Open Geospatial Consortium Inc.

Date:   2012-06-22

Reference number of this OGC® project document:    OGC 12-055r1

Version: 0.0.5

Category: OGC® Implementation Specification

Editor:   Clemens Portele, Satish Sankaran

GeoServices REST API — Part 2: Catalog

**Copyright notice**

Copyright © 2012 Open Geospatial Consortium, Inc. All Rights Reserved.
To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>.

**Warning**

This document is not an OGC Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard.

Document type:    OGC® Publicly Available Standard

Document subtype:    n/a

Document stage:    Draft

Document language:  English

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.

Draft copyright:

Copyright: Esri Inc.

The companies listed above have granted the Open Geospatial Consortium, Inc. (OGC) a nonexclusive, royalty-free, paid up, worldwide license to copy and distribute this document and to modify this document and distribute copies of the modified version.

Contents

1 Scope 1

2 Conformance 1

3 References 2

4 Terms and Definitions 2

5 Conventions 2

6 Catalog 2

6.1 Overview 2

6.2 Catalog 4

6.2.1 Catalog URI 4

6.2.2 Catalog resources 5

6.2.3 Examples 5

A.1 Conformance class: catalog 7

A.1.1 Test: catalog/requests 7

A.1.2 Test: catalog/uniqueServiceName 7

Preface

The “Esri GeoServices REST Specification Version 1.0” was originally developed by Esri to provide interoperability between ArcGIS Server and the broader information technology community. The Esri specification had been widely implemented by Esri users and business partners over 4 years. In 2010 it was released as a non-proprietary open specification and has been implemented by developers outside of the Esri user community.

In 2011, Esri has offered the GeoServices REST API for consideration to become an OGC standard. An OGC Standards Working Group was formed to document the specification in conformance with the modular specification policy of the OGC and to address comments received from the OGC membership and during the public review.

This candidate standard is designed to be implemented without the use of Esri products.

Submitting organizations

The following organizations submitted this Implementation Specification to the Open Geospatial Consortium Inc.:

Esri Inc.

interactive instruments GmbH

Oracle USA

52°North

Submission contact points

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

|  |  |
| --- | --- |
| CONTACT | COMPANY |
| 1. Satish Sankaran
 | 1. Esri Inc.
 |
| 1. Keith Ryden
 | 1. Esri Inc.
 |
| 1. Clemens Portele
 | 1. interactive instruments GmbH
 |
| 1. John Herring
 | 1. Oracle USA
 |
| 1. Andreas Wytzisk
 | 1. 52° North
 |

Revision history

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Release | Author | Paragraph modified | Description |
| 1. 25 Nov 2011
 | 1. 0.0.1
 | 1. CP, SS
 | 1. All
 | 1. New document (catalog extracted from core)
 |
| 1. 23 Jan 2012
 | 1. 0.0.2
 | 1. CP
 |  | 1. Harmonisation with other parts
 |
| 1. 01 May 2012
 | 1. 0.0.3
 | 1. CP
 | 1. All
 | 1. Include changes identified in review process
 |
| 1. 31 May 2012
 | 1. 0.0.4
 | 1. CP
 |  | 1. RFC package for OAB review
 |
| 1. 22 Jun 2012
 | 1. 0.0.5
 | 1. CP
 |  | 1. References updated
2. Editorial changes
 |

Changes to the OGC® Abstract Specification

The OGC**®** Abstract Specification does not require changes to accommodate this OGC**®** standard.

Versioning Rules

See the “Versioning Rules” section in OGC document 12-054r1, GeoServices REST API – Part 1: Core.

Foreword

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. Open Geospatial Consortium Inc. shall not be held responsible for identifying any or all such patent rights. However, to date, no such rights have been claimed or identified.

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 specification set forth in this document, and to provide supporting documentation.

This document is part 2 of the GeoServices REST API series:

Part 1: Core

Part 2: Catalog

Part 3: Map Service

Part 4: Feature Service

Part 5: Geometry Service

Part 6: Image Service

Part 7: Geoprocessing Service

Part 8: Geocoding Service

The relationship with other parts of the OGC standards baseline is described in document 12-062r1.

# Scope

The GeoServices REST API provides a standard way for web clients to communicate with geographic information system (GIS) servers based on Representational State Transfer (REST) principles. Clients issue requests to the resources on the server identified by structured URLs. The server responds with map images, text-based geographic information, or other representations of resources that satisfy the request.

This document specifies the catalog resources in an implementation of the GeoServices REST API and extends the GeoServices REST API – Core standard.

# Conformance

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

This Standard establishes one requirements class and a corresponding conformance class, extending the core conformance class of the GeoServices REST API series.

All requirements-classes and conformance-classes described in this document are owned by the standard identified as **http://www.opengis.net/spec/gsr-cs/1.0**. Requirements and conformance test URIs defined in this document are relative to this URI unless they start with "http://" and are absolute URIs.

Any implementation claiming conformance with a conformance class shall pass all the tests in the associated abstract test suite. Table 1 summarizes the requirements and conformance tests associated per conformance class.

Table 1 – Conformance class summary

|  |  |  |
| --- | --- | --- |
| **catalog** | **Title** | Catalog Service |
| **Standardization target type** | Web service |
| **Dependencies** | **http://www.opengis.net/spec/gsr/1.0/conf/core** |
| **Requirements** | All requirements in Clause 7 |
| **Conformance tests** | Annex A.1 |

# References

The following normative documents contain provisions that, through reference in this text, constitute provisions of this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

GeoServices REST API – Core, Version 1.0 (2012), OGC document 12-054r1

# Terms and Definitions

This document uses the terms defined in Sub-clause 5.3 of [OGC 06-121r9], which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

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

catalog

set of service interfaces which support organization, discovery, and access of geospatial information

[OGC Abstract Specification Topic 13]

# Conventions

See Clause 5 in the GeoServices REST API – Core document.

# Catalog

## Overview

The catalog typically serves as the initial entry point into a server providing GeoServices REST API web services.

A catalog resource references a set of folders (which are catalog resources themselves) and lists services published on the server.

The following figure provides an overview of the resource hierarchy in a Catalog Service. The Folder resource level is optional. The resources shown in grey are specified in other parts of the GeoServices REST API series.

Figure 1 – Resource overview

A Catalog resource optionally includes the specVersion, productName and currentVersion properties.

The specVersion is the version of the GeoServices REST API through which the catalog is implemented. If specVersion is not included in the response, its value is assumed to be 1.0.

The productName and currentVersion property MAY be used to specify a name and version of the implementer's software.

Table 2 – Catalog overview

|  |  |  |
| --- | --- | --- |
| **Resource** | **Parameters** | **Resource representation** |
| Catalog  | f=json | JSON representation validAll JSON schema elements supported |

## Catalog

### Catalog URI

In the following URI templates, these variables are used:

* serverRootURI: the URL of a Catalog resource without any parameter

Table 3 – Catalog reference

|  |  |
| --- | --- |
| **URI template** | {+serverRootURI}/{?f} |
| **HTTP methods** | GET |
| **Parent Resource Type** | - |
| **Child Resource Types** | CatalogMap ServiceGeocode ServiceGeoprocessing ServiceGeometry ServiceImage ServiceFeature Service |

Table 4 – Catalog parameters

|  |  |
| --- | --- |
| **Parameter** | **Details** |
| f | The response format.  |
| Required | Yes |
| Syntax | "json" |
| Example | f=json |

**Request requirements**

|  |
| --- |
| * + 1. The Catalog resource SHALL accept requests that conform to the URI template in Table 3 and use any HTTP method identified in the same table.

catalog/request |

|  |
| --- |
| * + 1. A Catalog resource SHALL support all parameters and values specified in Table 4.

catalog/parameters |

### Catalog resources

|  |
| --- |
| * + 1. The JSON representation of the Catalog resource SHALL validate against the JSON Schema http://schemas.opengis.net/gsr-cs/1.0/catalog.json or in case of an exception against JSON Schema http://schemas.opengis.net/gsr/1.0/exception.json.

catalog/valid |

|  |
| --- |
| * + 1. All service names in a Catalog resource SHALL be unique.

catalog/uniqueServiceNames |

|  |
| --- |
| * + 1. For every folder name in the array "folders" of a Catalog resource, the resource at {+serverRootURI}/{folderName}/{?f} SHALL be a conformant Catalog resource.

catalog/folderReferences |

NOTE Requirements on the services listed in the Catalog are part of the requirements classes of the specific service type, e.g. the Map Service.

### Examples

**Example 1**

URL for the root directory of a server:

http://www.example.com/rest/services/?f=json

Request

GET /rest/services/?f=json HTTP/1.1

Host: www.example.com

Response

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: nnn

{

"specVersion": 1.0,

"currentVersion": 10,

"folders":["Editing", "USA"],

"services":[

 {"name" : "Anaheim", "type" : "MapServer"},

 {"name" : "Switzerland", "type" : "MapServer"},

 {"name" : "USALocator", "type" : "GeocodeServer"}

]

}

**Example 2**

URL for the folder "Editing":

http://www.example.com/rest/services/Editing/?f=json

Request

GET /rest/services/Editing/?f=json HTTP/1.1

Host: www.example.com

Response

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: nnn

{

"specVersion": 1.0,

"currentVersion": 10,

"folders":[],

"services":[

 {"name" : "DamageReports", "type" : "FeatureServer"}

]

}

Annex A
(normative)

Abstract Test Suite

Conformance class: catalog

* 1. Test: catalog/requests

|  |  |
| --- | --- |
| Requirements | **catalog/request, catalog/parameters, catalog/valid, catalog/uniqueServiceNames, catalog/folderReferences** |
| Test purpose | Verify that the service supports the request and response requirements. |
| Test method | Set up a test service that includes folders. Construct valid requests for the catalogue and the folders.Inspect the responses and validate them against the JSON Schema http://schemas.opengis.net/gsr-cs/1.0/catalog.json or for exceptions against http://schemas.opengis.net/gsr/1.0/exception.json. Verify that all service names are unique. |
| Test type | Capability |

* 1. Test: catalog/uniqueServiceName

|  |  |
| --- | --- |
| Requirements | **catalog/uniqueServiceNames** |
| Test purpose | Verify that service names are always unique. |
| Test method | Set up a test service and attempt to define two services with identical service names. Verify that the implementation rejects service names that are not unique. |
| Test type | Capability |

1. [www.opengeospatial.org/cite](http://www.opengeospatial.org/cite) [↑](#footnote-ref-1)