Non-normative recommendations for Client Implementation

No.	Recommendation	Cardinality
1	A WFS client that claims to support the DGIWG WFS Basic profile should implement OGC 09-026R2 Spatial filters	O
2	A WFS client that claims to support the DGIWG WFS Basic profile should implement OGC 09-026R2 Temporal filters	O
3	A WFS client that claims to support the DGIWG WFS profile should implement Response paging.	O
4	A WFS client that claims to support the DGIWG WFS profile should be able to report from the GetCapabilities document which WFS profile (ows:profile) and conformance classes (ows:constraint) are supported by the server.	O
5	A WFS client that claims to support the DGIWG WFS profiles should provide URN/URL designations as defined in the profile	O
6	The followings CRSs should be supported as well: • EPSG:3395 World Mercator projection Among the following Coordinate Reference Systems, the service shall support all those which validity zone overlaps data published by the service: • UTM projections over WGS84 (north zones)… EPSG:32601 to EPSG:32660 • UTM projections over WGS84 (south zones)… EPSG:32701 to EPSG:32760 • UPS projection over WGS84 (north zone)…… EPSG:5041 • UPS projection over WGS84 (south zone)…… EPSG:5042 Optionally data may be provided in additional CRSs, depending on national requirements.	O

Annex F
DGIM groups and subgroups in baseline 2016-2.0
(Informative)

This annex shows the groups and subgroups in the DGFCD, as in DGIF baseline 2016-2.0. Each group and each subgroup is uniquely identified by either one of these two fields: alphaCode (*name)* and 531-Code (*alias)*.

Each subgroup belongs to exactly one group. One group (Abstract – G22) is not related to any subgroup.

Group G01: IndustriesServices
- Subgroup SG0101: Extraction
- Subgroup SG0102: FabricationProcessing
- Subgroup SG0103: Agriculture
- Subgroup SG0104: PowerSupplies
- Subgroup SG0105: Communication
- Subgroup SG0106: AssociatedSupportStruct
- Subgroup SG0107: StorageProvision
- Subgroup SG0108: WasteManagement

Group G02: SocioEconomicGeography
- Subgroup SG0201: Habitats
- Subgroup SG0202: SettlementsAssociated
- Subgroup SG0203: EconomicCommercial
- Subgroup SG0204: Leisure
- Subgroup SG0205: PoliticsAdministration
- Subgroup SG0206: SciencesEducation
- Subgroup SG0207: CulturalContext

Group G03: Transportation
- Subgroup SG0301: Railways
- Subgroup SG0302: RoadsTracks
- Subgroup SG0303: GuidedTransportation
- Subgroup SG0304: WaterBorneTransportation
- Subgroup SG0305: AirTransportation
- Subgroup SG0306: Restrictions
- Subgroup SG0307: CrossingsLinks
- Subgroup SG0308: TransportationAssociated
- Subgroup SG0309: SpaceTransportation
- Subgroup SG0310: DistributionNetworks

Group G04: HydrographyOceanography
- Subgroup SG0401: CoastalLittoralZones
- Subgroup SG0402: PortsHarbours
- Subgroup SG0403: Depths
- Subgroup SG0404: NatureOfSeabed
- Subgroup SG0405: OffshoreConstructInstall
- Subgroup SG0406: TidesCurrents
- Subgroup SG0407: RoutesNavigation

- Subgroup SG0408: HazardsObstructions
- Subgroup SG0409: SeaIce
- Subgroup SG0410: RegulatedRestrictedZones
- Subgroup SG0411: InlandWaters
- Subgroup SG0412: PhysicsOfWater

Group G05: Physiography
- Subgroup SG0501: Hypsography
- Subgroup SG0502: Geomorphology
- Subgroup SG0503: Rocks
- Subgroup SG0504: Soils
- Subgroup SG0505: NaturalResources
- Subgroup SG0506: SeismologyVolcanology
- Subgroup SG0507: Glaciers
- Subgroup SG0508: Anomalies
- Subgroup SG0509: GlobalEarthCover

Group G06: Biota
- Subgroup SG0601: CultivatedLand
- Subgroup SG0602: Rangeland
- Subgroup SG0603: Woodland
- Subgroup SG0604: Wetland
- Subgroup SG0605: AridAreas
- Subgroup SG0606: RegionsRestrictedAreas
- Subgroup SG0607: Fauna
- Subgroup SG0608: Flora

Group G07: Demarcation
- Subgroup SG0701: BoundariesLimits
- Subgroup SG0702: LandSurveyRealEstate

Group G08: Aeronautical
- Subgroup SG0801: AerodromesMoveSurfLighting
- Subgroup SG0802: AirspaceRoutes
- Subgroup SG0803: NavaidsLandAidsPointsObst
- Subgroup SG0804: ServicesOrgsTimetables
- Subgroup SG0805: TerminalProcedures

Group G09: Military
- Subgroup SG0901: DefensiveOperationalStruct
- Subgroup SG0902: RestrictedAreasBoundaries
- Subgroup SG0903: OperationsEvents

Group G12: WeatherClimate
- Subgroup SG1001: WeatherPhenomena
- Subgroup SG1002: ClimateConditions
- Subgroup SG1003: ClimateZonesRegions

Group G11: Characteristics
- Subgroup SG2001: Position

- Subgroup SG2002: MeasurableValues
- Subgroup SG2003: DatesDurations
- Subgroup SG2004: Appearance
- Subgroup SG2005: FunctionStatus

Group G20: NamesDesignations
- Subgroup SG2101: Names
- Subgroup SG2102: Designations

Group G21: MetadataReferences
- Subgroup SG2201: Annotation
- Subgroup SG2202: Portrayal
- Subgroup SG2203: DateCurrency
- Subgroup SG2204: Quality
- Subgroup SG2205: ReferencesSources
- Subgroup SG2206: SystemsOfClassification

Group G22: Abstract