

GeoSPARQL 1.1 Release Notes

Table of Contents

Introduction	2
Scope	2
Major changes between versions 1.0 and 1.1	2
Document contributor contact points	5
References	5
Terms and definitions	5
RDFS	6
coordinate reference system	6
Change Log	6
KEY	6
Change Table	7
Description of Critical Changes	9
Description of Substantive Changes	9
Collection classes	9
New size properties	10
Metric properties	10
New Feature properties	10
New geometry serializations	11
Extra functions for performing non-topological spatial operations	11
Functions for geometry collections	11
Spatial aggregate functions	12
Metric buffer and distance functions	12
Future Work	12
Appendix A: Revision History	13

Preface

This document provides the set of revision notes for OGC GeoSPARQL 1.1 [OGC 22-047] and does not modify that standard.

This document provides the details of edits, deficiency corrections, and enhancements of the above-referenced standard. It also documents those items that have been deprecated. Finally, this document provides implementations details related to issues of backwards compatibility.

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.

Abstract

GeoSPARQL defines a core set of Semantic Web classes, properties and datatypes for the representation of spatial information on the web and SPARQL extension functions for spatial data queries.

Introduction

Scope

The OGC GeoSPARQL standard supports representing and querying geospatial data on the Semantic Web. GeoSPARQL defines a vocabulary for representing geospatial data in RDF, and it defines extensions to the SPARQL query language for processing geospatial data.

GeoSPARQL Version 1.1 contains no breaking changes to 1.0, but does contain additions and clarifications. Profile resources, new ontology elements, and new functions were added. The major changes are summarized in the tables below.

Major changes between versions 1.0 and 1.1

The major changes are given in the tables below.

New resource	Location
Profile definition	http://www.opengis.net/def/geosparql
GeoSPARQL Rules in RIF	http://www.opengis.net/def/geosparql-rifrules
RDF validation file (SHACL Shapes)	http://www.opengis.net/def/geosparql-shapes
SPARQL Service Descriptions	http://www.opengis.net/def/geosparql/servicedescription
Requirements of GeoSPARQL modeled in RDF	http://www.opengis.net/def/geosparql/reqs

The SHACL Shapes are also available as a test suite.

The following new ontology elements and new functions are normatively defined in the OGC GeoSPARQL 1.1 specification document.

New element	Section
Classes	

New element	Section
Spatial Object Collection class	Class: geo:SpatialObjectCollection
Feature Collection class	Class: geo:FeatureCollection
Geometry Collection class	Class: geo:GeometryCollection
<i>Spatial Object Properties</i>	
hasSize	Property: geo:hasSize
hasMetricSize	Property: geo:hasMetricSize
hasLength	Property: geo:hasLength
hasMetricLength	Property: geo:hasMetricLength
hasPerimeterLength	Property: geo:hasPerimeterLength
hasMetricPerimeterLength	Property: geo:hasMetricPerimeterLength
hasArea	Property: geo:hasArea
hasMetricArea	Property: geo:hasMetricArea
hasVolume	Property: geo:hasVolume
hasMetricVolume	Property: geo:hasMetricVolume
<i>Feature Properties</i>	
hasBoundingBox	Property: geo:hasBoundingBox
hasCentroid	Property: geo:hasCentroid
<i>Geometry Serializations</i>	
geoJSONLiteral	RDFS Datatype: geo:geoJSONLiteral
asGeoJSON	Property: geo:asGeoJSON
asGeoJSON function	Function: geof:asGeoJSON
kmlLiteral	RDFS Datatype: geo:kmlLiteral
asKML	Property: geo:asKML
asKML function	Function: geof:asKML
dggsLiteral	RDFS Datatype: geo:dggsLiteral
asDGGS	Property: geo:asDGGS
asDGGS function	Function: geof:asDGGS
<i>Non-topological Query Functions</i>	
area	Function: geof:area
boundingCircle	Function: geof:boundingCircle
concaveHull	Function: geof:concaveHull
coordinateDimension	Function: geof:coordinateDimension
dimension	Function: geof:dimension

New element	Section
geometryN	Function: geof:geometryN
geometryType	Function: geof:geometryType
is3D	Function: geof:is3D
isEmpty	Function: geof:isEmpty
isMeasured	Function: geof:isMeasured
isSimple	Function: geof:isSimple
length	Function: geof:length
maxX	Function: geof:maxX
maxY	Function: geof:maxY
maxZ	Function: geof:maxZ
metricArea	Function: geof:metricArea
metricDistance	Function: geof:metricDistance
minX	Function: geof:minX
minY	Function: geof:minY
minZ	Function: geof:minZ
numGeometries	Function: geof:numGeometries
spatialDimension	Function: geof:spatialDimension
transform	Function: geof:transform
<i>Spatial Aggregate Functions</i>	
aggBoundingBox	Function: geof:aggBoundingBox
aggBoundingCircle	Function: geof:aggBoundingCircle
aggCentroid	Function: geof:aggCentroid
aggConcaveHull	Function: geof:aggConcaveHull
aggConvexHull	Function: geof:aggConvexHull
aggUnion	Function: geof:aggUnion

GeoSPARQL 1.1 also adds several new helpful resources:

- Profile Definition: a formal listing of all the GeoSPARQL 1.1 resources, defined using [the W3C Profiles Vocabulary](#).
- GeoSPARQL Rules in Rule Interchange Format (RIF)
 - and a RIF rule generation script
 - GeoSPARQL 1.0 indicated a RIF template but not a RIF file
- RDF data validation file
 - defined using SHACL shapes

- GeoSPARQL / CQL mappings
 - informative description of interoperability between the Common Query Language & GeoSPARQL
 - of relevance to the the delivery of GeoSPARQL data via systems such as the OGC's Web Feature Service [WFS] and OGC API Features [OGCAPIF] which implement CQL
- Alignments (mappings) to fifteen other well known and/or spatial ontologies
 - that are either commonly used with GeoSPARQL or could be

Document contributor contact points

All questions regarding this document should be directed to the contacts provided below or the referenced standard editor(s).

Table 1. Contacts

Name	Organization
Linda van den Brink	Geonovum
Nicholas J. Car	SURROUND Australia Pty Ltd

References

The following normative documents are new or updated references in the standard to which these Release Notes apply.

- [OGCOM ISO19156](#), *ISO 19156: Geographic information — Observations and measurements*
- [IETF3987](#), Internet Engineering Task Force, *RFC 3987: Internationalized Resource Identifiers (IRIs)*
- [OWL2](#) *OWL 2 Web Ontology Language Document Overview (Second Edition)*
- [RDF](#), *RDF 1.1 Concepts and Abstract Syntax*
- [RDFS](#) *RDF Schema 1.1*
- [RIFCORE](#), *RIF Core Dialect (Second Edition)*
- [SPARQL](#), *SPARQL 1.1 Query Language*
- [SPARQLPROT](#), *SPARQL 1.1 Protocol*
- [SPARQLRESX](#), *SPARQL Query Results XML Format (Second Edition)*
- [SPARQLRESJ](#), *SPARQL 1.1 Query Results JSON Format*

Terms and definitions

GeoSPARQL 1.0 contained only acronyms in its Terms & Definitions section, for example:

GeoJSON Geographic JavaScript Object Notation

GFM	General Feature Model (as defined in ISO 19109)
XML	Extensible Markup Language

GeoSPARQL 1.1 contains those acronyms, with additions such as *Spatial Reference System* (SRS), but it also contains a list of 11 defined terms such as:

RDFS

RDF Schema provides a data-modelling vocabulary for RDF data. RDF Schema is an extension of the basic RDF vocabulary.

and

coordinate reference system

A coordinate reference system (CRS) is a coordinate system that is related to an object by a datum.

These definitions have been included to clear up ambiguity regarding the origin and exact use of terms in GeoSPARQL.

Change Log

KEY

- Source:
 - Change Request
 - GitHub Issue
 - Other
- Identifier: Change Request number or issue number and pull request/commit in GitHub
- Type:
 - A=Administrative
 - S=Substantive
 - C=Critical

See [Description of Critical Changes](#) for more information on critical changes and [Description of Substantive Changes](#) for more information on substantive changes.

- Section: Section number in the updated document
- Description: Brief text describing the change
- Purpose: the reason for the change:
 - Clarity
 - Consistency

- Interoperability
- Perfunctory
- Readability
- Usability

Change Table

Table 2. Change Log

Source	Identifier	Type	Section	Description	Purpose
1	PR #48	S	8.8.3	GeoJSON literals in GeoSPARQL	Add support for a format that is very common for representing geospatial data on the Web.
2	PR #88	S	8.8.5	Discrete Global Grid System (DGGs) literals in GeoSPARQL	Add support for using a list of DGGs identifiers to describe a geometry.
7	PR #57	S	8.10	SPARQL functions for performing spatial aggregations of data.	To be able to work with sets of geometries.
33	PR #72	S	6.3	New properties <i>hasSize</i> , <i>hasLength</i> , <i>hasArea</i> , <i>hasVolume</i> and <i>hasSpatialResolution</i>	Extension of the ontology for indicating spatial resolution, areas, proportional relations (e.g. area overlaps).
69	PR #75	S	8.8.4	KML literals in GeoSPARQL	Add support for OGC KML for completeness.
5	PR #117	S	6.4.3, 6.4.4	New properties <i>hasCentroid</i> , <i>hasBoundingBox</i>	Useful for visualization, indexing and discovery.
	PR #140	A	Annex C	Simple Features for SQL mappings	Describe interoperability between functions and properties from Simple Features for SQL to GeoSPARQL
	PR #145	A	Annex B	Query example for min and max functions	Show how min and max work
	PR #146	S	8.9	Functions <i>isEmpty</i> , <i>isSimple</i> , <i>dimension</i> , <i>length</i> , <i>area</i> .	Extra functions for performing non-topological spatial operations
148	PR #150	S	6.2.3	Class SpatialObjectCollection	To support OGC API
148	PR #150	S	6.2.4	Class FeatureCollection	To support OGC API

Source	Identifier	Type	Section	Description	Purpose
148	PR #150	S	8.6.2	Class GeometryCollection	To support OGC API
	PR #151	S	8.9.15, 8.9.30	<code>geometryN</code> and <code>numGeometries</code> functions	To work with geometries in collections
142	PR #155	S	6.3,8.7	Metric properties	Convenience properties without the need to indicate units when working with metres.
164	PR #193	A	3, 4	Terms & definitions and normative references	To have direct access to explanations of terms that are used in the specification but not defined in the ontology.
204	PR #210	S	6.3.5, 6.3.6	Perimeter properties	To distinguish between length and perimeter of a thing.
	PR #216	A	6	Ontology overview figure	Understanding the spec
191	PR #221	A	Separate resource	JSON-LD context	Implementation of GeoSPARQL ontology in JSON-LD
113	PR #173	A	Separate resource	SHACL shapes	Validation
212	PR #225	A	B.1	Functions summary table	Better documentation of functions, inputs and outputs, and whether they apply specifically to 2d or 3d objects.
206	PR #226	A	8.9.1	Description of function behaviours	Clarification of behaviour of functions.
74	PR #234	A	Annex E	Alignments to other ontologies	Support common usage patterns for GeoSPARQL with external ontology elements, and provide guidance on how GeoSPARQL can be used with other, existing, spatial ontologies.
43	PR #240	A	Separate resource	German translation of ontology	For greater worldwide adoption.
	PR #245	A	8.4	Explanation of role of CRS in geometric computations	Clarification

Source	Identifier	Type	Section	Description	Purpose
259	PR #274	A	8.3	Recommendation to use QUDT for units of measure	Offers guidance on unit of measure vocabulary
273	PR #275	A	8.9.13	Clarification of distance function	Explanation of the unit argument.
247	PR #284	A	10.3.1	Improvement of examples	Documentation.
278	PR #300	A	8.2	Paragraph about GeoSPARQL and Simple Features Access - Common Architecture	Explain how GeoSPARQL is based on SFA-CA.
286	PR #324	S	8.9.5, 8.9.12	Metric buffer and distance functions	Functions without unit argument, to simplify implementation and use.

Description of Critical Changes

There are no critical changes in this release.

Description of Substantive Changes

Collection classes

GeoSPARQL 1.1 adds three classes for collection types:

- **SpatialObjectCollection**
- **FeatureCollection**
- **GeometryCollection**

These classes allow the organization of spatial objects, features and geometries in collections and are added to support the sorts of data models used by systems such as OGC APIs.

The collection class and the object class, such as **FeatureCollection** & **Feature** class, are related by the standard RDFS property **rdfs:member** whereby the collection class *has as a member* the object.

In RDF, an example of this for **FeatureCollection** / **Feature** may look like this:

NOTE

```
ex:col-1
  a geo:FeatureCollection ;
  rdfs:member ex:feature-a ;
  .
```

```
ex:feature-a
  a geo:Feature ;
  .
```

This is noted here to avoid confusion as to the direction of the object/collection relation.

New size properties

GeoSPARQL 1.1. includes several new scalar Spatial Object properties related to size for indicating spatial resolution, areas, and proportional relations (e.g. area overlaps):

- `hasSize`
- `hasLength`
- `hasPerimeterLength`
- `hasArea`
- `hasVolume`
- `hasSpatialResolution`

Note that a property `hasLength` was already present in GeoSPARQL 1.0; `hasPerimeterLength` was added to distinguish between length (e.g. the length of a football field) and perimeter (i.e. the added lengths of all sides, in the case of a polygon) of a thing.

These properties do not have tightly defined range values so that they may be used with any RDF measurement/metrology system although GeoSPARQL makes a recommendation to use [QUDT](#) for measurements and units of measure. Annex C exemplifies these properties with QUDT.

Metric properties

GeoSPARQL 1.1 also adds convenience properties for working with scalar values in metres as datatype properties, not value/unit properties as per QUDT standard use. For each `has...` property in the section above, there is a `hasMetric...` property with a datatype range specified to be in metres. For example `hasMetricArea` in `m^2` for `hasArea`.

New Feature properties

GeoSPARQL 1.1. includes two new properties `hasCentroid` and `hasBoundingBox`. These can be used to link a Feature with a point geometry corresponding with the centroid of its geometry, or with a simplified geometry-representation corresponding to the envelope of its geometry, respectively. The centroid is typically used to show location on a low-resolution image, and both centroid and bounding box are useful for indexing and discovery functions.

New geometry serializations

GeoSPARQL 1.1 includes support for representing geometry data in RDF in several additional formats:

- GeoJSON
- Keyhole Markup Language (KML)
- Discrete Global Grid System (DGGS)

As with the geometry serializations that were already supported in v1.0, for each new serialization an RDFS Datatype, a property and a function are defined:

- `geoJSONLiteral`
- `asGeoJSON`
- `asGeoJSON` function
- `kmlLiteral`
- `asKML`
- `asKML` function
- `dggsliteral`
- `asDGGS`
- `asDGGS` function

Extra functions for performing non-topological spatial operations

GeoSPARQL 1.1 adds several new functions for performing spatial operations:

- `isEmpty`
- `isSimple`
- `dimension`
- `length`
- `area`

Functions for geometry collections

GeoSPARQL 1.1 adds two functions for working with geometries in collections:

- `numGeometries`
- `geometryN`

These functions can be used to get the number of geometries in a collection, and to get the nth geometry in a collection.

Spatial aggregate functions

GeoSPARQL 1.1 defines a set of functions for performing spatial aggregations of data:

- `aggBoundingBox`
- `aggBoundingCircle`
- `aggCentroid`
- `aggConcaveHull`
- `aggConvexHull`
- `aggUnion`

These functions support working with sets of geometries, e.g. to calculate the minimum bounding box or centroid of a set of geometries.

Metric buffer and distance functions

To accompany metric properties, GeoSPARQL 1.1 also adds several functions that assume metres. These functions can be used without unit argument, and were added to simplify implementation and use.

The functions are:

- `metricBuffer`
- `metricDistance`

Future Work

As of the release of GeoSPARQL 1.1, the GeoSPARQL Standard Working Group proposes a next minor release of GeoSPARQL 1.2 in late 2022 or early 2023. The SWG is also considering a more major release, perhaps GeoSPARQL 2.0, after that. The intention is that backwards-compatible additions only will be added to GeoSPARQL 1.2, but potentially major, breaking, changes will be added to GeoSPARQL 2.0.

Determining scope for future GeoSPARQL releases is an open and public process undertaken by the SWG in the course of their regular meetings.

Release scope is shown by the SWG allocating Issues raised against the Standard in its online source code repository:

- <https://github.com/opengeospatial/ogc-geosparql/issues>

Milestones indicate Issue's potential release, for example the [Issue #301 "Handling of Z-coordinates by geometry functions is underspecified"](#) is allocated Milestone GeoSPARQL 1.2.

Appendix A: Revision History

Date	Release	Editor	Primary clauses modified	Description
2022-07-15	0.1	L. van den Brink	all	initial version
2022-07-21	0.2	N.J Car	all	minor formatting updates
2022-07-21	0.3	L. van den Brink	all	minor formatting updates
2022-07-27	0.4	N.J Car	all	minor formatting & template updates
2024-01-12	0.5	N.J Car	metadata	final published version numbers