OGC® DOCUMENT: 25-014

External identifier of this OGC® document: http://www.opengis.net/doc/IS/ogcapi-edr-3/1.0

Open Geospatial Consortium

OGC API - ENVIRONMENTAL DATA RETRIEVAL STANDARD - PART 3: PROFILES

STANDARD
Implementation

DRAFT

Version: 1.0
Submission Date: 2025-05-14
Approval Date: 2029-03-30
Publication Date: 2029-03-30
Editor: Mark Burgoyne, Charles Heazel

License Agreement

Use of this document is subject to the license agreement at https://www.ogc.org/license

Suggested additions, changes and comments on this document are welcome and encouraged. Such suggestions may be submitted using the online change request form on OGC web site: http://ogc.standardstracker.org/

Copyright notice

Note

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

CONTENTS

I. ABSTRACT vi

II. KEYWORDS vi

III. PREFACE vii

IV. SECURITY CONSIDERATIONS viii

V. SUBMITTING ORGANIZATIONS ix

VI. SUBMITTERS ix

VII. CONTRIBUTORS ix

PREFACE 11

1. SCOPE 2

2. CONFORMANCE 4

3. NORMATIVE REFERENCES 6

4. TERMS AND DEFINITIONS 8

5. CONVENTIONS 12
   5.1. Identifiers 12

6. CONTEXT 14
   6.1. Standardization Goal 14
   6.2. Standardization Target Type 14
   6.3. Profiles 15

7. REQUIREMENT CLASS CORE 17
   7.1. General 17
   7.2. Requests 19
   7.3. Response 23
   7.4. Query Expressions 24
   7.5. Asynchronous Operations 28

8. MEDIA TYPES FOR ANY DATA ENCODING(S) 31

ANNEX A (INFORMATIVE) CONFORMANCE CLASS ABSTRACT TEST SUITE (NORMATIVE) 33
A.1. Conformance Class A 33

ANNEX B (INFORMATIVE) TITLE 36

ANNEX C (INFORMATIVE) REVISION HISTORY 38

BIBLIOGRAPHY 40

LIST OF TABLES

Table C.1 38

LIST OF RECOMMENDATIONS

REQUIREMENTS CLASS 1: REQUIREMENTS CLASS 'CORE' 17
REQUIREMENT 1 17
REQUIREMENT 2 17
REQUIREMENT 3 18
REQUIREMENT 4 18
REQUIREMENT 5 19
REQUIREMENT 6 20
REQUIREMENT 7 21
REQUIREMENT 8 22
REQUIREMENT 9 22
REQUIREMENT 10 23
REQUIREMENT 11 23
REQUIREMENT 12 23
REQUIREMENT 13 24
REQUIREMENT 14 25
REQUIREMENT 15 25
REQUIREMENT 16 25

REQUIREMENT 17 26
REQUIREMENT 18 26
REQUIREMENT 19 27
REQUIREMENT 20 27
REQUIREMENT 21 28
REQUIREMENT 22 28
RECOMMENDATION 1 22
PERMISSION 1 28
CONFORMANCE CLASS A.1 33

I ABSTRACT

<Insert Abstract Text here>

II KEYWORDS

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

ogcdoc, OGC document, API, openapi, html

III PREFACE

The aim of an OGC API EDR profile is to ensure interoperability between API implementations. To achieve this, it is essential that providers use a consistent approach when defining collections and instances of collections. An OGC EDR profile will specify a set of requirements that an API implementation must support to be a compliant implementation.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

IV SECURITY CONSIDERATIONS

No security considerations have been made for this Standard.

V SUBMITTING ORGANIZATIONS

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

- UK Met Office
- Heazeltech

VI SUBMITTERS

All questions regarding this submission should be directed to the editor or the submitters:

Name	Affiliation

VII CONTRIBUTORS

Additional contributors to this Standard include the following:

Individual name(s), Organization

PREFACE

PREFACE

NOTE: Insert Preface Text here. Give OGC specific commentary: describe the technical content, reason for document, history of the document and precursors, and plans for future work. > Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

1 SCOPE

1 SCOPE

NOTE: Insert Scope text here. Give the subject of the document and the aspects of that scope covered by the document.

2

CONFORMANCE

2 CONFORMANCE

This standard defines XXXX.

Requirements for N standardization target types are considered:

- AAAA
- BBBB

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

In order to conform to this OGC® interface standard, a software implementation shall choose to implement:

- Any one of the conformance levels specified in Annex A (normative).
- Any one of the Distributed Computing Platform profiles specified in Annexes TBD through TBD (normative).

All requirements-classes and conformance-classes described in this document are owned by the standard(s) identified.

3

NORMATIVE REFERENCES

3 NORMATIVE REFERENCES

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

ISO: ISO 19106, *Geographic information — Profiles*. International Organization for Standardization, Geneva https://www.iso.org/standard/26011.html.

OGC API — Environmental Data Retrieval Standard

The ModSpec Model — Part 1: Core — A Standard for Designing and Writing Modular Standards

OpenAPI Initiative (OAI). **OpenAPI Specification 3.0** [online]. 2020 [viewed 2025-01-03]. The latest patch version at the time of publication of this standard was 3.0.4, available at https://spec.openapis.org/oas/v3.0.4

OpenAPI Initiative (OAI). **OpenAPI Specification 3.1** [online]. 2021 [viewed 2025-01-03]. The latest patch version at the time of publication of this standard was 3.1.1, available at https://spec.openapis.org/oas/v3.1.1

4

TERMS AND DEFINITIONS

4 TERMS AND DEFINITIONS

This document uses the terms defined in OGC Policy Directive 49, which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this document and OGC documents do not use the equivalent phrases in the ISO/IEC Directives, Part 2.

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

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

This document uses the terms defined in OGC Policy Directive 49, which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

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

4.1. Collection

Body of resources that belong or are used together. An aggregate, set, or group of related resources.

4.2. Conformance Module; Conformance Test Module

A set of related conformance classes and their associated components.

Note 1 to entry: When no ambiguity is possible, the word `test` may be omitted. i.e. conformance test module is the same as conformance module. Conformance modules may be nested in a hierarchical way.

[**SOURCE:** OGC 08-131r5]

4.3. Conformance Class; Conformance Test Class

A set of conformance tests that must be passed to receive a single certificate of conformance.

Note 1 to entry: When no ambiguity is possible, the word *test* may be left out, so conformance test class maybe called a conformance class.

[**SOURCE:** OGC 08-131r5]

4.4. Conformance Test

A test, abstract or real, of one or more requirements contained within a standard, or set of standards.

[**SOURCE:** OGC 08-131r5]

4.5. Requirement

Expression in the content of a standard conveying criteria to be fulfilled if compliance with the standard is to be claimed and from which no deviation is permitted.

[**SOURCE:** OGC 08-131r5]

4.6. Requirements Class

An aggregate of requirements with a single standardization target type that must all be satisfied to pass a conformance test Class.

[**SOURCE:** OGC 08-131r5]

4.7. Requirements Module

A set of related requirement classes and their associated components.

[**SOURCE:** OGC 08-131r5]

4.8. Standardization Goal

A concise statement of the problem that the standard helps address and the strategy envisioned for achieving a solution. This strategy typically identifies real-world entities that need to be modified or constrained. At the abstract level, those entities are the Standardization Target Types.

[**SOURCE:** OGC 08-131r5]

4.9. Standardization Target

Entity to which some requirements of a standard apply.

Note 1 to entry: The standardization target is the entity which may receive a certificate of conformance for a requirements class.

[**SOURCE:** OGC 08-131r5]

4.10. Standardization Target Type

Type of entity or set of entities to which the requirement of a standard apply

Note 1 to entry: For example, the standardization target type for The OGC API – Features Standard are Web APIs. The standardization target type for the CDB Standard is “datastore”. It is important to understand that a standard’s root standardization target type can have sub-types, and that there can be a hierarchy of target types. For example, a Web API can have sub types of client, server, security, and so forth. As such, each requirements class can have a standardization target type that is a sub-type of the root.

[**SOURCE:** OGC 08-131r5]

5

CONVENTIONS

5 CONVENTIONS

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

5.1. Identifiers

The normative provisions in this standard are denoted by the URI

`http://www.opengis.net/doc/IS/ogcapi-edr-3/1.0`

All requirements and conformance tests that appear in this document are denoted by partial URIs which are relative to this base.

5.1.1. Shortcuts

In the interest of readability, the following terms will be used as shorthand for more complex text:

- Profile: A Profile is a standard or specification which restricts and/or extends an existing standard. This standard defines the rules for creating a profile of the OGC API — Environmental Data Retrieval Standard. The term “Profile” will be used in this document as shorthand for “profile of the OGC API — Environmental Data Retrieval Standard”.
- OGC-EDR: The term OGC-EDR will be used in this document as shorthand for the term “OGC API — Environmental Data Retrieval Standard”

6 CONTEXT

6 CONTEXT

6.1. Standardization Goal

The goal of this Standard is to ensure interoperabilty between implementations of the OGC API — Environmental Data Retrieval Standard (EDR API).

The OGC EDR Standard does not try to address every possible application domain. Rather, it provides a foundation which can be tailored for a specific domain. The result of this tailoring is a domain specific “profile” of the EDR API Standard.

A significant risk to this approach is that, in the act of profiling, interoperabilty will be compromized. This risk can be mitigated by establishing rules for how the EDR API Standard can be profiled. The goal of this Standard is to define a set of rules sufficient to ensure interoperability while retaining the adaptability needed to support domain-specific requirements.

6.2. Standardization Target Type

The Standardization Target Type for this Standard is the set of standards and specifications which profile the OGC API — Environmental Data Retrieval Standard.

It is important to understand that:

- This Standard is a standard for writing standards. It does not address the EDR API implementation.
- This Standard is a profile of the OGC ModSpec Model — Part 1: Core — A Standard for Designing and Writing Modular Standards (ModSpec).
- Implementations of this Standard are Profiles of the OGC API — Environmental Data Retrieval Standard
- The profiling model used is defined in ISO 19106:2004 Geographic information — Profiles

6.3. Profiles

ISO 19106:2004 Geographic information — Profiles is the ISO TC211 Standard for developing profiles of ISO TC211 Standards. This standard defined two conformance classes. These conformance classes can be thought of as two classes of profile.

- A Class 1 profile is a pure subset of the ISO geographic information standards.
- A Class 2 profile has the same basis as Class 1 but include extensions within the contexts permitted in the base standard. Additionally, a Class 2 profile permits the profiling of non-ISO geographic information standards as part of the profile.

In other words, a Class 1 profile restricts the base standard while a Class 2 profile both restricts and extends the base standard.

Both Class 1 and Class 2 Profiles of the OGC-EDR Standard are allowed.

NOTE: Consider organizing requirements based on whether they are Class 1 or Class 2.

7

REQUIREMENT CLASS CORE

7 REQUIREMENT CLASS CORE

Paragraph

REQUIREMENTS CLASS 1: REQUIREMENTS CLASS 'CORE'	
IDENTIFIER	`http://www.opengis.net/spec/ABCD/m.n/req/req-class-a`
TARGET TYPE	Web API
CONFORMANCE CLASS	Conformance class A.1: `http://www.opengis.net/spec/name-of-standard/1.0/conf/example1`
NORMATIVE STATEMENTS	Requirement 1-1: `/req/req-class-a/req-name-1` Requirement 1-2: `/req/req-class-a/req-name-2`

7.1. General

7.1.1. Conformance to the ModSpec

REQUIREMENT 1	
IDENTIFIER	`/req/core/modspec`
STATEMENT	A profile of the OGC API — Environmental Data Retrieval Standard *SHALL* be conformant to the OGC Modular Specification.

7.1.2. Profile of OGC API — EDR

REQUIREMENT 2	
IDENTIFIER	`/req/core/edr-conformant`
STATEMENT	A profile of the OGC API — Environmental Data Retrieval Standard *SHALL* require that a conformant implementation (standardization target) of that profile also demonstrate conformance to the OGC API — Environmental Data Retrieval Standard.

7.1.3. Publish as an OpenAPI document

REQUIREMENT 3	
IDENTIFIER	`/req/core/publishing`
STATEMENT	An EDR profile *SHALL* be published as an OpenAPI JSON document.
A	The rules described in the requirements *SHALL* be encoded using the OpenAPI 3.1 specification.
B	The requirement rules *SHALL* be encoded in either the OpenAPI Path Item or in the Response object schema sections of the document.
C	The profile OpenAPI document *SHALL* describe the profile EDR API as follows: • The servers attributes of the OpenAPI root object *SHALL* be blank (the profile is not linked to specific implementations) • The Extent requirement rules *SHALL* be encoded in the JSON schema defined in the 200 responses for the /collections and /collections/{collection} id Paths object of the profile Open API document • The data_query type requirement rules *SHALL* be encoded in the JSON schema defined in the 200 responses for the /collections and /collections/{collection} id Paths object of the profile OpenAPI document • The data_query types *SHALL* be encoded as Paths objects in the OpenAPI document, where appropriate the output_format, default_output_format, crs, within_units, width-units, height-units and limit (paging) requirements SHALL be encoded in the child Parameter objects of the Paths object. • The output_format requirement rules *SHALL* be encoded in the 200 responses of the data_query type Paths objects • The Parameter_names requirements *SHALL* be encoded in the JSON schema defined in the 200 responses for the /collections and /collections/{collection} id Paths object of the profile Open API document.
D	An EDR API *SHALL* advertise the location of the profile OpenAPI document it complies with
E	An EDR API *SHALL* advertise the location of the profile OpenAPI document it complies with in the links section of the API root with a link relation value of 'profile'

NOTE: Consider moving part 3 to a separate requirement.

REQUIREMENT 4	
IDENTIFIER	`/req/core/api`
STATEMENT	A Profile of the OGC API — Environmental Data Retrieval Standard *SHALL* require that an implementation specify the versions of OpenAPI supported.

7.2. Requests

7.2.1. Parameter Names

REQUIREMENT 5	
IDENTIFIER	`/req/core/parameter-names`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* the valid values and definitions of parameter_names, then,
A	Requirements *SHALL* be defined which specify the parameter_names and their definitions.
B	The parameter_names requirement definitions *SHALL* specify the required parameter_names objects in full including: • name, • units, • data type and • measurement duration

STATEMENT	for example:
	`"parameter_names": {`
	`"prmsl": {`
	`"type": "Parameter",`
	`"description": "Air pressure at sea level",`
	`"unit": {`
	`"label": "Pascals",`
	`"symbol": {`
	`"value": "Pa",`
	`"type": "https://qudt.org/vocab/unit/PA"`
	`}`
	`},`
	`"observedProperty": {`
	`"id": "http://codes.wmo.int/grib2/codeflag/4.2/_0-3-1",`
	`"label": "MSL Pressure"`
	`}`
	`},`
	`"t2m": {`
	`"type": "Parameter",`
	`"description": "Air temperature at 2m",`
	`"unit": {`
	`"label": "Kelvin",`
	`"symbol": {`
	`"value": "K",`
	`"type": "https://qudt.org/vocab/unit/K"`
	`}`
	`},`
	`"observedProperty": {`
	`"id": "http://codes.wmo.int/grib2/codeflag/4.2/0-0-0",`
	`"label": "Air temperature at 2m"`
	`}`
	`},`
	`"dd": {`
	`"type": "Parameter",`

REQUIREMENT 5


```
                "description": "Wind Direction",
                "unit": {
                    "label": "degree true",
                    "symbol": {
                        "value": "°",
                        "type": "https://qudt.org/vocab/unit/DEG"
                    }
                },
                "observedProperty": {
                    "id": "http://codes.wmo.int/grib2/codeflag/4.2/0-2-0",
                    "label": "Wind Direction"
                },
                "measurementType": {
                    "method": "mean",
                    "duration": "-PT10M"
                }
            },
            "ff": {
                "type": "Parameter",
                "description": "10m Wind Speed",
                "unit": {
                    "label": "m/s",
                    "symbol": {
                      "value": "ms-1",
                      "type": "https://qudt.org/vocab/unit/M/s"
                    }
                },
                "observedProperty": {
                    "id": "http://codes.wmo.int/grib2/codeflag/4.2/0-2-1",
                    "label": "10m Wind Speed"
                },
                "measurementType": {
                    "method": "mean",
                    "duration": "-PT10M"
                }
            }
        }
```


7.2.2. Collection ID Parameter

REQUIREMENT 6

IDENTIFIER	`/req/core/collectionid`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* the valid values of the Collection ID parameter, then:
A	The Profile *SHALL* specify the rules that the Collection ID values must follow.
B	Those rules *SHALL* be specified using either: • identifier string or • Regular expression defining valid string patterns.

7.2.3. Extent Parameter

REQUIREMENT 7	
IDENTIFIER	`/req/core/extent`
A	A requirement MAY be defined specifying the rules for defining the Collection extent.
B	An extent requirement MAY specify the minimum spatial bounds that SHALL be supported
C	An extent requirement MAY specify the minimum temporal bounds that SHALL be supported
D	An extent requirement MAY specify the minimum vertical bounds that SHALL be supported
E	An extent requirement MAY specfiy the minimum bounds of custom extents that SHALL be supported
F	An extent requirements definition SHALL specify the rules for the spatial CRS. The attributes are constrained by one of: • Enumerated list of valid crs values • Regular expression defining valid crs string patterns.
G	If the collection has a temporal dimension an extent requirements definition SHALL specify the rules fo temporal TRS. The attributes are constrained by one of: • Enumerated list of valid trs values • Regular expression defining valid trs string patterns.
H	If the collection has a vertical dimension an extent requirements definition SHALL specify the rules for the vertical VRS. The attributes are constrained by one of: • Enumerated list of valid vrs values • Regular expression defining valid vrs string patterns.
I	For any custom dimensions an extent requirements definition SHALL specify the rules for the custom reference systems. The attributes are constrained by one of: • custom dimension name • custom dimension reference value • Where applicable enumerated list of valid custom dimension values
STATEMENT	(Regular expressions could be used to restrict reference system definitions to WKT2 or EPSG values)

NOTE: Needs to be re-visited

7.2.4. Instance ID Parameter

REQUIREMENT 8	
IDENTIFIER	`/req/core/instanceid`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* the valid values of the Instance ID parameter, then:
A	The Profile *SHALL* specify the rules that the Instance ID values must follow.
B	Those rules *SHALL* be specified using either: • identifier string • Regular expression defining valid string patterns.

7.2.5. Output Format Parameter

REQUIREMENT 9	
IDENTIFIER	`/req/core/output-format`
STATEMENT	For every output_format specified in any of the data_query enumerated lists, there *SHALL* be a requirement which defines the schema or structure of the data (depending on the format).

RECOMMENDATION 1	
IDENTIFIER	`/rec/core/output-format`
STATEMENT	The recommended definition approaches are as follows: • JSON – Link to a JSON Schema definition • XML – Link to a XML Schema definition • CSV, TSV, PSV, SSV – Link to a definition based on the CSV on the web recommendations available from the CSV on the Web Working Group. • Other types (e.g. binary file types) – Link to a description of the format

NOTE: Question: Where should the CSV citation point. There are multiple CSV on the Web Recommendataions.

7.2.6. Paging

REQUIREMENT 10	
IDENTIFIER	`/req/core/paging-support`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard is *extended* to support paging, then:
A	A requirement _SHALL _ be created for each combination of query pattern and output format that must support paging.
B	Each paging requirement *SHALL* specify the default number of items to return per page request.

7.3. Response

7.3.1. Response with Status Codes

REQUIREMENT 11	
IDENTIFIER	`/req/core/status-codes`
STATEMENT	A Profile of the OGC API — Environmental Data Retrieval Standard *SHALL* require that the definitions of all http status codes SHALL be provided.
A	These definitions *SHALL* provide the following: • A description of the cause of the error. • A JSON schema for the message body structure

7.3.2. Response with Links

REQUIREMENT 12	
IDENTIFIER	`/req/core/links`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* valid responses to only those which include links, then:

REQUIREMENT 12	
A	The Profile *SHALL* require that link objects are included in a response.
B	The link objects *SHALL* defined the required link objects in full.
C	The link objects *SHALL* require that the `href`, `rel` and `type` attributes are populated.

STATEMENT	for example:
	`{`
	`"href": "https://creativecommons.org/licenses/by-nc/4.0/",`
	`"rel": "licence",`
	`"type": "text/html"`
	`},`
	`{`
	`"href": "https://docs.ogc.org/cs/21-069r2/21-069r2.html",`
	`"rel": "service-doc",`
	`"title": "CoverageJSON Community Standard v1.0"`
	`"type": "text/html"`
	`}`

7.4. Query Expressions

7.4.1. Data Query

REQUIREMENT 13	
IDENTIFIER	`/req/core/data-query`
STATEMENT	A Profile of the OGC API — Environmental Data Retrieval Standard Standard *SHALL* require definition of the supported data queries.
A	The data_queries definitions *SHALL* specify which data queries a service supports. This can be defined as follows: • Enumerated list of query types
B	Each data_query type listed *SHALL* have a requirement definition.

7.4.2. Area Query

REQUIREMENT 14	
IDENTIFIER	`/req/core/data-query-area`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* data queries by making the Area query mandatory, then:
A	The Profile *SHALL* include a requirement mandating the Area query.
B	The Area query requirement *SHALL* specify the following: • Enumerated list of output_format types • The default_output_format • Enumerated list of crs_details values • Enumerated list of the operations that the query supports (i.e. GET, POST)

7.4.3. Corridor Query

REQUIREMENT 15	
IDENTIFIER	`/req/core/data-query-corridor`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* data queries by making the Corridor query mandatory, then:
A	The Profile *SHALL* include a requirement mandating the Corridor query.
B	The Corridor requirement *SHALL* specify the following: • Enumerated list of output_format types • The default_output_format • Enumerated list of crs_details values • Enumerated list of width-units values • Enumerated list of height-units values • Enumerated list of the operations that the query supports (i.e. GET, POST)

7.4.4. Cube Query

REQUIREMENT 16	
IDENTIFIER	`/req/core/data-query-cube`

REQUIREMENT 16	
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* data queries by making the Cube query mandatory, then:
A	The Profile *SHALL* include a requirement mandating the Cube query.
B	The Cube query requirement *SHALL* specify the following: • Enumerated list of output_format types • The default_output_format • Enumerated list of crs_details values • Enumerated list of the operations that the query supports (i.e. GET, POST)

7.4.5. Instance Query

REQUIREMENT 17	
IDENTIFIER	`/req/core/data-query-instances`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *extends* data queries by making the Instances within a Collection queryable, then:
A	Instances *SHALL* be defined in the data_queries enumerated list.
B	A *NULL* value *SHALL* be used to indicate that no child instances can be queried.

7.4.6. Position Query

REQUIREMENT 18	
IDENTIFIER	`/req/core/data-query-position`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* data queries by making the Position query mandatory, then:
A	The Profile *SHALL* include a requirement mandating the Position query.
B	The Position query requirement *SHALL* specify the following: • Enumerated list of output_format types • The default_output_format • Enumerated list of crs_details values

REQUIREMENT 18	
	• Enumerated list of the operations that the query supports (i.e. GET, POST)
C	The position query requirement *SHALL* also specify the logic used in selecting the data returned by the response, i.e. exact, nearest neighbour, most representative or interpolated.

7.4.7. Radius Query

REQUIREMENT 19	
IDENTIFIER	`/req/core/data-query-radius`
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *restricts* data queries by making the Radius query mandatory, then:
A	The Profile *SHALL* include a requirement mandating the Radius query.
B	The Radius query requirement *SHALL* specify the following: • Enumerated list of output_format types • The default_output_format • Enumerated list of crs_details values • Enumerated list of within_units values • Enumerated list of the operations that the query supports (i.e. GET, POST)

7.4.8. Trajectory Query

REQUIREMENT 20	
IDENTIFIER	`/req/core/data-query-trajectory`
STATEMENT	If a Profile of the OGC-EDR Standard *restricts* data queries by making the Trajectory query mandatory, then:
A	The Profile *SHALL* include a requirement mandating the Trajectory query.
B	The Trajectory query requirement *SHALL* specify the following: • Enumerated list of output_format types • The default_output_format • Enumerated list of crs_details values • Enumerated list of the operations that the query supports (i.e. GET, POST)

7.5. Asynchronous Operations

While Web protocols typically use request-response operations, there is also support for asychonous operations.

7.5.1. HTTP Asynchrouous

This requirement address the use of HTTP asynchronous operations such as Webhooks and Callbacks.

REQUIREMENT 21	
IDENTIFIER	`/req/core/asynchronous`
STATEMENT	If a Profile *extends* the OGC API — Environmental Data Retrieval Standard with support for asynchronous opperations, then:
A	Requirements *SHALL* be defined for each query type that is asynchronous
B	Each asynchronous query type requirement *SHALL* define the HTTP Status Code and provide a message schema and text used to inform the user that the response is asynchronous.
C	Each asynchronous query type requirement *SHALL* document the mechanism for delivering the result of the asynchronous query.

PERMISSION 1	
IDENTIFIER	`/per/core/asynchronous`
STATEMENT	The documentation of the mechanism for delivering the result of the asynchronous query *MAY* be provided through a link to an external document.

7.5.2. Publish-Subscribe

This requirement addresses the use of Publish-Subscribe protocols. These are protocols supported in addition to HTTP.

REQUIREMENT 22	
IDENTIFIER	`/req/core/pubsub`

REQUIREMENT 22	
STATEMENT	If a Profile of the OGC API — Environmental Data Retrieval Standard *extends* the supported operations to include Publish-Subscribe operations, then:
A	Support for the OGC API — Environmental Data Retrieval — Part 2: Publish-Subscribe workflow Standard *SHALL* be required.
B	The pubsub requirement *SHALL* specify the channels that *SHALL* be supported
C	The pubsub requirement *SHALL* specify the payloads that a pubsub channel *SHALL* support

8

MEDIA TYPES FOR ANY DATA ENCODING(S)

8 MEDIA TYPES FOR ANY DATA ENCODING(S)

A section describing the MIME-types to be used is mandatory for any standard involving data encodings. If no suitable MIME type exists in http://www.iana.org/assignments/media-types/index.html then this section may be used to define a new MIME type for registration with IANA.

A

ANNEX A (INFORMATIVE) CONFORMANCE CLASS ABSTRACT TEST SUITE (NORMATIVE)

A

ANNEX A (INFORMATIVE) CONFORMANCE CLASS ABSTRACT TEST SUITE (NORMATIVE)

NOTE: Ensure that there is a conformance class for each requirements class and a test for each requirement (identified by requirement name and number)

A.1. Conformance Class A

CONFORMANCE CLASS A.1	
IDENTIFIER	`http://www.opengis.net/spec/name-of-standard/1.0/conf/example1`
REQUIREMENTS CLASS	Requirements class 1: `http://www.opengis.net/spec/ABCD/m.n/req/req-class-a`
TARGET TYPE	Web API
CONFORMANCE TESTS	Abstract test A.1: `/conf/core/map-response` Abstract test A.2: `/conf/core/http`

A.1.1. Example 1

ABSTRACT TEST A.1	
IDENTIFIER	`/conf/core/map-response`
REQUIREMENT	Requirement 1-1: `/req/req-class-a/req-name-1`
TEST PURPOSE	Verify that the implementation's response for the map retrieval operation is correct
TEST METHOD	A single step test method can be documented as a single line.

A.1.2. Example 2

ABSTRACT TEST A.2	
IDENTIFIER	`/conf/core/http`
REQUIREMENT	Requirement 1-2: `/req/req-class-a/req-name-2`
TEST PURPOSE	Validate that the resource paths advertised through the API conform with HTTP 1.1 and, where appropriate, TLS.
DESCRIPTION	**Example:** A sequential multi-step test method can be documented as shown here. This is the first step. This is the second step.

B

ANNEX B (INFORMATIVE) TITLE

B

ANNEX B (INFORMATIVE) TITLE

NOTE: Place other Annex material in sequential annexes beginning with “B” and leave final two annexes for the Revision History and Bibliography

C

ANNEX C (INFORMATIVE) REVISION HISTORY

C

ANNEX C (INFORMATIVE) REVISION HISTORY

Table C.1

DATE	RELEASE	EDITOR	PRIMARY CLAUSES MODIFIED	DESCRIPTION
2016-04-28	0.1	G. Editor	all	initial version

BIBLIOGRAPHY

BIBLIOGRAPHY

NOTE: The TC has approved Springer LNCS as the official document citation type.

Springer LNCS is widely used in technical and computer science journals and other publications

– Actual References:

[n] Journal: Author Surname, A.: Title. Publication Title. Volume number, Issue number, Pages Used (Year Published)

[1] *OGC: OGC Testbed 12 Annex B: Architecture* (2015).