

**OGC® DOCUMENT: 20-040R3**

External identifier of this OGC® document: <http://www.opengis.net/doc/AS/dggs/2.0>

**OGC®**  
Making location count.

# TOPIC 21 - DISCRETE GLOBAL GRID SYSTEMS - PART 1 CORE REFERENCE SYSTEM AND OPERATIONS AND EQUAL AREA EARTH REFERENCE SYSTEM

---

**ABSTRACT SPECIFICATION TOPIC**

**APPROVED**

**Version:** 2.0.0

**Submission Date:** 2020-12-07

**Approval Date:** 2021-02-15

**Publication Date:** 2021-09-23

**Editor:** Robert Gibb

**Notice:** This document is not an OGC Standard. This document is an OGC Abstract Specification Topic and is therefore not an official position of the OGC membership. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard.

Further, an OGC Abstract Specification Topic should not be referenced as required or mandatory technology in procurements.

## License Agreement

Permission is hereby granted by the Open Geospatial Consortium, (“Licensor”), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD. THE INTELLECTUAL PROPERTY IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER’S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR’s sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications. This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable.

## Copyright notice

Copyright © 2021 Open Geospatial Consortium  
To obtain additional rights of use, visit <http://www.ogc.org/legal/>

## Note

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

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

# CONTENTS

I. ABSTRACT .....	ix
II. KEYWORDS .....	ix
III. PREFACE .....	x
IV. SECURITY CONSIDERATIONS .....	xi
V. SUBMITTING ORGANIZATIONS .....	xii
VI. SUBMITTERS .....	xii
VII. FOREWORD .....	xiii
VIII. INTRODUCTION .....	xiii
1. SCOPE .....	2
2. CONFORMANCE .....	4
3. NORMATIVE REFERENCES .....	7
4. TERMS AND DEFINITIONS .....	9
5. CONVENTIONS .....	25
5.1. Abbreviated terms .....	25
5.2. Universal Resource Identifiers .....	26
5.3. Unified Modelling Language notation .....	26
5.4. Naming conventions .....	26
5.5. Attribute and association role status .....	27
6. DGGs SPECIFICATION OVERVIEW .....	29
6.1. Package overview .....	29
7. COMMON SPATIO-TEMPORAL CLASSES PACKAGE .....	34
7.1. Common Spatio-temporal Classes overview .....	34
7.2. Temporal and Zonal Geometry package .....	35
7.3. Temporal and Zonal RS using Identifiers package .....	51
8. DGGs CORE REFERENCE SYSTEM AND FUNCTIONS PACKAGE .....	70
8.1. DGGs Core package conformance classes .....	70

8.2. Core RS using Zonal Identifiers with Structured Geometry module .....	70
8.3. DGGs Core functions .....	81
9. DGGs EQUAL AREA EARTH REFERENCE SYSTEM PACKAGE .....	109
9.1. Equal-Area Earth DGGs package .....	109
ANNEX A (NORMATIVE) CONFORMANCE CLASS ABSTRACT TEST SUITE .....	134
A.1. Common Spatio-temporal Classes package conformance categories .....	134
A.2. DGGs Core package conformance categories .....	137
A.3. Equal-Area Earth DGGs package conformance categories .....	143
ANNEX B (INFORMATIVE) EQUAL AREA DGGs THEORY .....	150
ANNEX C (INFORMATIVE) BACKGROUND TO DGGs .....	158
C.1. DGGs as a digital information medium .....	158
C.2. History .....	158
C.3. Global grid taxonomy .....	159
C.4. Criterion .....	161
C.5. A digital spatial reference system .....	162
C.6. Application .....	162
ANNEX D (INFORMATIVE) TEMPORAL GEOMETRY, TOPOLOGY, AND TEMPORAL REFERENCING BY NAMED PERIODS – CONTEXT FOR MODELLING .....	165
D.1. General .....	165
D.2. Geometry and topology .....	165
D.3. Referencing by named periods .....	166
ANNEX E (INFORMATIVE) REQUIREMENTS CROSSWALK FROM TOPIC 21 V1.0 TO V2.0 .....	168
ANNEX F (NORMATIVE) REVISION HISTORY .....	173
BIBLIOGRAPHY .....	180

## LIST OF TABLES

---

Table 1 – Conformance classes for Common Spatio-temporal Classes package .....	4
Table 2 – Conformance classes for DGGs Core package .....	4
Table 3 – Categories of conformance for Equal-Area Earth DGGs package .....	5
Table 4 – Module names used in UML diagrams in this document .....	31
Table 5 – Elements of Temporal Geometry and Topology::Duration .....	38
Table 6 – Elements of Temporal Geometry and Topology::EdgeT .....	39
Table 7 – Elements of Temporal Geometry and Topology::Instant .....	39



Table 8 – Elements of Temporal Geometry and Topology::Interval .....	39
Table 9 – Elements of Temporal Geometry and Topology::NodeT .....	40
Table 10 – Elements of Temporal Geometry and Topology::TemporalGeometricCollection .....	40
Table 11 – Elements of Temporal Geometry and Topology::TemporalGeometricComplex .....	41
Table 12 – Elements of Temporal Geometry and Topology::TemporalGeometricPrimitive .....	42
Table 13 – Elements of Temporal Geometry and Topology::TemporalGeometry .....	42
Table 14 – Elements of Temporal Geometry and Topology::TemporalTopologicalComplex .....	43
Table 15 – Elements of Temporal Geometry and Topology::TemporalTopologicalPrimitive .....	44
Table 16 – Elements of Temporal Geometry and Topology::TemporalTopology .....	44
Table 17 – Elements of Zonal Geometry and Topology::ZoneCompoundGeometry .....	47
Table 18 – Elements of Zonal Geometry and Topology::ZoneCompoundTopology .....	47
Table 19 – Elements of Zonal Geometry and Topology::ZoneGeometry .....	48
Table 20 – Elements of Zonal Geometry and Topology::ZoneSimpleGeometry .....	49
Table 21 – Elements of Zonal Geometry and Topology::ZoneSimpleTopology .....	49
Table 22 – Elements of Zonal Geometry and Topology::ZoneTopology .....	50
Table 23 – Elements of Spatial Location::LocationS .....	52
Table 24 – Elements of Temporal RS using Identifiers::Period .....	55
Table 25 – Elements of Temporal RS using Identifiers::PeriodClass .....	56
Table 26 – Elements of Temporal RS using Identifiers::PeriodIdentifier .....	56
Table 27 – Elements of Temporal RS using Identifiers::TemporalReferenceSystemUsingPeriodIdentifiers .....	57
Table 28 – Elements of Zonal RS using Identifiers::RSUsingZonalIdentifiers .....	62
Table 29 – Elements of Zonal RS using Identifiers::RSUsingZonalIdentifiersCompound .....	62
Table 30 – Elements of Zonal RS using Identifiers::RSUsingZonalIdentifiersPrimitive .....	63
Table 31 – Elements of Zonal RS using Identifiers::ZonalIdentifier .....	63
Table 32 – Elements of Zonal RS using Identifiers::ZonalIdentifierComplex .....	64
Table 33 – Elements of Zonal RS using Identifiers::ZonalIdentifierPrimitive .....	64
Table 34 – Elements of Zonal RS using Identifiers::Zone .....	65
Table 35 – Elements of Zonal RS using Identifiers::ZoneClass .....	66
Table 36 – Elements of Zonal RS using Identifiers::ZoneClassComplex .....	66
Table 37 – Elements of Zonal RS using Identifiers::ZoneClassPrimitive .....	67
Table 38 – Elements of Zonal RS using Identifiers::ZoneCompound .....	67
Table 39 – Elements of Zonal RS using Identifiers::ZonePrimitive .....	68
Table 40 – Elements of Core RS using Zonal Identifiers with Structured Geometry::Cell .....	73
Table 41 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG_ReferenceSystem .....	73
Table 42 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG_ReferenceSystemType .....	74
Table 43 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DiscreteGlobalGrid .....	75
Table 44 – Elements of Core RS using Zonal Identifiers with Structured Geometry::GlobeGeometry .....	76

Table 45 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG_GridConstraint .....	76
Table 46 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG_RefinementStrategy .....	77
Table 47 – Elements of Core Quantization Functions::DataAssignmentProcess .....	84
Table 48 – Elements of Core Quantization Functions::DGG_Observation .....	84
Table 49 – Elements of Core Quantization Functions::QuantizationContext .....	85
Table 50 – Elements of Core Quantization Functions::QuantizationFunctions .....	85
Table 51 – Elements of Core Quantization Functions::Quantization .....	86
Table 52 – Elements of Core Topological Query Functions::refinementLevelRange .....	93
Table 53 – Elements of Core Topological Query Functions::ZoneQuery .....	94
Table 54 – Elements of Core Topological Query Functions::RelativePosition enumeration .....	96
Table 55 – Elements of Interoperation::InteroperationFunctions .....	99
Table 56 – Elements of Interoperation Query::QueryComponents .....	101
Table 57 – Elements of Interoperation Query::QueryFunctions .....	102
Table 58 – Elements of Interoperation Query::QueryResult .....	102
Table 59 – Elements of Interoperation Query::QueryType .....	103
Table 60 – Elements of Interoperation Broadcast::BroadcastFunctions .....	105
Table 61 – Elements of Interoperation Broadcast::BroadcastType .....	105
Table 62 – Elements of Interoperation Broadcast::TranslationType .....	106
Table 63 – Elements of Equal-Area Earth RS::EA_DGG_ReferenceSystem .....	113
Table 64 – Elements of Equal-Area Earth RS::EA_GlobeGeometry .....	113
Table 65 – Elements of Equal-Area Earth RS::EA_ReferenceSystemType .....	114
Table 66 – Elements of Equal-Area Earth RS::EA_SurfaceInterpolation class .....	115
Table 67 – Elements of Equal-Area Tessellation::EA_BaseUnitPolyhedron .....	118
Table 68 – Elements of Equal-Area Tessellation::EA_DiscreteGlobalGrid .....	119
Table 69 – Elements of Equal-Area Tessellation::EA_DiscreteGlobalGridTessellation .....	120
Table 70 – Elements of Equal-Area Tessellation::EA_InitialDiscreteGlobalGrid .....	120
Table 71 – Elements of Equal-Area Tessellation::EA_PolyhedralTessellation .....	121
Table 72 – Elements of Equal-Area Tessellation::EA_RefinedDiscreteGlobalGrid .....	122
Table 73 – Elements of Equal-Area Cell::EA_BoundaryData .....	127
Table 74 – Elements of Equal-Area Cell::EA_Cell .....	128
Table 75 – Elements of Equal-Area Cell::EA_Zone .....	129
Table 76 – Elements of Equal-Area Cell::EA_BoundaryType .....	131
Table A.1 – Conformance classes for Common Spatio-temporal Classes package .....	134
Table A.2 – Conformance classes for DGGs Core package .....	137
Table A.3 – Categories of conformance for Equal-Area Earth DGGs package .....	143
Table B.1 .....	151
Table B.2 .....	154
Table B.3 .....	155
Table C.1 .....	160
Table C.2 .....	162

Table E.1 – Requirements crosswalk from Topic 21 v1.0 to v2.0 .....	168
Table E.2 – Requirements crosswalk from Topic 21 v2.0 to v1.0 .....	170
Table F.1 .....	173

## LIST OF FIGURES

---

Figure 1 – Referencing by identifiers with structured geometry .....	xv
Figure 2 – DGGs Package Diagram .....	30
Figure 3 – Primitives of Temporal Geometry and Topology .....	36
Figure 4 – ISO 19107:2019 Context for Temporal Geometry and Topology .....	37
Figure 5 – Components of Zonal Geometry and Topology module .....	46
Figure 6 – Context for LocationS .....	51
Figure 7 – Context for Temporal RS using Identifiers module .....	54
Figure 8 – Primitives of ZonalIdentifier .....	58
Figure 9 – Primitives of Zone .....	59
Figure 10 – Primitives of ZoneClass .....	59
Figure 11 – Primitives of RSUsingZonalIdentifiers .....	60
Figure 12 – Components of RSUsingZonalIdentifiers .....	61
Figure 13 – Core RS using Zonal Identifiers with Structured Geometry module .....	71
Figure 14 – Components of the Core Quantization Functions module .....	83
Figure 15 – Examples of parent, child, and sibling query operations .....	90
Figure 16 – Components of Topological Zonal Query Functions module .....	92
Figure 17 – Components of Core Interoperation Functions module .....	98
Figure 18 – Components of Interoperation Query module .....	100
Figure 19 – Interoperation Broadcast .....	104
Figure 20 – Components of Equal-Area Earth RS module .....	110
Figure 21 – Components of Equal-Area Tessellation module .....	111
Figure 22 – Components of Equal-Area Cell module .....	112
Figure B.1 – Regular polyhedra (top) and their corresponding initial equal area tessellation (bottom) .....	151
Figure B.2 – Comparison of a grid (in this case radial) represented on both (a) curved and (b) planar surfaces. ....	152
Figure B.3 – Planar square grid with nested child cells. ....	153
Figure B.4 – Geometries and spatial properties of cells on curved surfaces. ....	154
Figure B.5 – Tessellations of the Earth to equal-area cells. ....	155
Figure B.6 – Using Morton space filling curve for defining. ....	156
Figure C.1 – Global Grid Taxonomy (Based on Figure 4 in [2]). ....	160
Figure C.2 – Examples of DGGs based on the mapping of the faces of Platonic solids to the surface model of the Earth .....	161

# LIST OF RECOMMENDATIONS

---

- REQUIREMENT 1 ..... 45
- REQUIREMENT 2 ..... 50
- REQUIREMENT 3 ..... 53
- REQUIREMENT 4 ..... 57
- REQUIREMENT 5 ..... 68
- REQUIREMENT 6 ..... 78
- REQUIREMENT 7 ..... 78
- REQUIREMENT 8 ..... 79
- REQUIREMENT 9 ..... 79
- REQUIREMENT 10 ..... 79
- REQUIREMENT 11 ..... 80
- REQUIREMENT 12 ..... 80
- REQUIREMENT 13 ..... 80
- REQUIREMENT 14 ..... 81
- REQUIREMENT 15 ..... 81
- REQUIREMENT 16 ..... 87
- REQUIREMENT 17 ..... 97
- REQUIREMENT 18 ..... 104
- REQUIREMENT 19 ..... 107
- REQUIREMENT 20 ..... 116
- REQUIREMENT 21 ..... 116
- REQUIREMENT 22 ..... 117
- REQUIREMENT 23 ..... 117
- REQUIREMENT 24 ..... 117
- REQUIREMENT 25 ..... 118
- REQUIREMENT 26 ..... 124
- REQUIREMENT 27 ..... 124
- REQUIREMENT 28 ..... 126
- REQUIREMENT 29 ..... 126



## ABSTRACT

---

This Abstract Specification lays the foundations for Discrete Global Grid Systems (DGGs). It defines *Common classes for spatio-temporal geometry, topology, and reference systems using identifiers*, a *DGGs Core Reference system* as a reference system using zonal identifiers with structured geometry that may be spatio-temporal, a suite of *DGGs Core Functions*, and it specifies Equal-Area Earth DGGs. The OGC DGGs Abstract Specification supports the specification of standardized DGGs infrastructures that enable the integrated analysis of very large, multi-source, multi-resolution, multi-dimensional, distributed geospatial data. Interoperability between OGC DGGs implementations is anticipated through implementation standards, and extension interface encodings of OGC Web Services.



## KEYWORDS

---

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

ogcdoc, OGC document, Discrete Global Grid System, DGGs, Digital Earth, DGGs-core, Spatial Reference System, Global Data Structure, Geographic Information Systems, DE-9IM, standard, specification





## PREFACE

---

This document is consistent with the first edition ISO 19170-1:2020, Geographic Information – Discrete Global Grid Systems Specifications – Core Reference System and Operations, and Equal Area Earth Reference System. ISO/DIS 19170-1:2020 was prepared by Technical Committee ISO/TC 211, Geographic information/Geomatics, in close collaboration with the Open Geospatial Consortium (OGC). It replaces the first OGC edition published as document 15-104r5.

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.*



## SECURITY CONSIDERATIONS

---

No security considerations have been made for this document.

V

## SUBMITTING ORGANIZATIONS

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

- Manaaki Whenua - Landcare Research, New Zealand
- Geoscience Australia, Joint Research Centre (JRC), Pangaea Innovations Pty Ltd
- Peking University Collaborative Innovation Center for Geospatial Big Data
- University of Calgary

VI

## SUBMITTERS

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

.

Name	Representing	OGC member
Joseph Bell	Geoscience Australia	Yes
Chenqi Cheng	Peking University Collaborative Innovation Center for Geospatial Big Data	Yes
Robert Gibb (ed.)	Manaaki Whenua – Landcare Research, New Zealand	Yes
Perry Peterson	University of Calgary	Yes
Matthew Purss	Pangaea Innovations Pty Ltd	Yes
Fuhu Ren	Peking University Collaborative Innovation Center for Geospatial Big Data	Yes
Faramarz Samavati	University of Calgary	Yes
Peter Strobl	Joint Research Centre (JRC)	Yes

This second OGC edition of Abstract Specification Topic 21 – Discrete Global Grid Systems replaces the first edition (OGC 15-104r5), which has been technically revised.

The changes in this edition compared to the previous OGC edition are:

- separation into DGGs Core and a specification for Equal Area Earth DGGs;
- explicit handling of Dynamic vs Static datums for Equal Area Earth DGGs;
- remodelling of the Core to incorporate new editions of ISO 19107, ISO 19111 & ISO 19112;
- remodelling of the Core reference system to more explicitly include metadata elements;
- extension of the Core to support up to three spatial dimensions;
- extension of the Core to support up to one temporal dimension;
- extension of the Core to support DGGs reference systems with conformal, or axis aligned cells;
- additional modelling of temporal geometry and temporal reference systems sufficient for spatio-temporal DGGs, introducing the concept of a zone as a region of space-time equivalent to a location or an era;
- change in name from ‘DGGs Reference Frame’ to ‘Referencing using zonal identifiers with structured geometry’, due to the inclusion of non-spatial geometry, reference systems, spatio-temporal zones and the new capabilities in ISO 19107:2019, ISO 19111:2019 & ISO 19112:2019.
- remodelling of algebraic functions, to extensions of ISO 19107 Query, now referred to as topological query functions

Further details are given in Annex D and Annex E.

DGGs (Discrete Global Grid Systems) provide a new way to organise, store and analyse spatio-temporal data. This document contains a normative definition for DGGs and informative annexes. Annex B discusses the theoretical basis for equal area Earth DGGs, and Annex C discusses DGGs’s historical background. At the heart of DGGs is a new Referencing system.

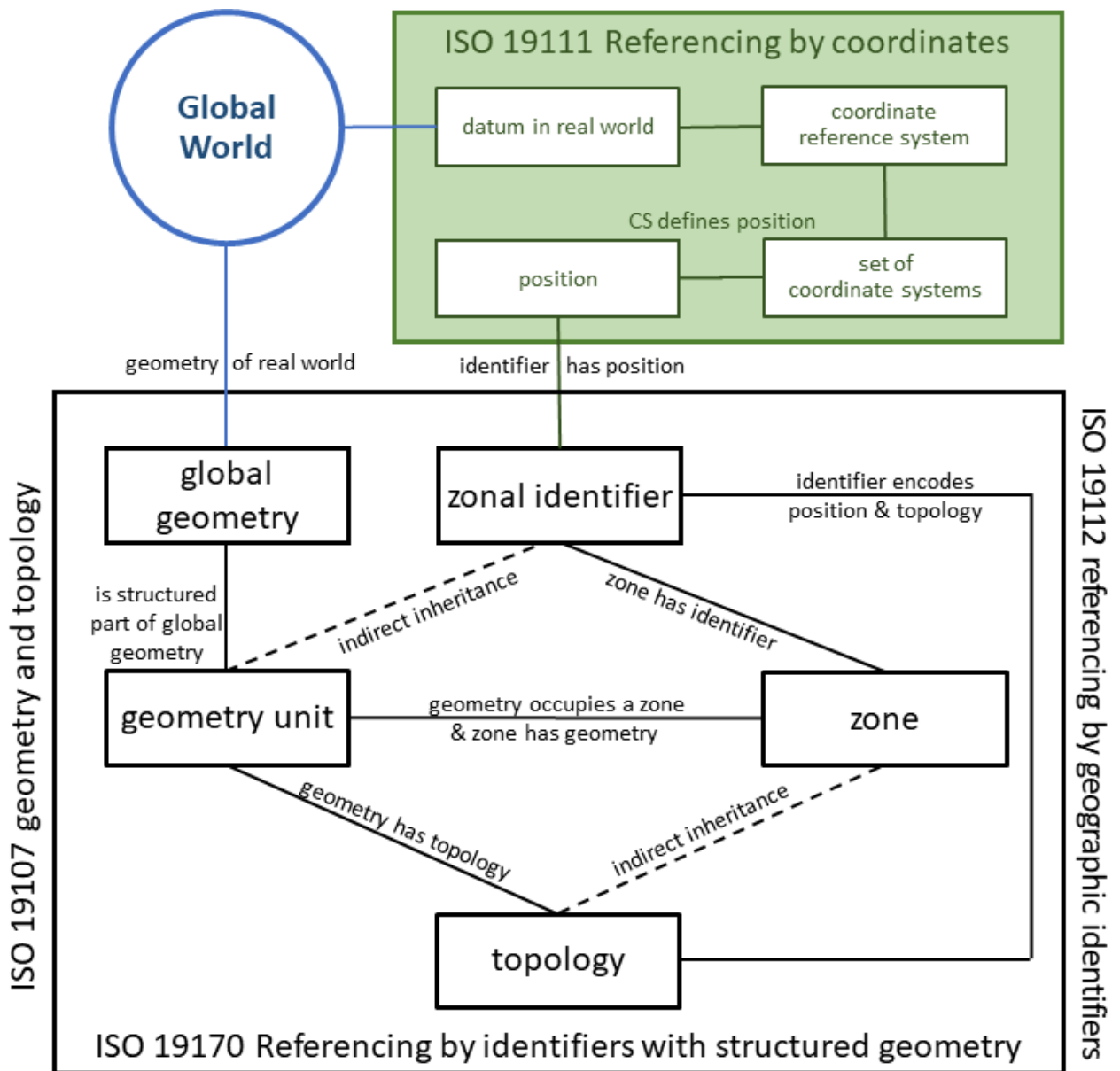
Spatial and temporal RSs described elsewhere by ISO/TC211 and the OGC (Open Geospatial Consortium) fall into two types:

1. Referencing by coordinates (ISO 19111:2019), and
2. Referencing by identifiers (geographic in ISO 19112:2019 & ordinal era in ISO 19108:2002).

In spatial Referencing by identifiers, the only required geometry is an extent, which may be expressed as a simple bounding box. Formal geometry need not be defined and sometimes follows societal whim. Similarly in ordinal temporal RSs, the topology of ordinal era's are known, but the start and finish times are often only an estimation and are not required by the data model. DGGs introduce a third type – Referencing by identifiers with structured geometry, illustrated in Figure 1. A single parent global geometry is chosen to define the dimensionality and orientation of the region of space-time occupied by the DGG – it's global world.



Figure 1 – Referencing by identifiers with structured geometry



The structure for the DGGs geometry is provided by a strictly controlled process of recursive tessellation of the parent geometry that creates the DGGs RS's units of geometry. The region occupied by each unit of geometry is called a zone. Each zone is given a unique name, called a zonal identifier. Each zonal identifier is associated with a representative spatio-temporal position in a base CRS (coordinate reference system) defined by a datum for the DGGs's global world. Best practice is for a zonal identifier to be an encoding of both its position and its topology. Referencing by identifiers with structured geometry gives rise to RSs using zonal identifiers with structured geometry. Geographic information is inherently four-dimensional and includes time. So, a unified spatio-temporal data model for coordinate systems, geometry, topology, identifiers and RSs using identifiers is a pre-requisite for spatio-temporal DGGs.

The approach taken in this document to specifying spatio-temporal data classes is to apply the spatio-temporal data model pattern in ISO 19111:2019 to spatial data classes in both ISO

19107:2019 and ISO 19112:2019 to produce their spatio-temporal equivalents. The set of common spatio-temporal classes for geometry, topology, identifiers, and RSs using identifiers specified in this document are therefore consistent with spatio-temporal CRS and coordinate systems in ISO 19111. Like ISO 19111, the temporal data model in this document does not reference ISO 19108:2002. The similarities and differences are described in Annex D.

In this document the spatio-temporal scope is constrained to spatial classes that are invariant through all time, and to temporal classes that are invariant throughout space. While this approach excludes certain spatio-temporal situations, it is flexible enough for a very large body of social and environmental modelling. Oceanic, climate and weather modelling often need geometries with a constant mass of gaseous fluid under changing pressure and temperature. These models can be run outside a DGGS. However, the results coming from these environmental models can be stored in a DGGS for efficient later use with other data.

This document specifies data models for a consistent set of common spatio-temporal classes, a DGGS core built on the common spatio-temporal classes, and a DGGS EAERS (equal-area Earth RS). The Common Spatio-temporal Classes, DGGS Core, and Equal-Area Earth DGGS packages each have their own conformance classes with their associated specifications and requirements.

The DGGS Core package is comprised of a RS, and functions for quantization, topological query and interoperability.

The DGGS Core RS is a RS using zonal identifiers with structured geometry located in its real world by coordinates in a base CRS. The DGGS Core RS is designed to support:

- temporal, surface, volumetric and spatio-temporal DGGS,
- DGGS with different grid constraints,
- DGGS with different refinement strategies, and
- DGGS referencing either the Earth or other celestial bodies.

The RS in Equal-Area Earth DGGS is a specialization of the DGGS Core RS. It describes a RS, comprising:

- a base unit polyhedron,
- a discrete hierarchical sequence of global grids,
- global grids with equal-area zones each with a unique identifier, and
- located in a geodetic CRS, that is typically also a geographic CRS.

This document does not prescribe any specific Earth surface model, base polyhedron or class of polyhedra, but is intended to allow for a range of options that produce DGGSs with compatible and interoperable functional characteristics.

Future additions to the OGC Topic 21 / ISO 19170 series are intended to cover:

- Part 2 – Three-dimensional and equal-volume Earth RS.

- Part 3 – Spatio-temporal Earth RS.
- Part 4 – Axis-aligned RS with all zone edges parallel to the base CRS's axes.
- Specification for a DGGs-API to formalise client-server, and server-server operations, both between DGGs and between DGGs and non-DGGs architectures.
- Creation of a register system for DGGs definitions analogous to the register for CRSs.
- Additions to other specifications, such as for OWS OGC 05-042r2, OGC 17-089r1 architectures, spatial features and data formats to support DGGs data structures.

1

# SCOPE

---

This document supports the definition of:

- A DGGs core comprising:
  - a RS using zonal identifiers with structured geometry, and
  - functions providing import, export and topological query,
- Common spatio-temporal classes for geometry, topology, RS using zonal identifiers, zonal identifiers and zones, based on ISO 19111 CRS. The spatio-temporal scope is constrained to
  - spatial elements that are invariant through all time, and
  - temporal elements that are invariant across all space.
- EAERS for equal-area Earth DGGs.



2

# CONFORMANCE

---

## CONFORMANCE

This standard defines the conformance classes listed in tables Table 1–Table 3

**Table 1 – Conformance classes for Common Spatio-temporal Classes package**

Conformance class	Requirement tests
Temporal geometry and topology	Annex A.1, Requirement test A.1
Temporal reference systems using period identifiers	Annex A.1, Requirement test A.4
Spatial zone geometry and topology	Annex A.1, Requirement test A.2
Spatial reference systems using zonal identifiers	Annex A.1, Requirement test A.3, & Annex A.1, Requirement test A.5
Spatio-temporal zone geometry and topology	Annex A.1, Requirement test A.1, & Annex A.1, Requirement test A.2
Spatio-temporal reference systems using zonal identifiers	Annex A.1, Requirement test A.3, Annex A.1, Requirement test A.4, & Annex A.1, Requirement test A.5

**Table 2 – Conformance classes for DGGs Core package**

Conformance class	Requirement tests
Spatial DGGs Core	Annex A.1, Requirement test A.2, Annex A.1, Requirement test A.3, Annex A.1, Requirement test A.5, and all Core requirements in Annex A.2.1, Requirement test A.6–Annex A.2.2, Requirement test A.19.
Spatio-temporal DGGs Core	Annex A.1, Requirement test A.1, Annex A.1,

Requirement test A.2, Annex A.1, Requirement test A.3, Annex A.1, Requirement test A.4, Annex A.1, Requirement test A.5, and all Core requirement tests in Annex A.2. 1, Requirement test A.6–Annex A.2.2, Requirement test A.19.

**Table 3 – Categories of conformance for Equal-Area Earth DGGs package**

Conformance class	Requirement tests
Equal-Area Earth DGGs	Annex A.1, Requirement test A.2, Annex A.1, Requirement test A.3, & Annex A.1, Requirement test A.5, and all Core requirement tests in Annex A.2. 1, Requirement test A.6–Annex A.2.2, Requirement test A.19, and all EAERS requirement tests in Annex A.3, Requirement test A.20–Annex A.3, Requirement test A.29.

Conformance with this standard shall be checked using 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.

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



3

# NORMATIVE REFERENCES

---

## NORMATIVE REFERENCES

---

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

ISO: ISO 19103:2015, *Geographic information – Conceptual schema language*. International Organization for Standardization, Geneva (2015). <https://www.iso.org/standard/56734.html>

ISO: ISO 19107:2019, *Geographic information – Spatial schema*. International Organization for Standardization, Geneva (2019). <https://www.iso.org/standard/66175.html>

The following pair of OGC and ISO documents are identical in normative content:

Roger Lott: OGC 18-005r5, *Topic 2 – Referencing by coordinates Corrigendum*. Open Geospatial Consortium (2021). <https://docs.ogc.org/as/18-005r5/18-005r5.html>

ISO: ISO 19111:2019, *Geographic information – Referencing by coordinates*. International Organization for Standardization, Geneva (2019). <https://www.iso.org/standard/74039.html>

ISO: ISO 19112:2019, *Geographic information – Spatial referencing by geographic identifiers*. International Organization for Standardization, Geneva (2019). <https://www.iso.org/standard/70742.html>

The following pair of OGC and ISO documents are identical in normative content:

ISO: OGC 11-111r1, *Topic 11 – Metadata*. Open Geospatial Consortium (2016). [http://www.iso.org/iso/home/store/catalogue\\_ics/catalogue\\_detail\\_ics.htm?csnumber=53798](http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=53798)

ISO: ISO 19115-1:2014, *Geographic information – Metadata – Part 1: Fundamentals*. International Organization for Standardization, Geneva (2014). <https://www.iso.org/standard/53798.html>

The following pair of OGC and ISO documents are identical in normative content:

Simon Cox: OGC 10-004r3, *Topic 20 – Observations and Measurements*. Open Geospatial Consortium (2010). [https://portal.ogc.org/files/?artifact\\_id=41579](https://portal.ogc.org/files/?artifact_id=41579)

ISO: ISO 19156:2011, *Geographic information – Observations and measurements*. International Organization for Standardization, Geneva (2011). <https://www.iso.org/standard/32574.html>





4

# TERMS AND DEFINITIONS

---

## TERMS AND DEFINITIONS

---

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

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

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

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

### 4.1. boundary

---

set that represents the limit of an entity

**Note 1 to entry:** Boundary is most commonly used in the context of geometry, where the set is a collection of points or a collection of objects that represent those points. In other domains, the term is used metaphorically to describe the transition between an entity and the rest of its domain of discourse.

**Source:** ISO 19107:2019, Clause 3.6

### 4.2. cell

---

<DGGS> spatial, spatio-temporal or temporal unit of geometry with dimension greater than 0, associated with a *zone* (Clause 4.52)

**Note 1 to entry:** All cells within a DGGS (Clause 4.13) share the dimensionality of the DGGS’s parent *global* (Clause 4.24) geometry. DGGS with dimensionality of 0, are not supported.

**Note 2 to entry:** Cells are the unit of geometry in a DGGS, and the geometry of the region of space-time occupied by a zone is a cell.

**Note 3 to entry:** While the terms cell and zone are often used interchangeably, strictly zone is the preferred term. Cell is entirely appropriate when specifically discussing a zone's geometry or topology.

## 4.3. cell refinement

---

<DGGS> process of subdividing *parent cells* (Clause 4.33) into descendant *child cells* (Clause 4.4) using a specified *refinement ratio* (Clause 4.38) and suite of refinement strategies

**Note 1 to entry:** Iterative application of cell refinements creates a hierarchy of descendant *discrete global grids* (Clause 4.12).

**Note 2 to entry:** Cell refinement methods may result in child cells that each have a single parent or that have multiple parents.

## 4.4. child cell    child ADMITTED

---

<DGGS> immediate descendant of a *parent cell* (Clause 4.33)

**Note 1 to entry:** Child cells are either within a single parent cell or overlapped by multiple parent cells

## 4.5. class

---

description of a set of objects that share the same attributes, operations, methods, relationships, and semantics

**Note 1 to entry:** A class may use a set of interfaces to specify collections of operations it provides to its environment. The term was first used in this way in the general theory of object-oriented programming, and later adopted for use in this same sense in UML.

**Source:** ISO 19103:2015, Clause 4.7 , modified – Note 1 to entry has been added from clause 4.2

## 4.6. compound coordinate reference system compound CRS ADMITTED

---

*coordinate reference system* (Clause 4.7) using at least two independent coordinate reference systems

**Note 1 to entry:** Coordinate reference systems are independent of each other if coordinate values in one cannot be converted or transformed into coordinate values in the other.

**Source:** ISO 19111:2019, Clause 3.1.3

## 4.7. coordinate reference system CRS ADMITTED

---

coordinate system that is related to an object by a *datum* (Clause 4.10)

**Note 1 to entry:** Geodetic and vertical datums are referred to as reference frames.

**Note 2 to entry:** For geodetic and vertical datums, the object will be the Earth. In planetary applications, geodetic and vertical reference frames may be applied to other celestial bodies.

**Source:** ISO 19111:2019, Clause 3.1.9

## 4.8. coordinate system

---

set of mathematical rules for specifying how coordinates are to be assigned to points

**Source:** ISO 19111:2019, Clause 3.1.11

## 4.9. data type

---

specification of a *value* (Clause 4.49) domain with operations allowed on values in this domain

**Note 1 to entry:** Data types include primitive predefined types and user-definable types. All instances of a data type lack identity.

Example Integer, Real, Boolean, String, Date (conversion of a date into a series of codes).

**Source:** ISO 19103:2015, Clause 4.14 , modified – Note 1 to entry has been added from ISO 19156:2011, Clause 4.3

## 4.10. datum reference frame ADMITTED

---

parameter or set of parameters that realize the positions of the origin, the scale, and the orientation of a *coordinate system* (Clause 4.8)

**Source:** ISO 19111:2019, Clause 3.1.15

## 4.11. datum ensemble

---

group of multiple realizations of the same terrestrial or vertical reference system that, for approximate spatial referencing purposes, are not significantly different

**Note 1 to entry:** Datasets referenced to the different realizations within a datum ensemble may be merged without coordinate transformation.

**Note 2 to entry:** ‘Approximate’ is for users to define but typically is in the order of under 1 decimetre but may be up to 2 metres.

Example “WGS 84” as an undifferentiated group of realizations including WGS 84 (TRANSIT), WGS 84 (G730), WGS 84 (G873), WGS 84 (G1150), WGS 84 (G1674) and WGS 84 (G1762). At the surface of the Earth these have changed on average by 0.7 m between the TRANSIT and G730 realizations, a further 0.2 m between G730 and G873, 0.06 m between G873 and G1150, 0.2 m between G1150 and G1674 and 0.02 m between G1674 and G1762).

**Source:** ISO 19111:2019, Clause 3.1.16

## 4.12. discrete global grid

---

<DGGS> set of *cells* (Clause 4.2) at the same *refinement level* (Clause 4.37), that uniquely and completely cover a *globe* (Clause 4.24)

**Note 1 to entry:** the set of cell *zonal identifiers* (Clause 4.50) comprising a discrete global grid form a single Zone Class with its associated refinement level.

**Note 2 to entry:** the configuration of the set of cells comprising a discrete global grid satisfy at least one grid constraint in the DGG\_GridConstraint codelist.

## 4.13. discrete global grid system **DGGS** ADMITTED

---

integrated system comprising a *hierarchy* (Clause 4.26) of *discrete global grids* (Clause 4.12), *spatio-temporal referencing* (Clause 4.42) by *zonal identifiers* (Clause 4.50) and functions for *quantization* (Clause 4.36), *zonal query* (Clause 4.51), and *interoperability* (Clause 4.28)

## 4.14. duration

---

non-negative quantity of time equal to the difference between the final and initial *instants* (Clause 4.29) of a time *interval* (Clause 4.30)

**Note 1 to entry:** The duration is one of the base quantities in the International System of Quantities (ISQ) on which the International System of Units (SI) is based. The term “time” instead of “duration” is often used in this context and also for an infinitesimal duration.

**Note 2 to entry:** For the term “duration”, expressions such as “time” or “time interval” are often used, but the term “time” is not recommended in this sense and the term “time interval” is deprecated in this sense to avoid confusion with the concept of “time interval”.

**Note 3 to entry:** The exact duration of a time scale unit depends on the time scale used. For example, the durations of a year, month, week, day, hour or minute, may depend on when they occur [in a Gregorian calendar, a calendar month can have a duration of 28, 29, 30, or 31 days; in a 24-hour clock, a clock minute can have a duration of 59, 60, or 61 seconds, etc.]. Therefore, the exact duration can only be evaluated if the exact duration of each is known.

**Note 4 to entry:** This definition is closely related to NOTE 1 of the terminological entry “duration” in IEC 60050-113:2011, 113-01-13.

**Source:** ISO 8601-1:2019, Clause 3.1.1.8

## 4.15. dynamic coordinate reference system **dynamic CRS** ADMITTED

---

*coordinate reference system* (Clause 4.7) that has a *dynamic reference frame* (Clause 4.16)

**Note 1 to entry:** Coordinates of points on or near the crust of the Earth that are referenced to a dynamic coordinate reference system may change with time, usually due to crustal deformations such as tectonic motion and glacial isostatic adjustment.

**Note 2 to entry:** Metadata for a dataset referenced to a dynamic coordinate reference system should include coordinate epoch information.

**Source:** ISO 19111:2019, Clause 3.1.19

## 4.16. dynamic reference frame    dynamic datum ADMITTED

---

*reference frame* (Clause 4.10) in which the defining parameters include time evolution

**Note 1 to entry:** The defining parameters that have time evolution are usually a coordinate set.

**Source:** ISO 19111:2019, Clause 3.1.20

## 4.17. error budget

---

<metric> statement of or methodology for describing the nature and magnitude of the errors which affect the results of a calculation

**Source:** ISO 19107:2019, Clause 3.35 , modified — Note 1 to entry has been removed

## 4.18. feature

---

abstraction of real-world phenomena

**Note 1 to entry:** A feature can occur as a type or an instance. In this document, feature instance is meant unless otherwise specified.

**Source:** ISO 19101-1:2014, Clause 4.1.11 , modified — Note 1 to entry has been added from ISO 19156:2011, Clause 4.6, and modified

## 4.19. feature type

---

*class* (Clause 4.5) of *features* (Clause 4.18) having common characteristics

**Source:** ISO 19156:2011, Clause 4.7

## 4.20. geodetic coordinate reference system geodetic CRS ADMITTED

---

three-dimensional *coordinate reference system* (Clause 4.7) based on a geodetic reference frame and having either a three-dimensional Cartesian or a spherical coordinate system

**Note 1 to entry:** In this document a CRS based on a geodetic reference frame and having an ellipsoidal coordinate system is geographic.

**Source:** ISO 19111:2019, Clause 3.1.31

## 4.21. geographic coordinate reference system geographic CRS ADMITTED

---

coordinate reference system (Clause 4.7) that has a geodetic reference frame and an ellipsoidal coordinate system

**Source:** ISO 19111:2019, Clause 3.1.35

## 4.22. geographic identifier

---

*spatial reference* (Clause 4.41) in the form of a label or code that identifies a *location* (Clause 4.31)

**Example** “Spain” is an example of a label (country name); “SW1P 3AD” is an example of a code (postcode).

**Source:** ISO 19112:2019, Clause 3.1.2



## 4.23. geometric primitive

---

geometric object representing a single, connected, homogeneous (isotropic) element of space

**Note 1 to entry:** Geometric primitives are non-decomposed objects that present information about geometric configuration. They include points, curves, surfaces, and solids. Many geometric objects behave like primitives (supporting the same interfaces defined for geometric primitives) but are actually composites composed of some number of other primitives. General collections may be aggregates and incapable of acting like a primitive (such as the lines of a complex network, which is not connected and thus incapable of being traceable as a single line). By this definition, a geometric primitive is topological open, since the *boundary* (Clause 4.1) points are not isotropic to the interior points. Geometry is assumed to be closed. For points, the boundary is empty.

**Source:** ISO 19107:2019, Clause 3.50

## 4.24. globe global ADMITTED

---

<DGGS> region of space-time enclosing a celestial body

**Note 1 to entry:** In this document globe is used in its most general form to refer to any celestial body or region of space-time enclosing a celestial body that may be referenced by a *DGGS* (Clause 4.13). When a specific body, such as the Earth is referred to, an explicit term is used.

## 4.25. grid

---

network composed of two or more sets of curves in which the members of each set intersect the members of the other sets in an algorithmic way

**Note 1 to entry:** The curves partition a space into grid cells.

**Source:** ISO 19123:2005, Clause 4.1.23

## 4.26. hierarchy

---

<DGGS> organization and ranking of successive levels of *cell refinement* (Clause 4.3) of *discrete global grids* (Clause 4.12)

## 4.27. initial discrete global grid

---

<DGGS> discrete global grid tessellation created by circumscribing a defined path along the chosen surface model of the Earth between the vertices of the scaled base unit polyhedron

## 4.28. interoperability

---

capability to communicate, execute programmes, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units

**Note 1 to entry:** in this document interoperability specifically refers to functions that initiate and process transfers of data from a DGGS (Clause 4.13).

**Source:** ISO/IEC 2382:2015, Clause 2121317 , modified – The original domain and Notes to Entry have been deleted. A Note 1 to entry has been added.

## 4.29. instant

---

<DGGS> temporal geometry primitive representing a point in time

**Note 1 to entry:** On *temporal coordinate systems* (Clause 4.46) as specified in ISO 19107:2019, the temporal *geometric primitives* (Clause 4.23) *instant* and *interval* (Clause 4.30) are the equivalent of points and lines as specified in ISO 19107:2019.

## 4.30. interval

---

<DGGS> temporal geometry primitive representing a line in time

**Note 1 to entry:** On *temporal coordinate systems* (Clause 4.46) as specified in (ISO 19107:2019), the temporal *geometric primitives* (Clause 4.23) *instant* (Clause 4.29) and *interval* are the equivalent of points and lines as specified in ISO 19107:2019.

## 4.31. location

---

particular place or position

**Note 1 to entry:** A location identifies a geographic place.

**Note 2 to entry:** In the context of *DGGS* (Clause 4.13), locations have dimension greater than one, and so are not points.

Example "Madrid", "SW1P 3AD".

**Source:** ISO 19112:2019, Clause 3.1.3 , modified – Note two has been added and an additional example provided

## 4.32. observation

---

act of measuring or otherwise determining the *value* (Clause 4.49) of a property

**Source:** ISO 19156:2011, Clause 4.11

## 4.33. parent cell    parent ADMITTED

---

<DGGS> *cell* (Clause 4.2) in a higher refinement level of discrete global grid with immediate descendants

**Note 1 to entry:** parent cells either overlap or contain their *child cells* (Clause 4.4).

## 4.34. period

---

<DGGS> particular era or span of time

**Note 1 to entry:** Periods are *intervals* (Clause 4.30) named with a *period identifier* (Clause 4.35)

## 4.35. period identifier

---

<DGGS> temporal reference in the form of a label or code that identifies a *period* (Clause 4.34)

**Note 1 to entry:** Period identifiers are the temporal equivalent of *geographic identifiers* (Clause 4.22) as specified in ISO 19112:2019

## 4.36. quantization

---

<DGGS> function assigning data from external sources to cell values

## 4.37. refinement level

---

<DGGS> numerical order of a *discrete global grid* (Clause 4.12) in the tessellation sequence

**Note 1 to entry:** The tessellation with the smallest number of cells has a refinement level = 0.

## 4.38. refinement ratio

---

<DGGS> ratio of the number of *child cells* (Clause 4.4) to *parent cells* (Clause 4.33)

**Note 1 to entry:** A positive integer ratio  $n$  refinement of DGGS (Clause 4.13) parent cells yield  $n$  times as many child cells as parent cells.

**Note 2 to entry:** For a two-dimensional DGGS (as defined for EAERS in this document) this is the surface area ratio.

**Note 3 to entry:** In DGGS literature [2] the term aperture has been used instead of refinement ratio. Refinement ratio is preferred because it is clearer in meaning to audiences outside the early DGGS community.

## 4.39. sibling cell sibling ADMITTED

---

<DGGS> *cell* (Clause 4.2) in a discrete global grid with the same *parent cell* (Clause 4.33)

**Note 1 to entry:** all the *child cells* (Clause 4.4) of a parent cell are each-others' sibling cells.

## 4.40. simple

---

<topology, geometry> homogeneous (all points have isomorphic neighbourhoods) and with a simple *boundary* (Clause 4.1)

**Note 1 to entry:** The interior is everywhere locally isomorphic to an open disc in a Euclidean coordinate space of the appropriate dimension  $D^n = \{P \mid \|P\| < 1.0\}$ . The boundary is a dimension one smaller. This essentially means that the object does not intersect nor touch itself. Generally used for a curve that does not cross not touch itself with the possible exception of boundary points. Simple closed curves are isomorphic to a circle.

**Source:** ISO 19107:2019, Clause 3.84

## 4.41. spatial reference

---

description of position in the real world

**Note 1 to entry:** This may take the form of a label, code or coordinate tuple.

**Source:** ISO 19111:2019, Clause 3.1.56

## 4.42. spatio-temporal reference

---

system for identifying position in the real world that may include time

**Note 1 to entry:** This may take the form of a label, code or coordinate tuple.

## 4.43. spatio-temporal coordinate reference system **spatio-temporal CRS** ADMITTED

---

*compound coordinate reference system* (Clause 4.6) in which one constituent *coordinate reference system* (Clause 4.7) is a spatial coordinate reference system and one is a *temporal coordinate reference system* (Clause 4.47)

**Source:** ISO 19111:2019, Clause 3.1.59

## 4.44. static coordinate reference system **static CRS** ADMITTED

---

*coordinate reference system* (Clause 4.7) that has a *static reference frame* (Clause 4.45)

**Note 1 to entry:** Coordinates of points on or near the crust of the Earth that are referenced to a dynamic coordinate reference system do not change with time.

**Note 2 to entry:** Metadata for a dataset referenced to a static coordinate reference system does not require coordinate epoch information.

**Source:** ISO 19111:2019, Clause 3.1.61

## 4.45. static reference frame **static datum** ADMITTED

---

*reference frame* (Clause 4.10) in which the defining parameters exclude time evolution

**Source:** ISO 19111:2019, Clause 3.1.62

## 4.46. temporal coordinate system

---

<geodesy> one-dimensional coordinate system where the axis is time

**Source:** ISO 19111:2019, Clause 3.1.64

## 4.47. temporal coordinate reference system temporal CRS ADMITTED

---

coordinate reference system (Clause 4.7) based on a temporal datum

**Source:** ISO 19111:2019, Clause 3.1.63

## 4.48. tessellation

---

partitioning of a space into a set of conterminous subspaces having the same dimension as the space being partitioned

**Note 1 to entry:** A tessellation composed of congruent regular polygons or polyhedra is a regular tessellation. One composed of regular, but non-congruent polygons or polyhedra is a semi-regular tessellation. Otherwise the tessellation is irregular. Tessellations on curved surfaces cannot be congruent, so all tessellations in DGGs (Clause 4.13) are either semi-regular or irregular.

**Example** Graphic examples of tessellations may be found in Figures 11, 13, 20, and 22 of [ISO19123].

**Source:** ISO 19123:2005, Clause 4.1.39 , modified – Note 1 to entry has been modified and new notes to entry have been added.

## 4.49. value

---

element of a type domain

**Note 1 to entry:** A value considers a possible state of an object within a *class* (Clause 4.5) or type (domain).

**Note 2 to entry:** A data value is an instance of a datatype, a value without identity.

**Note 3 to entry:** A value can use one of a variety of scales including nominal, ordinal, ratio and interval, spatial and temporal. Primitive datatypes can be combined to form aggregate datatypes with aggregate values, including vectors, tensors and images.

**Source:** ISO 19156:2011, Clause 4.18

## 4.50. zonal identifier

---

<DGGS> *spatio-temporal reference* (Clause 4.42) in the form of a label or code that identifies a *zone* (Clause 4.52)

**Note 1 to entry:** A zonal identifier may be a *geographic identifier* (Clause 4.22), *period identifier* (Clause 4.35), or a compound of the two.

**Note 2 to entry:** A zone's *ZonalIdentifier* provides the coordinates of a representative position for the zone, and *spatio-temporal feature* (Clause 4.18) geometry is represented by sets of *ZonalIdentifiers*.

## 4.51. zonal query

---

<DGGS> geometry or topology function using a cell's *zonal identifiers* (Clause 4.50) to specify geometry

**Note 1 to entry:** ISO 19107:2019 specifies a suite of geometry and topology functions in the *Query2D* and *Query3D* classes, where geometry elements used in each function's parameters are described by sets of coordinates. In DGGS (Clause 4.13) all geometry can be referenced as sets of *cells* (Clause 4.2) represented solely by a list (or set) of their zonal identifiers. This document specifies *ZoneQuery* to implement the operations in both *Query2D* and *Query3D* using zonal identifiers to reference each operation's source and target geometry.

## 4.52. zone

---

<DGGS> particular region of space-time

**Note 1 to entry:** The primitives of zone are *spatial location* (Clause 4.31) and *temporal period* (Clause 4.34).

**Note 2 to entry:** A zone may be either a single zonal primitive or a compound zone comprising one spatial location and one temporal period. Zones can be regions of space-time associated with any celestial body.

**Note 3 to entry:** Zones are the primary container for storing and retrieving data within a DGGS implementation. DGGSs reference zones by their *zonal identifier* (Clause 4.50), for instance in databases or through tile nomenclature.

**Note 4 to entry:** Each zone's geometry is represented by a cell (Clause 4.2).





5

# CONVENTIONS

---

## 5.1. ABBREVIATED TERMS

DE-9IM	Dimensionally Extended 9-Intersection Model
EAERS	equal-area Earth RS
EC	earth-centered
ECEF	earth-centered earth fixed
GEM	Geodesic Elevation Model
GIS	geographic information system
GUID	globally unique identifier
HPC	high-performance computing
HPD	high-performance data
ICT	information and communications technology
ISEA	Icosahedral Snyder Equal Area
ISEA3H	Icosahedral Snyder Equal Area Aperture 3 Hexagon
ISO	International Organization for Standardization
OGC	Open Geospatial Consortium
OWS	OGC Web-Service
QTM	Quaternary Triangular Mesh
rHEALPix	rearranged Hierarchical Equal Area isoLatitude Pixelization
RS	reference system
UML	Unified Modeling Language
URI	Uniform Resource Identifier

## 5.2. UNIVERSAL RESOURCE IDENTIFIERS

---

The normative provisions in this specification are denoted by the URI:

<http://www.opengis.net/spec/dggs/2.0>

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

## 5.3. UNIFIED MODELLING LANGUAGE NOTATION

---

In this document, the conceptual schema for describing DGGs are presented in the UML. ISO 19103:2015 presents the specific profile of UML used in this document.

The UML diagrams in this document refer to classifiers in five other standards. Each standard has been assigned a colour that is used consistently across all UML diagrams. Each diagram has a key listing the standards referred to in that diagram and their colours. Interface names in the figure have the structure <module-name>::<interface-name>. For reference, Table 1 lists all the module names and the standard they belong to. Both colour and module name can be used as quick reference to a classifier's standard.

## 5.4. NAMING CONVENTIONS

---

Where possible, when a classifier represents the common behaviour of a set of defined things from the terms defined in Clause 3, the UML classifier will generally use the defined terms as its name. Since classifier names are capitalized and contain no space, and the defined term may contain several words, the classifier name will separate words using upper-camel-case concatenations (no spaces but each word beginning with a capital with all other letters in lowercase). Similarly, the name may be some simplified key phrase. This “UpperCamelCase” rule is generally followed but may be violated if clarity or consistency with other standards is improved by minor violations. For example:

- Zone identifier values are represented by the interface `ZonalIdentifier` or stored using the datatype `DirectPosition` defined in ISO 19107:2019.
- Instances of primitives will realize the interface `Primitive` in the package `CommonSpatio-temporalClasses` and other interfaces for their specific dimension and interpolation mechanism. For example, `Point`, `Instant`, `Interval`, `Line`, `NodeT`, `LocationS`.

- Any classifier name referenced from another standard retains its original format.

Classifier names for attributes and operations in the UML models may similarly use key phrases in lowerCamelCase (same as UpperCamelCase, but the first word begins in a lowercase letter). For example: parent; child; parentOf; childOf and relatePosition are all used as operation names.

Module and package names can contain spaces. In some situations, a phrase that has an abbreviation is used in its unabbreviated form as a package name. Where a package or module is referred to in the text, both the capitalisation and unabbreviated form are preserved. This distinguishes them from a phrase in general use. For example: DGGs Equal-Area Earth Reference System; Zone and Temporal Geometry; and Reference System defined in this standard and Coordinate Reference Systems defined in ISO 19111:2019.

In summary, the use of capitals for a term in the general text indicates a reference to a classifier from the UML

## 5.5. ATTRIBUTE AND ASSOCIATION ROLE STATUS

---

In this document, conceptual schema in Clauses 6–8 are defined by tables. In these tables:

- attributes and association roles are given an **Obligation** status:
  - **M**: mandatory – this attribute or association role shall be supplied.
  - **C**: conditional – this attribute or association role shall be supplied if the condition (given in the description) is true. It may be supplied if the condition is false.
  - **O**: optional – this attribute or association role may be supplied.
- the **Maximum Occurrence** column in the tables indicates the maximum number of occurrences of attribute values that are permissible, with \* indicating no upper limit.
- non-navigable associations are not included in the UML diagrams or tables.

The tables provide a summary of the UML diagrams. In particular, association roles, attributes, operations, and constraints that are inherited from another class unchanged are not described in the tables. In the event of any discrepancies between the UML diagrams and text, the UML shall prevail.

6

# DGGS SPECIFICATION OVERVIEW

---

## 6.1. PACKAGE OVERVIEW

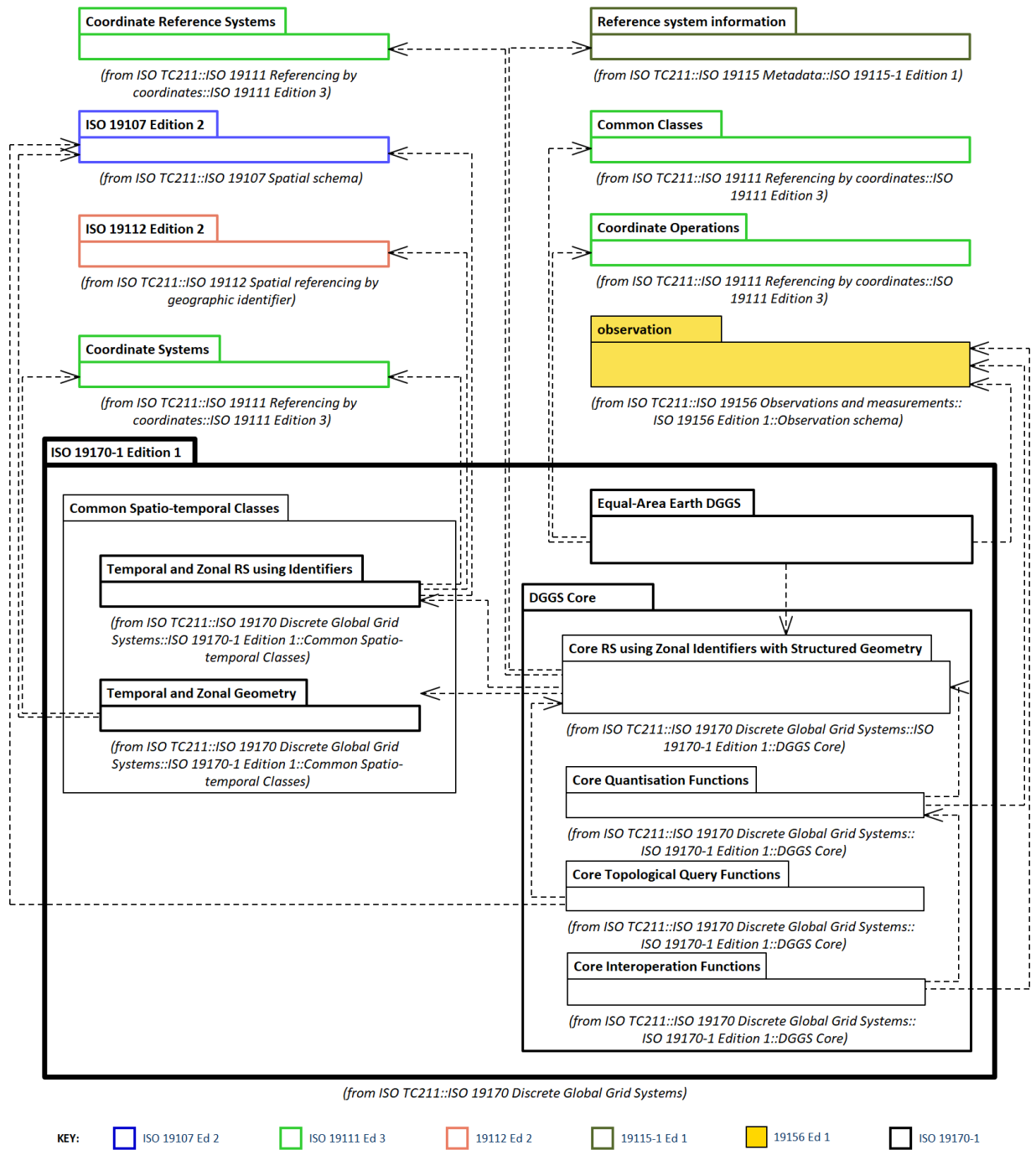
---

The specification for DGGSs is described in this document in the form of a UML model with supplementary defining tables and text. The UML data model is organised into three primary packages. The Common Spatio-temporal Classes package contains two UML sub-packages, DGGS Core contains four UML sub-packages, and DGGS Equal-Area Earth Reference System package is a single UML package, as shown in Figure 2.

- Common Spatio-temporal Classes package contains
  - Zone and Temporal Geometry package, comprising temporal and spatio-temporal primitives for geometry and topology,
  - Zone, Identifier and RS package, comprising temporal and spatio-temporal RS using identifiers and their primitives.
- DGGS Core package contains:
  - Core RS using zonal identifiers with structured geometry package,
  - Core Quantisation Functions package,
  - Core Query Functions package,
  - Core Interoperation Functions package.
- Equal-Area Earth DGGS package.

In Figure 2, each box represents a package and contains the package name. Each arrowed line shows the normative dependency of one package upon another package (at the head of the arrow). The Common Spatio-temporal Classes, DGGS Core and Equal-Area Earth DGGS packages dependent on packages in five other ISO standards.

Figure 2 – DGGS Package Diagram

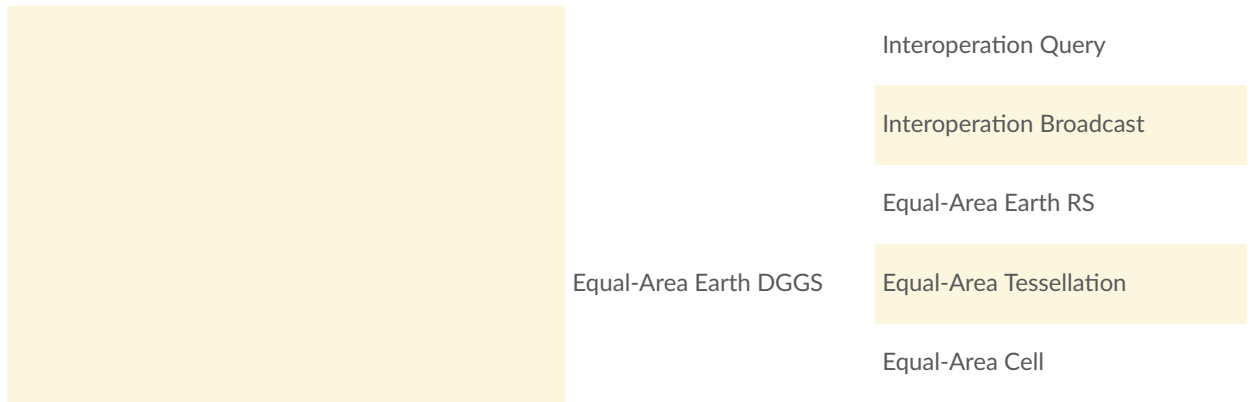


Packages are grouped in a hierarchy of sub-packages, with modules containing interfaces at the leaves of the hierarchy. In the UML diagrams that follow, interface names are often shown as <module-name>::<interface-name>. For reference, Table 4 lists all the module names that are referred in the diagrams and names the standard they come from.

**Table 4 – Module names used in UML diagrams in this document**

Standard name	Module name
ISO 19107:2019 Spatial Schema Ed 2	Edge
	Geometry
	Node
	Topology
	Common Classes
	Coordinates
	ISO 19111:2019 Referencing by Coordinates Ed 3
Coordinate Reference System	
Coordinate Systems	
ISO 19112:2019 Spatial referencing by geographic identifier Ed 2	ISO 19112 Edition 2
ISO 19115-1:2014 Metadata Ed 1	Reference system information
ISO 19156:2011 Observation and Measurements Ed 1	Observation
ISO 19170-1 Discrete Global Grid Systems Ed 1	Temporal Geometry and Topology
	Zonal Geometry and Topology
	Temporal RS using Identifiers
	Spatial Location
	Zonal RS using Identifiers
DGGs Core	Core RS using Zonal Identifiers with Structured Geometry
	Core Quantization Functions
	Core Topological Query Functions
	Core Interoperation Functions





Conformance classes for the modules in the Common Spatio-temporal Classes, DGGs Core, and Equal-Area Earth DGGs packages are described in Annex A.

One product conformance class is also defined for an Equal-Area Earth DGGs product, that brings modules together as a system.

7

# COMMON SPATIO-TEMPORAL CLASSES PACKAGE

---

# COMMON SPATIO-TEMPORAL CLASSES PACKAGE

---

## 7.1. COMMON SPATIO-TEMPORAL CLASSES OVERVIEW

---

This clause specifies the common spatio-temporal classes to support temporal and spatio-temporal geometry, topology, zones, zonal identifiers, zonal query, and RSs using temporal or zonal identifiers.

These classes are defined here in such a way that they can be used in any context which requires an internally consistent set of temporal and spatio-temporal classes for use with spatial classes from ISO 19107, 19111 and 19112. They are further specialised in the DGGs Core for use in DGGs. In this restricted model for spatio-temporal systems, the spatio-temporal scope is constrained to spatial classes that are invariant through all time, and to temporal classes that are invariant throughout space.

The Common Spatio-temporal Classes are organized into two packages with five modules:

- a) Temporal and Zonal Geometry package comprises:
  1. Temporal Geometry and Topology module
  2. Zonal Geometry and Topology module
  
- b) RS using Temporal and Zonal Identifiers package comprises:
  1. Spatial Location module
  2. Temporal RS using Identifiers module
  3. Zonal RS using Identifiers module

In each package separate spatial and temporal classes are defined first, followed by spatio-temporal classes that bring temporal and spatial classes together.

## 7.2. TEMPORAL AND ZONAL GEOMETRY PACKAGE

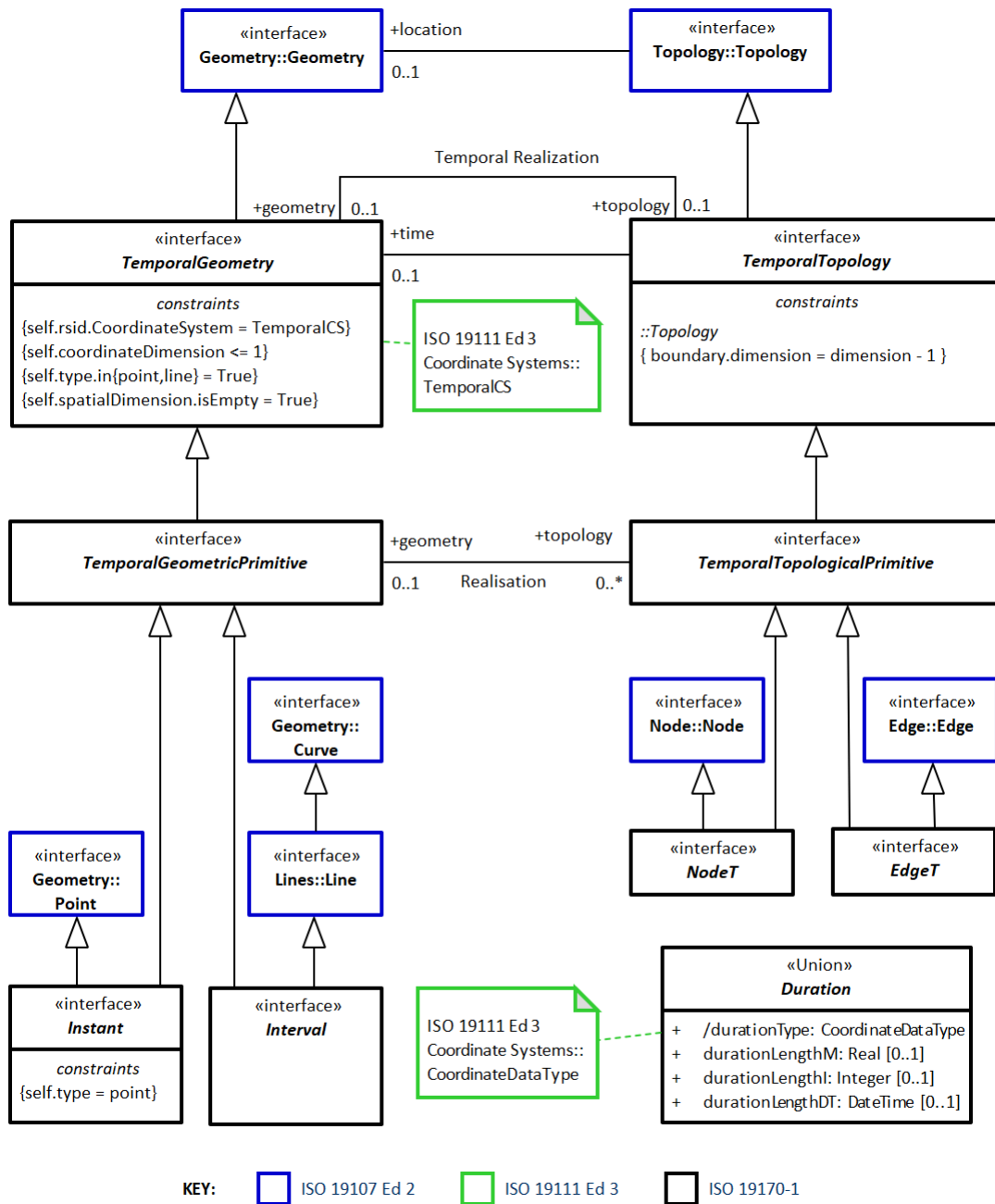
---

### 7.2.1. Temporal Geometry and Topology module

#### 7.2.1.1. Context and data model

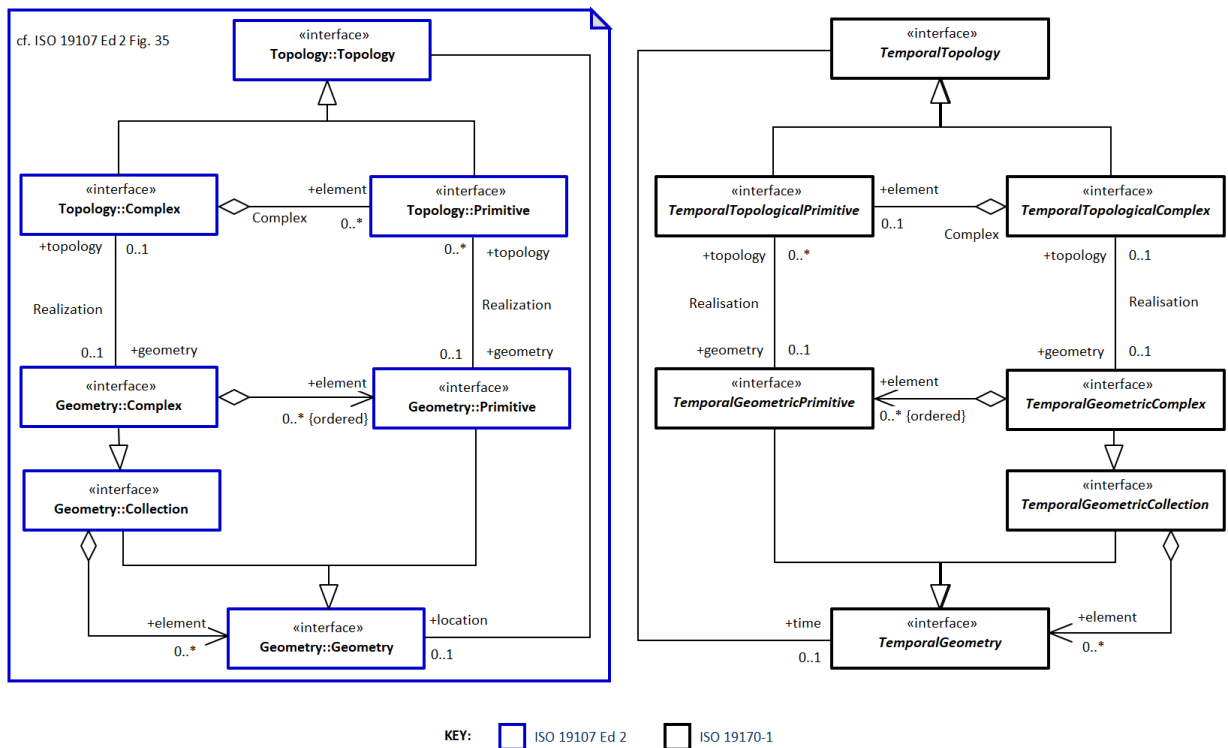
Temporal geometry is geometry constrained to one of the temporal coordinate systems, defined by TemporalCS in ISO 19111:2019. Temporal geometry primitives instant and interval implement the temporal analogues of point and line respectively in ISO 19107:2019. All geometry has topology, and the temporal topology primitives nodeT and edgeT implement the temporal analogues of node and edge respectively. These are shown in Figure 3.

Figure 3 – Primitives of Temporal Geometry and Topology



Temporal interfaces paralleling the spatial geometry and topology interface structure are built from these temporal primitives. Each temporal interface has the same meaning and semantics as their equivalent spatial interface, with the constraint that all temporal interfaces are constrained to the same coordinate system as their temporal primitives. Figure 4 shows the spatial classes on the left and their temporal equivalents on the right.

Figure 4 – ISO 19107:2019 Context for Temporal Geometry and Topology



### 7.2.1.2. Defining tables

- a) Table 5 – Elements of Temporal Geometry and Topology::Duration
- b) Table 6 – Elements of Temporal Geometry and Topology::EdgeT
- c) Table 7 – Elements of Temporal Geometry and Topology::Instant
- d) Table 8 – Elements of Temporal Geometry and Topology::Interval
- e) Table 9 – Elements of Temporal Geometry and Topology::NodeT
- f) Table 10 – Elements of Temporal Geometry and Topology::TemporalGeometricCollection
- g) Table 11 – Elements of Temporal Geometry and Topology::TemporalGeometricComplex
- h) Table 12 – Elements of Temporal Geometry and Topology::TemporalGeometricPrimitive
- i) Table 13 – Elements of Temporal Geometry and Topology::TemporalGeometry

- j) Table 14 – Elements of Temporal Geometry and Topology::TemporalTopologicalComplex
- k) Table 15 – Elements of Temporal Geometry and Topology::TemporalTopologicalPrimitive
- l) Table 16 – Elements of Temporal Geometry and Topology::TemporalTopology

**Table 5 – Elements of Temporal Geometry and Topology::Duration**

**Name:** Duration

**Definition:** Duration implements Length on a Temporal Coordinate System.

**Stereotype:** Union

**Abstract:** true

**Associations:** (none)

	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
<b>Public attributes:</b>	durationLengthDT	Length of time for Duration of type Date Time.		C	1	DateTime
	durationLengthI	Length of time for Duration of type Integer count.		C	1	Integer
	durationLengthM	Length of time for Duration of type Real measure.		C	1	Real
	durationType	Type of unit of measure for time.	true	M	1	CoordinateDataType (code list)

**Constraints:** (none)

**Table 6 – Elements of Temporal Geometry and Topology::EdgeT**

<b>Name:</b>	EdgeT
<b>Definition:</b>	Topological temporal edge, one-dimensional topological primitive.
<b>Stereotype:</b>	Interface
<b>Inheritance from:</b>	Edge, TemporalTopologicalPrimitive
<b>Abstract:</b>	true
<b>Associations:</b>	(none)
<b>Public attributes:</b>	(none)
<b>Constraints:</b>	(none)

**Table 7 – Elements of Temporal Geometry and Topology::Instant**

<b>Name:</b>	Instant
<b>Definition:</b>	Instant implements Point geometry on a Temporal Coordinate System.
<b>Stereotype:</b>	Interface
<b>Inheritance from:</b>	TemporalGeometricPrimitive, Point
<b>Abstract:</b>	true
<b>Associations:</b>	(none)
<b>Public attributes:</b>	(none)
<b>Constraints:</b>	self.type = point

**Table 8 – Elements of Temporal Geometry and Topology::Interval**

<b>Name:</b>	Interval
--------------	----------



<b>Definition:</b>	Interval implements Line geometry on a Temporal Coordinate System.
<b>Stereotype:</b>	Interface
<b>Inheritance from:</b>	Line, TemporalGeometricPrimitive
<b>Abstract:</b>	true
<b>Associations:</b>	(none)
<b>Public attributes:</b>	(none)
<b>Constraints:</b>	(none)

**Table 9 – Elements of Temporal Geometry and Topology::NodeT**

<b>Name:</b>	NodeT
<b>Definition:</b>	Topological temporal node, zero-dimensional topological primitive, its boundary being empty.
<b>Stereotype:</b>	Interface
<b>Inheritance from:</b>	Node, TemporalTopologicalPrimitive
<b>Abstract:</b>	true
<b>Associations:</b>	(none)
<b>Public attributes:</b>	(none)
<b>Constraints:</b>	(none)

**Table 10 – Elements of Temporal Geometry and Topology::TemporalGeometricCollection**

<b>Name:</b>	TemporalGeometricCollection
<b>Definition:</b>	Temporal Geometric Collection implements geometric Collection for Temporal Geometry.
<b>Stereotype:</b>	Interface

<b>Inheritance from:</b>	TemporalGeometry			
<b>Generalization of:</b>	TemporalGeometricComplex			
<b>Abstract:</b>	true			
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	TemporalGeometry (feature type)	C	*	element
<b>Public attributes:</b>	(none)			
<b>Constraints:</b>	(none)			

**Table 11 – Elements of Temporal Geometry and Topology::TemporalGeometricComplex**

<b>Name:</b>	TemporalGeometricComplex			
<b>Definition:</b>	Temporal Geometric Complex implements geometric Complex for Temporal Geometry.			
<b>Stereotype:</b>	Interface			
<b>Inheritance from:</b>	TemporalGeometricCollection			
<b>Abstract:</b>	true			
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	TemporalGeometricPrimitive (feature type)	C	*	element
	TM_GeometricPrimitive (feature type)	C	*	element
	TemporalTopologicalComplex (feature type)	C	1	topology
<b>Public attributes:</b>	(none)			
<b>Constraints:</b>	(none)			

**Table 12 – Elements of Temporal Geometry and Topology::TemporalGeometricPrimitive**

<b>Name:</b>	TemporalGeometricPrimitive				
<b>Definition:</b>	Temporal Geometric Primitive implements geometric Primitive for Temporal Geometry.				
<b>Stereotype:</b>	Interface				
<b>Inheritance from:</b>	TemporalGeometry, Primitive				
<b>Generalization of:</b>	Instant, Interval				
<b>Abstract:</b>	true				
<b>Associations:</b>	<i>Association with:</i>		<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	TemporalTopologicalPrimitive (feature type)		C	*	topology
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>
	spatialDimension	Dimension of its spatial geometry component.		M	1
	temporalDimension	Dimension of its temporal geometry component.		M	1
<b>Constraints:</b>	(none)				

**Table 13 – Elements of Temporal Geometry and Topology::TemporalGeometry**

<b>Name:</b>	TemporalGeometry				
<b>Definition:</b>	Temporal Geometry implements 1D Geometry on a Temporal Coordinate System.				
<b>Stereotype:</b>	Interface				
<b>Inheritance from:</b>	Geometry, ZoneSimpleGeometry				
<b>Generalization of:</b>	TemporalGeometricCollection, TemporalGeometricPrimitive				
<b>Abstract:</b>	true				

<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	TemporalTopology (feature type)	C	1	topology
<b>Public attributes:</b>	(none)			
<b>Constraints:</b>	self.coordinateDimension <= 1			
	self.rsid.CoordinateSystem = TemporalCS			
	self.spatialDimension.isEmpty = True			
	self.type.in{point,line} = True			

**Table 14 – Elements of Temporal Geometry and Topology::TemporalTopologicalComplex**

<b>Name:</b>	TemporalTopologicalComplex			
<b>Definition:</b>	Temporal Topological Complex implements topological Complex for Temporal Topology.			
<b>Stereotype:</b>	Interface			
<b>Inheritance from:</b>	TemporalTopology			
<b>Abstract:</b>	true			
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	TemporalTopologicalPrimitive (feature type)	C	1	element
	TemporalGeometricComplex (feature type)	C	1	geometry
<b>Public attributes:</b>	(none)			
<b>Constraints:</b>	(none)			

**Table 15 – Elements of Temporal Geometry and Topology::TemporalTopologicalPrimitive**

<b>Name:</b>	TemporalTopologicalPrimitive			
<b>Definition:</b>	Temporal Topological Primitive implements topological Primitive for Temporal Topology.			
<b>Stereotype:</b>	Interface			
<b>Inheritance from:</b>	TemporalTopology			
<b>Generalization of:</b>	EdgeT, NodeT			
<b>Abstract:</b>	true			
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	TemporalGeometricPrimitive (feature type)	C	1	geometry
<b>Public attributes:</b>	(none)			
<b>Constraints:</b>	(none)			

**Table 16 – Elements of Temporal Geometry and Topology::TemporalTopology**

<b>Name:</b>	TemporalTopology			
<b>Definition:</b>	Temporal Topology implements 1D Topology for Temporal Geometry.			
<b>Stereotype:</b>	Interface			
<b>Inheritance from:</b>	Topology, ZoneSimpleTopology			
<b>Generalization of:</b>	TemporalTopologicalComplex, TemporalTopologicalPrimitive			
<b>Abstract:</b>	true			
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	TemporalGeometry (feature type)	C	1	geometry
	TemporalGeometry (feature type)	C	1	time

Public attributes: (none)

Constraints: (none)

The following requirement applies:

### REQUIREMENT 1

[www.opengis.net/spec/DGGS/2.0/req/cc/temporal/geometry](http://www.opengis.net/spec/DGGS/2.0/req/cc/temporal/geometry)

---

*The common classes for temporal geometry and topology shall conform to the data model in Figure 3 and Figure 4 and defining tables in Table 5–Table 16. .*

---

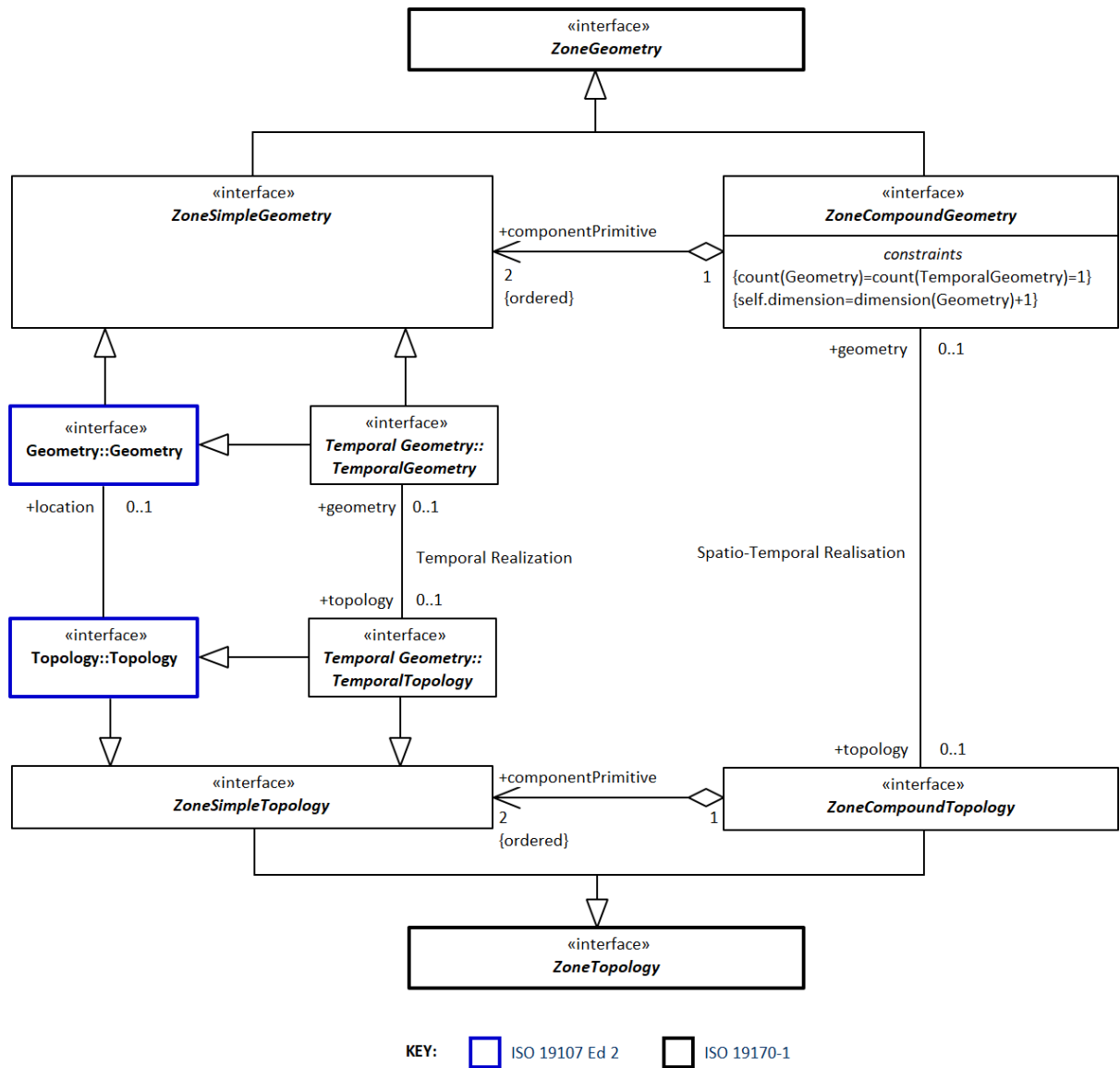
## 7.2.2. Zonal geometry and topology module

### 7.2.2.1. Context and data model

Referring to Figure 5, ZoneGeometry is either a primitive of ZoneSingleGeometry or a compound of two ZoneSingleGeometry primitives – one spatial and one temporal.

Zones exhibit topology of the same spatio-temporal dimension as their geometry.

Figure 5 – Components of Zonal Geometry and Topology module



### 7.2.2.2. Defining tables

- a) Table 17 – Elements of Zonal Geometry and Topology::ZoneCompoundGeometry
- b) Table 18 – Elements of Zonal Geometry and Topology::ZoneCompoundTopology
- c) Table 19 – Elements of Zonal Geometry and Topology::ZoneGeometry
- d) Table 20 – Elements of Zonal Geometry and Topology::ZoneSimpleGeometry

- e) Table 21 – Elements of Zonal Geometry and Topology::ZoneSimpleTopology
- f) Table 22 – Elements of Zonal Geometry and Topology::ZoneTopology

**Table 17 – Elements of Zonal Geometry and Topology::ZoneCompoundGeometry**

**Name:** ZoneCompoundGeometry

**Definition:** ZoneCompoundGeometry is a Compound of two ZoneSimpleGeometry elements, comprising one one-, two- or three-dimensional spatial geometry and one one-dimensional temporal geometry.  
**NOTE** This is analogous to an ISO 19111 Compound set of orthogonal space time axes comprising a set of orthogonal spatial axes and one temporal axis orthogonal to the spatial axes. ZoneCompoundGeometry has ZoneCompoundTopology.

**Stereotype:** Interface

**Inheritance from:** ZoneGeometry

**Abstract:** true

Associations:	Association with:	Obligation	Maximum occurrence	Provides:
	ZoneSimpleGeometry (feature type)	M	2	componentPrimitive
	ZoneCompoundTopology (feature type)	C	1	topology

**Public attributes:** (none)

**Constraints:**  
 count(Geometry)=count(TemporalGeometry)=1  
 self.dimension=dimension(Geometry)+1

**Table 18 – Elements of Zonal Geometry and Topology::ZoneCompoundTopology**

**Name:** ZoneCompoundTopology

**Definition:** ZoneCompoundTopology exhibits both spatial topology with respect to the spatial component of its geometry and temporal topology with respect to the temporal component of its geometry.

**Stereotype:** Interface



<b>Inheritance from:</b>	ZoneTopology			
<b>Abstract:</b>	true			
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	ZoneSimpleTopology (feature type)	M	2	componentPrimitive
	ZoneCompoundGeometry (feature type)	C	1	geometry
<b>Public attributes:</b>	(none)			
<b>Constraints:</b>	(none)			

**Table 19 – Elements of Zonal Geometry and Topology::ZoneGeometry**

<b>Name:</b>	ZoneGeometry
<b>Definition:</b>	ZoneGeometry is a ZoneSimpleGeometry or a ZoneCompoundGeometry. It is the root geometry for all spatio-temporal geometry.
<b>Stereotype:</b>	Interface
<b>Generalization of:</b>	ZoneCompoundGeometry, ZoneSimpleGeometry
<b>Abstract:</b>	true
<b>Associations:</b>	(none)

	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
<b>Public attributes:</b>	spatialDimension	Topological dimension of the spatial geometry component. <b>NOTE</b> For geometries on the surface of ellipsoids, the spatial dimension is 3, and the		M	1	Integer

	topological dimension is 2.			
temporalDimension	Topological dimension of the temporal geometry component.	M	1	Integer
topologicalDimension	Sum of dimensions of topological primitives.	M	1	Integer

**Constraints:** (none)

**Table 20 – Elements of Zonal Geometry and Topology::ZoneSimpleGeometry**

**Name:** ZoneSimpleGeometry

**Definition:** ZoneSimpleGeometry is a one-, two- or three-dimensional spatial geometry that is invariant over all time, OR a one-dimensional temporal geometry invariant over all space. A ZoneSimpleGeometry has topology appropriate for its geometry.

**Stereotype:** Interface

**Inheritance from:** ZoneGeometry

**Abstract:** true

<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	Topology (feature type)	C	1	topology

**Public attributes:** (none)

**Constraints:** (none)

**Table 21 – Elements of Zonal Geometry and Topology::ZoneSimpleTopology**

**Name:** ZoneSimpleTopology

<b>Definition:</b>	ZoneSimpleTopology is a one-, two- or three-dimensional spatial topology that is invariant over all time, OR a one-dimensional temporal topology that is invariant over all space.
<b>Stereotype:</b>	Interface
<b>Inheritance from:</b>	ZoneTopology
<b>Abstract:</b>	true
<b>Associations:</b>	(none)
<b>Public attributes:</b>	(none)
<b>Constraints:</b>	(none)

**Table 22 – Elements of Zonal Geometry and Topology::ZoneTopology**

<b>Name:</b>	ZoneTopology
<b>Definition:</b>	ZoneTopology is a ZoneSimpleTopology or a ZoneCompoundTopology
<b>Stereotype:</b>	Interface
<b>Generalization of:</b>	ZoneCompoundTopology, ZoneSimpleTopology
<b>Abstract:</b>	true
<b>Associations:</b>	(none)
<b>Public attributes:</b>	(none)
<b>Constraints:</b>	(none)

The following requirement applies:

### REQUIREMENT 2

[www.opengis.net/spec/DGGS/2.0/req/cc/zone/geometry](http://www.opengis.net/spec/DGGS/2.0/req/cc/zone/geometry)

## REQUIREMENT 2

The common classes for zonal geometry and topology shall conform to the data model in Figure 5 and defining tables in Table 17–Table 22. .

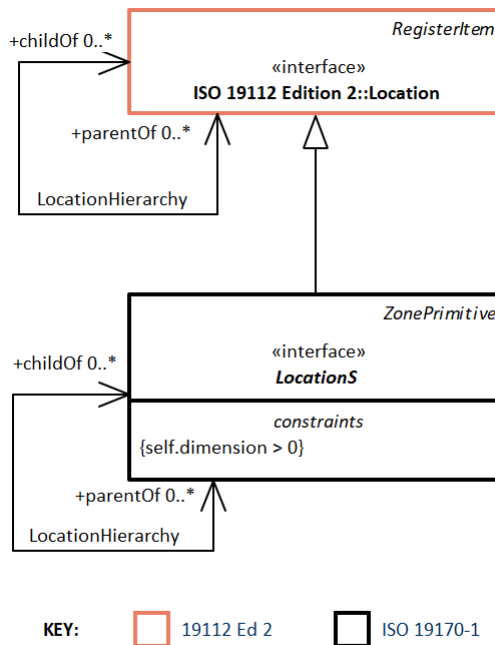
# 7.3. TEMPORAL AND ZONAL RS USING IDENTIFIERS PACKAGE

## 7.3.1. Spatial Location module

### 7.3.1.1. Context and data model

ISO 19112:2019 describes spatial referencing by geographic identifiers, locations and location classes. The LocationS interface is a specialisation of the Location interface from ISO 19112:2019. See Figure 6. In the next section the Period interface is introduced as the temporal equivalent of LocationS.

Figure 6 – Context for LocationS



### 7.3.1.2. Defining tables

a) Table 23 – Elements of Spatial Location::LocationS

**Table 23 – Elements of Spatial Location::LocationS**

**Name:** LocationS

**Definition:** Particular place or position.  
**NOTE** unlike a Location as specified in (<<ISO19112>>), all LocationS are owned and defined by their ReferenceSystem and not by an independent authority.

**Stereotype:** Interface

**Inheritance from:** Location, ZonePrimitive

**Abstract:** true

	Association with:	Obligation	Maximum occurrence	Provides:
<b>Associations:</b>	LocationS (feature type)	C	*	childOf
	LocationS (feature type)	C	*	parentOf

	Name	Definition	Derived	Obligation	Maximum occurrence	Data type
<b>Public attributes:</b>	coordinateTuple	Point within the extent of the spatial location.		M	3	DirectPosition (data type)
	extent	Spatial extent of the true location.	true	M	1	EX_Extent (data type)
	identifier	Identifier of the spatial location.		M	1	GeographicIdentifier (data type)

**Constraints:** self.dimension > 0

The following requirement applies:

## REQUIREMENT 3

[www.opengis.net/spec/DGGS/2.0/req/cc/spatial/location](http://www.opengis.net/spec/DGGS/2.0/req/cc/spatial/location)

---

*The common classes for spatial location shall conform to the data model in Figure 6 and defining table in Table 23.*

---

## 7.3.2. Temporal RS using Identifiers module

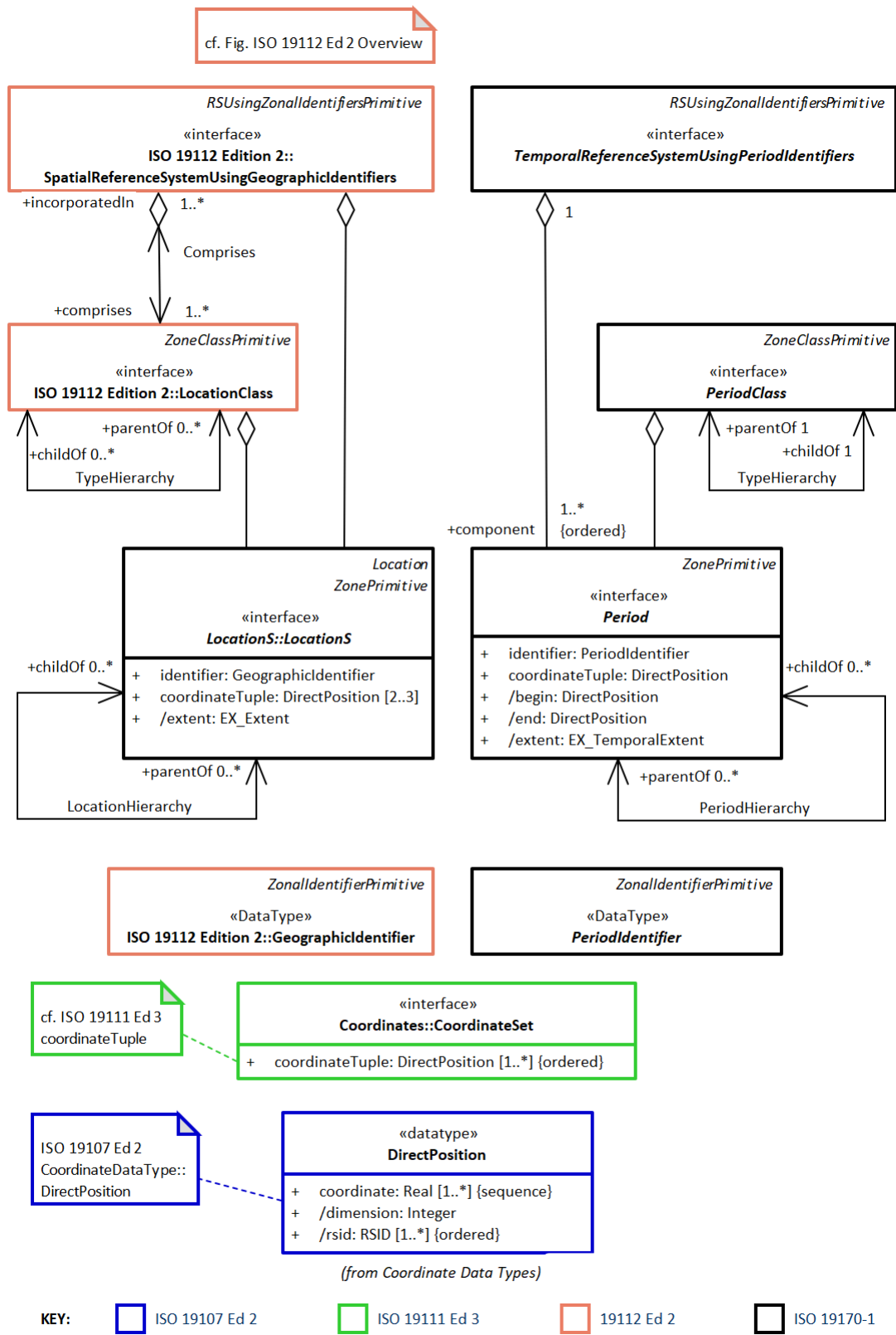
### 7.3.2.1. Context and data model

Semantically, the terms period and zone are defined as the temporal and spatio-temporal equivalents of a location. These are represented in the data model by the interfaces Period and Zone.

Referring to Figure 7 showing the spatial classes on the left and the temporal classes on the right, it is noted that Period is augmented by:

- PeriodIdentifier data-type as the temporal equivalent of GeographicIdentifier,
- Period interface as the temporal equivalent of LocationS, see Figure 6,
- PeriodClass interface as the temporal equivalent of LocationClass, and
- TemporalReferenceSystemusingPeriodIdentifiers interface as the temporal equivalent of SpatialReferenceusingGeographicIdentifiers.

Figure 7 – Context for Temporal RS using Identifiers module



### 7.3.2.2. Defining tables

- a) Table 24 – Elements of Temporal RS using Identifiers::Period
- b) Table 25 – Elements of Temporal RS using Identifiers::PeriodClass
- c) Table 26 – Elements of Temporal RS using Identifiers::PeriodIdentifier
- d) Table 27 – Elements of Temporal RS using Identifiers::TemporalReferenceSystemUsingPeriodIdentifiers

**Table 24 – Elements of Temporal RS using Identifiers::Period**

<b>Name:</b>	Period					
<b>Definition:</b>	Particular time span or era between two instants.					
<b>Stereotype:</b>	Interface					
<b>Inheritance from:</b>	ZonePrimitive					
<b>Abstract:</b>	true					
<b>Associations:</b>	<i>Association with:</i>			<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	Period (feature type)			C	*	childOf
	Period (feature type)			C	*	parentOf
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	begin	Instant at the beginning of the period.	true	M	1	DirectPosition (data type)
	coordinateTuple	Position within the extent of the period.		M	1	DirectPosition (data type)
	end	Instant at the end of the period.	true	M	1	DirectPosition (data type)



extent	Temporal extent of the period.	true	M	1	EX_TemporalExtent
identifier	Identifier of the period.		M	1	PeriodIdentifier (data type)

**Constraints:** (none)

**Table 25 – Elements of Temporal RS using Identifiers::PeriodClass**

**Name:** PeriodClass

**Definition:** Categorization of Periods.

**Stereotype:** Interface

**Inheritance from:** ZoneClassPrimitive

**Abstract:** true

	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
<b>Associations:</b>	PeriodClass (feature type)	M	1	childOf
	TM_OrdinalReferenceSystem (feature type)	M	*	incorporatedIn
	PeriodClass (feature type)	M	1	parentOf
<b>Public attributes:</b>	(none)			

**Constraints:** (none)

**Table 26 – Elements of Temporal RS using Identifiers::PeriodIdentifier**

**Name:** PeriodIdentifier

**Definition:** Temporal reference in the form of a label or code that identifies a period.

**Stereotype:** DataType

Inheritance from:	ZonalIdentifierPrimitive
Abstract:	true
Associations:	(none)
Public attributes:	(none)
Constraints:	(none)

**Table 27 – Elements of Temporal RS using Identifiers::TemporalReferenceSystemUsingPeriodIdentifiers**

Name:	TemporalReferenceSystemUsingPeriodIdentifiers			
Definition:	A temporal RS based on period identifiers.			
Stereotype:	Interface			
Inheritance from:	RSUsingZonalIdentifiersPrimitive			
Abstract:	true			
Associations:	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	Period (feature type)	M	*	component
Public attributes:	(none)			
Constraints:	(none)			

The following requirement applies:

**REQUIREMENT 4**

[www.opengis.net/spec/DGGS/2.0/req/cc/temporal/rsupi](http://www.opengis.net/spec/DGGS/2.0/req/cc/temporal/rsupi)

\_The common classes for reference systems using period identifiers shall conform to the data model in Figure 7 and defining tables in Table 24–Table 27.

### 7.3.3. Zonal RS using Identifiers module

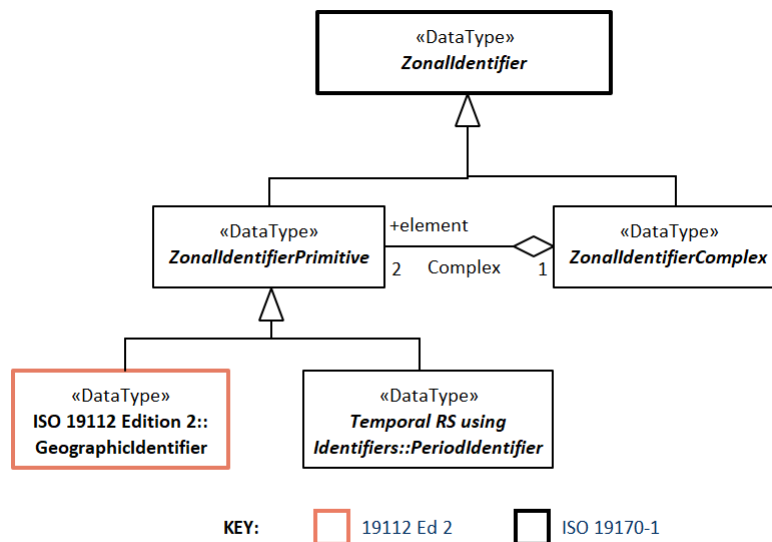
#### 7.3.3.1. Context and data model

Semantically zones are the spatio-temporal equivalent of periods and locations. Zones are represented in the data model by the interface Zone and along with it the following interfaces are also established.

- ZonalIdentifier,
- ZoneClass, and
- RSUsingZonalIdentifiers.

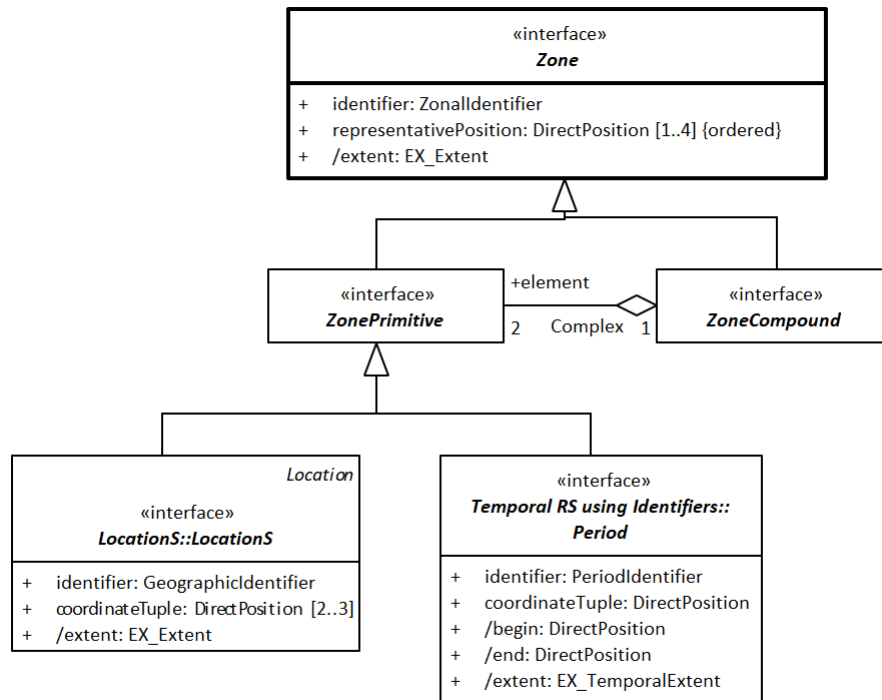
Referring to Figure 8, a ZonalIdentifier is either a zonal identifier primitive or a compound of two zonal identifier primitives, one spatial and one temporal.

Figure 8 – Primitives of ZonalIdentifier



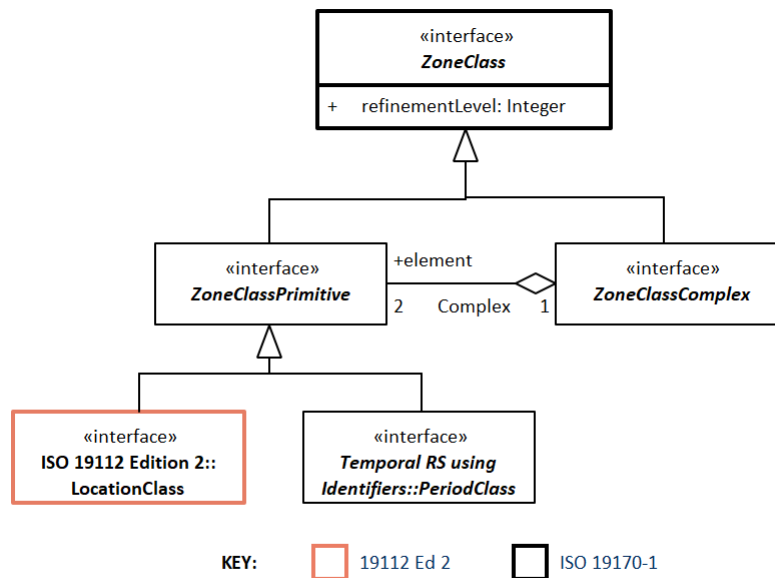
Referring to Figure 9, a Zone is either a zonal primitive or a compound of two zonal primitives, one spatial and one temporal.

Figure 9 – Primitives of Zone



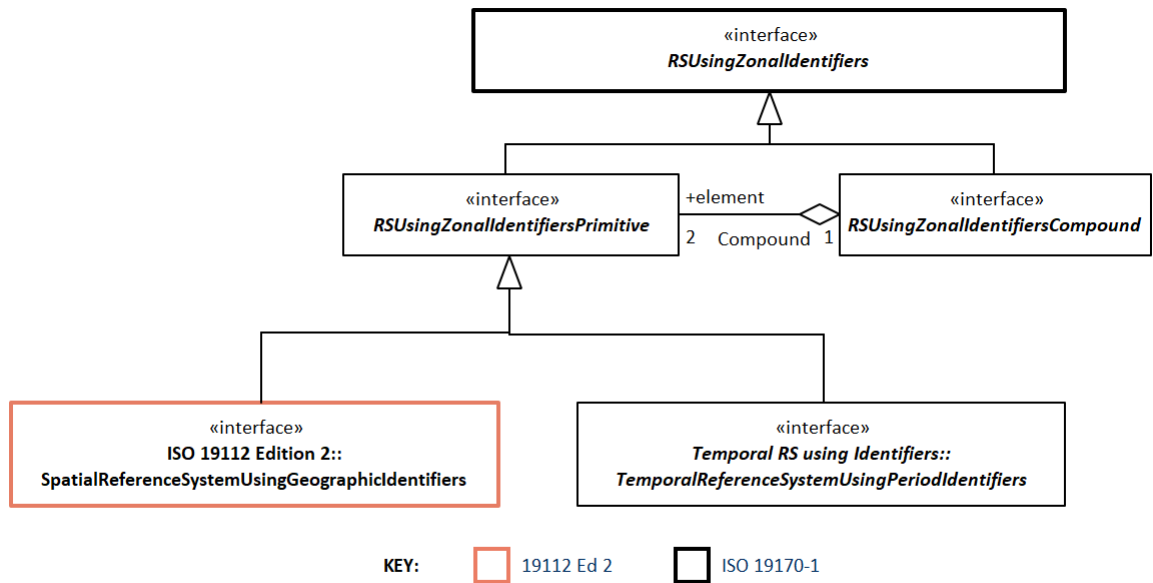
Referring to Figure 10, a ZoneClass is either a zonal class primitive or a compound of two zonal class primitives, one spatial and one temporal.

Figure 10 – Primitives of ZoneClass



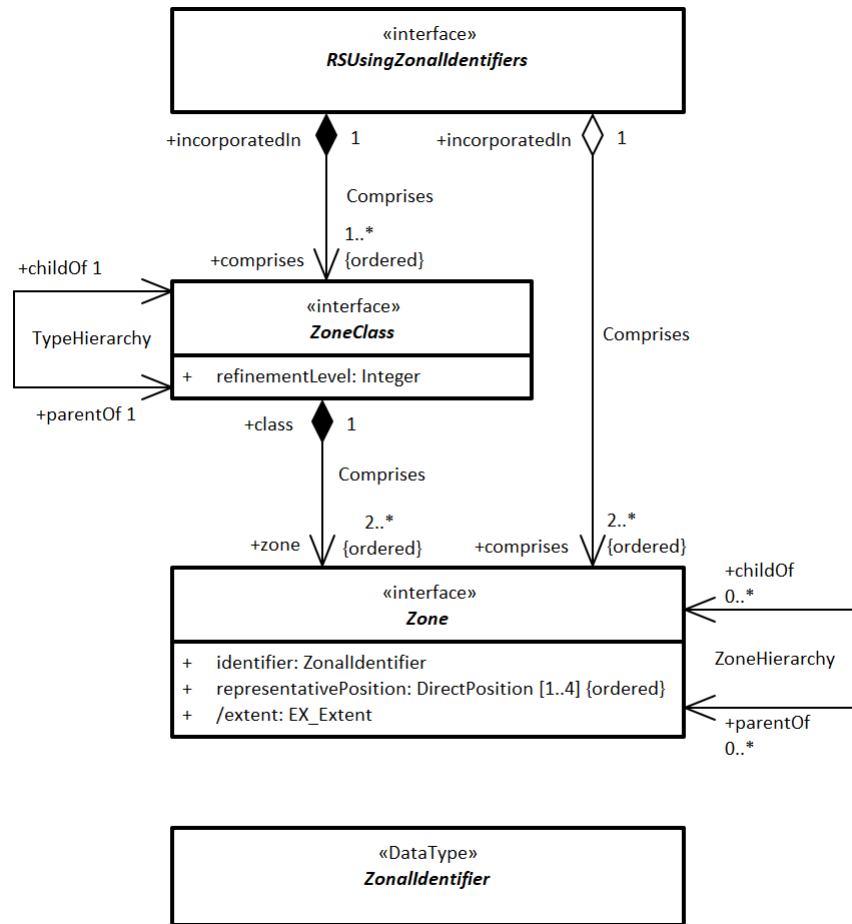
Referring to Figure 11, an *RSUsingZonalIdentifiers* is either a zonal RS using identifiers primitive or a compound of two of its primitives, one spatial and one temporal.

Figure 11 – Primitives of *RSUsingZonalIdentifiers*



Referring to Figure 12, *Zone*, *ZonalIdentifier* and *ZoneClass* come together to form *RSUsingZonalIdentifiers*.

Figure 12 – Components of RSUsingZonalIdentifiers



### 7.3.3.2. Defining tables

- Table 28 – Elements of Zonal RS using Identifiers::RSUsingZonalIdentifiers
- Table 29 – Elements of Zonal RS using Identifiers::RSUsingZonalIdentifiersCompound
- Table 30 – Elements of Zonal RS using Identifiers::RSUsingZonalIdentifiersPrimitive
- Table 31 – Elements of Zonal RS using Identifiers::ZonalIdentifier
- Table 32 – Elements of Zonal RS using Identifiers::ZonalIdentifierComplex
- Table 33 – Elements of Zonal RS using Identifiers::ZonalIdentifierPrimitive
- Table 34 – Elements of Zonal RS using Identifiers::Zone
- Table 35 – Elements of Zonal RS using Identifiers::ZoneClass

- i) Table 36 – Elements of Zonal RS using Identifiers::ZoneClassComplex
- j) Table 37 – Elements of Zonal RS using Identifiers::ZoneClassPrimitive
- k) Table 38 – Elements of Zonal RS using Identifiers::ZoneCompound
- l) Table 39 – Elements of Zonal RS using Identifiers::ZonePrimitive

**Table 28 – Elements of Zonal RS using Identifiers::RSUsingZonalIdentifiers**

<b>Name:</b>	RSUsingZonalIdentifiers			
<b>Definition:</b>	A reference system using zonal identifiers is either a reference system using zonal identifiers primitive or a compound of one spatial reference system using zonal identifiers and one temporal reference system using period identifiers primitives.			
<b>Stereotype:</b>	Interface			
<b>Generalization of:</b>	RSUsingZonalIdentifiersCompound, RSUsingZonalIdentifiersPrimitive			
<b>Abstract:</b>	true			
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
		M	*	comprises
	Zone (feature type)	M	*	comprises
<b>Public attributes:</b>	(none)			
<b>Constraints:</b>	(none)			

**Table 29 – Elements of Zonal RS using Identifiers::RSUsingZonalIdentifiersCompound**

<b>Name:</b>	RSUsingZonalIdentifiersCompound			
<b>Definition:</b>	A reference system using zonal identifiers compound is a compound of one spatial reference system using zonal identifiers and one temporal reference system using zonal identifiers primitives.			
<b>Stereotype:</b>	Interface			
<b>Inheritance from:</b>	RSUsingZonalIdentifiers			

**Abstract:** true

<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	RUsingZonalIdentifiersPrimitive (feature type)	M	2	element

**Public attributes:** (none)

**Constraints:** (none)

**Table 30 – Elements of Zonal RS using Identifiers::RUsingZonalIdentifiersPrimitive**

**Name:** RUsingZonalIdentifiersPrimitive

**Definition:** A reference system using zonal identifiers primitive is either a spatial reference system using geographic identifiers or a temporal reference system using period Identifiers.

**Stereotype:** Interface

**Inheritance from:** RUsingZonalIdentifiers

**Abstract:** true

**Associations:** (none)

**Public attributes:** (none)

**Constraints:** (none)

**Table 31 – Elements of Zonal RS using Identifiers::ZonalIdentifier**

**Name:** ZonalIdentifier

**Definition:** Spatial, temporal or spatio-temporal reference in the form of a label or code that identifies a zone.

**Stereotype:** DataType

**Generalization of:** ZonalIdentifierComplex, ZonalIdentifierPrimitive



**Abstract:** true

**Associations:** (none)

**Public attributes:** (none)

**Constraints:** (none)

**Table 32 – Elements of Zonal RS using Identifiers::ZonalIdentifierComplex**

**Name:** ZonalIdentifierComplex

**Definition:** Zonal identifier complex is a complex of two zonal identifier primitives, one geographic identifier and one period identifier.

**Stereotype:** DataType

**Inheritance from:** ZonalIdentifier

**Abstract:** true

<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	ZonalIdentifierPrimitive (feature type)	M	2	element

**Public attributes:** (none)

**Constraints:** (none)

**Table 33 – Elements of Zonal RS using Identifiers::ZonalIdentifierPrimitive**

**Name:** ZonalIdentifierPrimitive

**Definition:** Zonal identifier primitive is either a geographic identifier or a period identifier.

**Stereotype:** DataType

**Inheritance from:** ZonalIdentifier

**Abstract:** true

**Associations:** (none)

**Public attributes:** (none)

**Constraints:** (none)

**Table 34 – Elements of Zonal RS using Identifiers::Zone**

**Name:** Zone

**Definition:** A zone is a particular spatial, temporal or spatio-temporal place.

**Stereotype:** Interface

**Generalization of:** ZoneCompound, ZonePrimitive

**Abstract:** true

	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
<b>Associations:</b>	Zone (feature type)	C	*	childOf
	Zone (feature type)	C	*	parentOf

	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
<b>Public attributes:</b>	extent	extent of the zone	true	M	1	EX_Extent (data type)
	identifier	Name or label for the Zone.		M	1	ZonalIdentifier (data type)
	representativePosition	Interior position for the Zone.		M	4	DirectPosition (data type)

**Constraints:** (none)

**Table 35 – Elements of Zonal RS using Identifiers::ZoneClass**

<b>Name:</b>	ZoneClass					
<b>Definition:</b>	Categorization of zones.					
<b>Stereotype:</b>	Interface					
<b>Generalization of:</b>	ZoneClassComplex, ZoneClassPrimitive					
<b>Abstract:</b>	true					
<b>Associations:</b>	<i>Association with:</i>		<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>	
	ZoneClass (feature type)		M	1	childOf	
	ZoneClass (feature type)		M	1	parentOf	
<b>Public attributes:</b>			M	*	zone	
	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	refinementLevel	Refinement level used to define the zone class.		M	1	Integer
<b>Constraints:</b>	(none)					

**Table 36 – Elements of Zonal RS using Identifiers::ZoneClassComplex**

<b>Name:</b>	ZoneClassComplex					
<b>Definition:</b>	Zone class complex is a complex of two zone class primitives, one location class and one period class.					
<b>Stereotype:</b>	Interface					
<b>Inheritance from:</b>	ZoneClass					
<b>Abstract:</b>	true					
<b>Associations:</b>	<i>Association with:</i>		<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>	

	ZoneClassPrimitive (feature type)	M	2	element
<b>Public attributes:</b>	(none)			
<b>Constraints:</b>	(none)			

**Table 37 – Elements of Zonal RS using Identifiers::ZoneClassPrimitive**

<b>Name:</b>	ZoneClassPrimitive
<b>Definition:</b>	Zone class primitive is either a location class or a period class.
<b>Stereotype:</b>	Interface
<b>Inheritance from:</b>	ZoneClass
<b>Abstract:</b>	true
<b>Associations:</b>	(none)
<b>Public attributes:</b>	(none)
<b>Constraints:</b>	(none)

**Table 38 – Elements of Zonal RS using Identifiers::ZoneCompound**

<b>Name:</b>	ZoneCompound			
<b>Definition:</b>	A zone compound is a compound of two zone primitives, one spatial location and one temporal period.			
<b>Stereotype:</b>	Interface			
<b>Inheritance from:</b>	Zone			
<b>Abstract:</b>	true			
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	ZonePrimitive (feature type)	M	2	element

Public attributes:	(none)
Constraints:	(none)

**Table 39 – Elements of Zonal RS using Identifiers::ZonePrimitive**

Name:	ZonePrimitive
Definition:	A zonal primitive is either a spatial location or a temporal period.
Stereotype:	Interface
Inheritance from:	Zone
Abstract:	true
Associations:	(none)
Public attributes:	(none)
Constraints:	(none)

The following requirement applies:

**REQUIREMENT 5**

[www.opengis.net/spec/DGGS/2.0/req/cc/zone/rsuzi](http://www.opengis.net/spec/DGGS/2.0/req/cc/zone/rsuzi)

*The common classes for reference systems using zonal identifiers shall conform to the data model in Figure 8–Figure 12 and defining tables in Table 28–Table 39.*

8

# DGGS CORE REFERENCE SYSTEM AND FUNCTIONS PACKAGE

---

# DGGS CORE REFERENCE SYSTEM AND FUNCTIONS PACKAGE

---

## 8.1. DGGS CORE PACKAGE CONFORMANCE CLASSES

---

This clause specifies the DGGS Core RS conformance class and the DGGS Core Functions conformance class. These cover:

- a) DGGS Core RS – comprising parent global geometry, base CRS, and RS using zonal identifiers with structured geometry;
- b) DGGS Core Functions – comprising quantization, topological query, and interoperation.

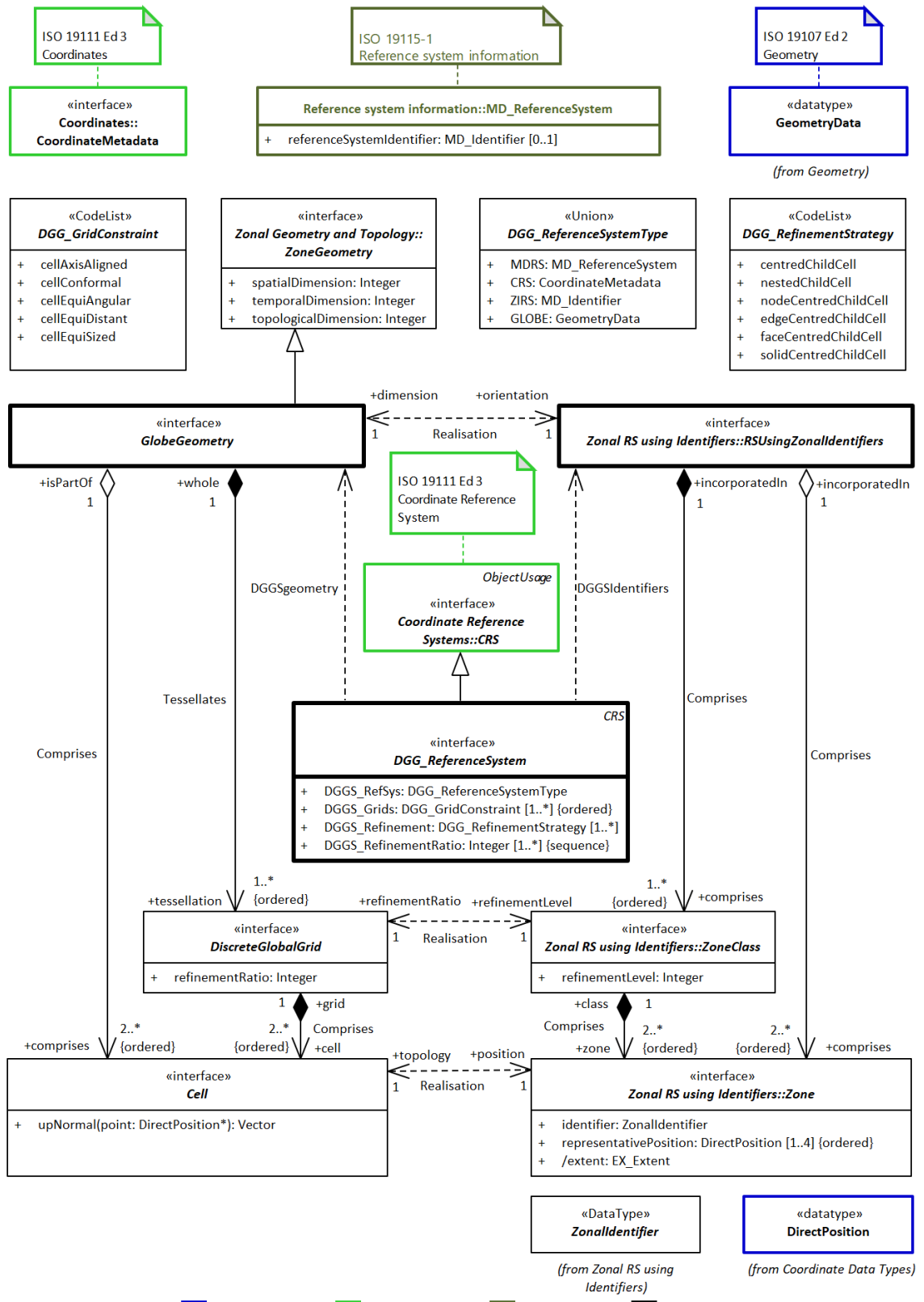
## 8.2. CORE RS USING ZONAL IDENTIFIERS WITH STRUCTURED GEOMETRY MODULE

---

### 8.2.1. Core RS data model and base CRS

The DGGS Core RS shown in Figure 13 is a RS using zonal identifiers with structured geometry for a globe. It is defined by the attributes of the DGG\_ReferenceSystem interface Table 41 and its metadata. In the context of the DGGS, and particularly in the DGGS Core, the term globe is used in its most general sense to represent a mathematical model of any planetary body and, depending on need, potentially its surroundings out to the orbit of a planet's outer moons, and for spatio-temporal DGGS over a defined time span. The spatio-temporal extent of the entire globe is referred to as the domain of the DGGS. The DGGS Core itself is dimension agnostic.

Figure 13 – Core RS using Zonal Identifiers with Structured Geometry module



KEY:   ISO 19107 Ed 2   ISO 19111 Ed 3   19115-1 Ed 1   ISO 19170-1



The DGGs Core RS model in Figure 13 describes a hierarchy of paired elements — at each tier of the hierarchy a geometry element, shown on the left of the figure is paired with the equivalent zonal element, shown on the right of the figure.

At the root of the hierarchy, the globe's reference model and base CRS for the DGGs are defined. A single parent geometry (GlobeGeometry) is defined to coincide with the globe's reference model. The parent geometry's dimensionality governs the dimensionality of the DGGs (See Clause 8.2.3). The parent global geometry is paired with a RS using zonal identifiers (RSUsingZonalIdentifiers).

A sequence of discrete global grids (DiscreteGlobalGrids), each paired with a ZoneClass, define the lower levels of the hierarchy.

Each DiscreteGlobalGrid is made up of Cells. Each Cell occupies a region of space-time (a Zone). The Zones corresponding to Cells in a DiscreteGlobalGrid all belong to the corresponding ZoneClass (See Clause 8.2.5).

Cells provide zones with geometry and topology, and zones provide cells with names in the form of zonal identifiers and representative positions in the base CRS in the form of direct positions (See Clause 8.2.4).

Each cell is the child of one or more parent cells in the parent discrete global grid in the level above in the hierarchy. The cells of a discrete global grid are topologically related to the cells in the parent(s) by the collection of refinement strategies defined in DGG\_ReferenceSystem(DGGs\_Refinement).

Each cell's geometry, orientation and size is governed by the constraint defined in DGG\_ReferenceSystem(DGGs\_Grids) and by the sequence of refinement ratios defined in DGG\_ReferenceSystem(DGGs\_RefinementRatio). If a sequence of refinementRatio is defined, the values are applied to each level in the hierarchy in a recurring sequence starting at the top with the first refinementRatio in the sequence and working down through the levels in order.

## 8.2.2. Defining tables

- a) Table 40 – Elements of Core RS using Zonal Identifiers with Structured Geometry::Cell
- b) Table 41 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG\_ReferenceSystem
- c) Table 42 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG\_ReferenceSystemType
- d) Table 43 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DiscreteGlobalGrid
- e) Table 44 – Elements of Core RS using Zonal Identifiers with Structured Geometry::GlobeGeometry
- f) Table 45 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG\_GridConstraint

g) Table 46 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG\_RefinementStrategy

Table 40 – Elements of Core RS using Zonal Identifiers with Structured Geometry::Cell

<b>Name:</b>	Cell
<b>Definition:</b>	Reference system unit of geometry associated with a Zone. As part of GlobeGeometry, it has the same spatial, temporal and topological dimensionality as GlobeGeometry.
<b>Stereotype:</b>	Interface
<b>Abstract:</b>	true
<b>Associations:</b>	(none)
<b>Public attributes:</b>	(none)
<b>Constraints:</b>	(none)

Table 41 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG\_ReferenceSystem

<b>Name:</b>	DGG_ReferenceSystem					
<b>Definition:</b>	Defining characteristics of a Reference system using zonal identifiers with structured geometry.					
<b>Stereotype:</b>	Interface					
<b>Inheritance from:</b>	CRS, CRS					
<b>Abstract:</b>	true					
<b>Associations:</b>	(none)					
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	DGGS_Grids	List of characteristics that constraint the grid cells in this DGGS		M	*	DGG_GridConstraint (code list)

		in decreasing order of priority.			
DGGS_Refinement		List of topological relationships between parent and child cells in this DGGS.	M	*	DGG_Refinement Strategy (code list)
DGGS_Refinement Ratio		List of refinement ratios of parent cell size to child cell size, in the order that they are used in constructing child cells in the DGGS. If the list is shorter than the number of discrete global grids in the DGGS, then it is used as a recurring sequence.	M	*	Integer
DGGS_RefSys		Reference system metadata.	M	1	DGG_ReferenceSystem Type (union data type)
<b>Constraints:</b>	(none)				

**Table 42 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG\_ReferenceSystemType**

**Name:** DGG\_ReferenceSystemType

**Definition:** Defining metadata elements of the base CRS for this DGGS.

**Stereotype:** Union

**Abstract:** true

**Associations:** (none)

	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
<b>Public attributes:</b>	CRS	Metadata required to reference coordinates.		M	1	CoordinateMetadata

GLOBE	GeometryData for the chosen GlobeGeometry that specifies geometry, spatial, temporal and topological dimensionality and domain of the globe for this DGGs.	M	1	GeometryData (data type)
MDRS	RS information describing this whole DGGs.	M	1	MD_ReferenceSystem
ZIRS	Identifier for the RSUsing ZonalIdentifiers used by this DGGs.	M	1	MD_Identifier (data type)

Constraints: (none)

**Table 43 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DiscreteGlobalGrid**

Name: DiscreteGlobalGrid

Definition: Set of Cells at the same refinement level.

Stereotype: Interface

Abstract: true

Associations:	Association with:	Obligation	Maximum occurrence	Provides:
		M	*	cell

Public attributes:	Name	Definition	Derived	Obligation	Maximum occurrence	Data type
refinementRatio		Ratio of the number of cells in the parent DiscreteGlobalGrid to the number in this Discrete GlobalGrid.		M	1	Integer

Constraints: (none)

**Table 44 – Elements of Core RS using Zonal Identifiers with Structured Geometry::GlobeGeometry**

<b>Name:</b>	GlobeGeometry												
<b>Definition:</b>	Parent geometry specifying the geometry, dimensionality and domain of the globe for this DGGs.												
<b>Stereotype:</b>	Interface												
<b>Inheritance from:</b>	ZoneGeometry												
<b>Abstract:</b>	true												
<b>Associations:</b>	<table border="1"> <thead> <tr> <th><i>Association with:</i></th> <th><i>Obligation</i></th> <th><i>Maximum occurrence</i></th> <th><i>Provides:</i></th> </tr> </thead> <tbody> <tr> <td>Cell (feature type)</td> <td>M</td> <td>*</td> <td>comprises</td> </tr> <tr> <td></td> <td>M</td> <td>*</td> <td>tessellation</td> </tr> </tbody> </table>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>	Cell (feature type)	M	*	comprises		M	*	tessellation
	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>									
Cell (feature type)	M	*	comprises										
	M	*	tessellation										
<b>Public attributes:</b>	(none)												
<b>Constraints:</b>	(none)												

**Table 45 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG\_GridConstraint**

<b>Name:</b>	DGG_GridConstraint				
<b>Definition:</b>	CodeList for constraints that are used to define different categories of DGGs. Each constraint is a constraint on the shape, size, or orientation of cells in a Discrete GlobalGrid.				
<b>Stereotype:</b>	CodeList				
<b>Abstract:</b>	true				
<b>Associations:</b>	(none)				
<b>Values:</b>	<table border="1"> <thead> <tr> <th><i>Name</i></th> <th><i>Definition</i></th> </tr> </thead> <tbody> <tr> <td>cellAxisAligned</td> <td>Cell edges are parallel to the base CRS's coordinate system axes.</td> </tr> </tbody> </table>	<i>Name</i>	<i>Definition</i>	cellAxisAligned	Cell edges are parallel to the base CRS's coordinate system axes.
	<i>Name</i>	<i>Definition</i>			
cellAxisAligned	Cell edges are parallel to the base CRS's coordinate system axes.				

cellConformal	Variation in shape between all the cells in each DiscreteGlobalGrid is minimized.
cellEquiAngular	Variation in bearing from one cell's representative position to the next neighboring cell's representative positions in each DiscreteGlobalGrid is minimized.
cellEquiDistant	Variation in distance from a cell's representative position to all of its neighboring cell's representative positions in each DiscreteGlobalGrid is minimized.
cellEquiSized	Variation in interior size between all cells in each DiscreteGlobalGrid is minimized.

**Table 46 – Elements of Core RS using Zonal Identifiers with Structured Geometry::DGG\_RefinementStrategy**

<b>Name:</b>	DGG_RefinementStrategy								
<b>Definition:</b>	CodeList for strategies that are used to define different categories of DGGs. Each strategy defines the topological relationship of one or more elements of cell geometry belonging to a child cell with one or more elements of geometry of its parent cell.								
<b>Stereotype:</b>	CodeList								
<b>Abstract:</b>	true								
<b>Associations:</b>	(none)								
<b>Values:</b>	<table border="1"> <thead> <tr> <th>Name</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>centredChildCell</td> <td>parent#zone.representativePosition() = child#zone.representativePosition() for one child. Each parent cell shares a cell←zone.representativePosition with one of its child cells.</td> </tr> <tr> <td>nestedChildCell</td> <td>parent.boundary = parent.child().boundary. The boundary of the set of child cells for a parent is identical to the parent's boundary.</td> </tr> <tr> <td>nodeCentredChildCell</td> <td>Each parent cell has a child#zone.representativePosition coincident</td> </tr> </tbody> </table>	Name	Definition	centredChildCell	parent#zone.representativePosition() = child#zone.representativePosition() for one child. Each parent cell shares a cell←zone.representativePosition with one of its child cells.	nestedChildCell	parent.boundary = parent.child().boundary. The boundary of the set of child cells for a parent is identical to the parent's boundary.	nodeCentredChildCell	Each parent cell has a child#zone.representativePosition coincident
Name	Definition								
centredChildCell	parent#zone.representativePosition() = child#zone.representativePosition() for one child. Each parent cell shares a cell←zone.representativePosition with one of its child cells.								
nestedChildCell	parent.boundary = parent.child().boundary. The boundary of the set of child cells for a parent is identical to the parent's boundary.								
nodeCentredChildCell	Each parent cell has a child#zone.representativePosition coincident								

		with each of the parent's <i>nodes</i> (zero-Dimensional topological boundary element).
	edgeCentredChildCell	Each parent cell of dimension greater than 1 has a child cell for which the <code>cell#zone.representativePosition</code> lies on each of the parent's <i>edges</i> (one-Dimensional topological boundary element)
	faceCentredChildCell	Each parent cell of dimension greater than 2 has a child cell for which the <code>cell#zone.representativePosition</code> lies on each of the parent's <i>faces</i> (two-Dimensional topological boundary element)
	solidCentredChildCell	Each parent cell of dimension greater than 3 has a child cell for which the <code>cell#zone.representativePosition</code> lies on each of the parent's <i>solids</i> (three-Dimensional topological boundary element)

The following requirements apply:

#### REQUIREMENT 6

[www.opengis.net/spec/DGGS/2.0/req/core/rs/harmonized\\_model](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/harmonized_model)

*The data model for a DGGS RS and its element definitions shall comply with the DGGS Core RS data model in Figure 13 and definitions in Table 40–Table 46.*

#### REQUIREMENT 7

[www.opengis.net/spec/DGGS/2.0/req/core/rs/crs](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/crs)

*A DGGS RS shall define a CRS, and comply with requirements for provision of coordinate epoch as specified for MD\_ReferenceSystem.*

### 8.2.3. Global domain

The domain of the DGGS shall be defined as the entire globe, with cells that “*exhaustively cover the globe without overlapping or underlapping*” (Goodchild [7]). Applying these criteria to the

cells in each discrete global grid, there shall be no gaps between cells and no positions that are covered by more than one cell.

Each cell inherits its dimensionality from the choice of geometry for the GlobalGeometry class, so a reference system that specifies a geometry and domain as the globe's surface will have two-dimensional cells on the surface of the globe, and one that specifies a geometry and domain as a globe's volume will have three-dimensional cells filling the globe. A reference system that specifies a linear geometry and domain, for instance for time, will result in one-dimensional cells.

The following requirements apply:

#### REQUIREMENT 8

[www.opengis.net/spec/DGGS/2.0/req/core/rs/global\\_domain](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain)

---

*DGGS RS global domain – the reference system shall specify a global domain, and its spatial, temporal and topological dimensionality.*

---

#### REQUIREMENT 9

[www.opengis.net/spec/DGGS/2.0/req/core/rs/global\\_domain/complete](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain/complete)

---

*DGGS RS domain completeness – the level zero discrete global grid shall cover the entire global domain.*

---

#### REQUIREMENT 10

[www.opengis.net/spec/DGGS/2.0/req/core/rs/global\\_domain/unique](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain/unique)

---

*DGGS RS location uniqueness – every location in the domain of the reference system shall be in exactly one cell of the level zero discrete global grid.*

---

## 8.2.4. Cells and zones

### 8.2.4.1. Cell simple geometry

Semantically, the terms cell and zone refer to different characteristics of the same region of space-time. Cells in a DGGS shall be geometrically simple. Simple geometries have the following properties:

1. they do not self-intersect;



2. they are topologically the same as a circle, or the circle's equivalent in the dimension of the cell. e.g. to a sphere in three-dimensions; and
3. they enclose a region which is always measurable using a metric of the same dimensionality as the cell.

The following requirement applies:

#### REQUIREMENT 11

[www.opengis.net/spec/DGGS/2.0/req/core/rs/cell/simple](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/cell/simple)

---

*For each successive level of grid refinement, a DGGS specification shall define cells with simple geometry.*

---

#### 8.2.4.2. Cell position

Each cell's zone has a fixed representative position in the space of the base CRS, recorded as a direct position.

The following requirement applies:

#### REQUIREMENT 12

[www.opengis.net/spec/DGGS/2.0/req/core/rs/cell/direct\\_position](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/cell/direct_position)

---

*All zones in each discrete global grid shall be assigned a direct position that is within the zone's boundary.*

---

#### 8.2.4.3. Cell address

Each cell's zone shall be assigned a unique address in the form of a zonal identifier. The value assigned to each address shall be structured on one or more of these four general indexing methods: hierarchy-based, space-filling curve based, coordinate [1] and encoded address schemas (such as those used for IP addresses [10]).

The following requirement applies:

#### REQUIREMENT 13

[www.opengis.net/spec/DGGS/2.0/req/core/rs/cell/address](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/cell/address)

---

### REQUIREMENT 13

*All zones in all discrete global grids shall have a globally unique zonal identifier (or cell index) that provides a spatio-temporal reference.*

---

## 8.2.5. Discrete global grid and its sequence

Cells at the same level in the tessellation hierarchy are aggregated into discrete global grids. The hierarchy of discrete global grids is an ordered sequence, typically also of decreasing cell size, representing the lowest resolution to higher resolutions. The discrete sequence of grids forms a multi-resolution grid hierarchy that is the basis for the DGGS RS.

The following requirements apply:

### REQUIREMENT 14

[www.opengis.net/spec/DGGS/2.0/req/core/rs/discrete\\_global\\_grid](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/discrete_global_grid)

---

*A DGGS RS shall define discrete global grids as aggregations of all the cells at the same level in the hierarchy.*

---

### REQUIREMENT 15

[www.opengis.net/spec/DGGS/2.0/req/core/rs/discrete\\_global\\_grid/sequence](http://www.opengis.net/spec/DGGS/2.0/req/core/rs/discrete_global_grid/sequence)

---

*A DGGS RS shall sort its hierarchy of discrete global grids in order of increasing refinement level.*

---

## 8.3. DGGS CORE FUNCTIONS

---

### 8.3.1. Core Quantization Functions module

A DGGS is defined based on the geometry of the globe in a data agnostic manner. Therefore, a DGGS specification shall define quantization methods for assigning data to cells so that the data is accessible for future use. Different quantization strategies may be used for sampling content into cells. For example, a single DGGS may be used as a data structure for integrating multiple datasets of different types (e.g. vector and raster datasets)[9] and in different ways (e.g. DGGS cells as data tiles, or one raster pixel per DGGS cell or DGGS cell indices as vector coordinate-pairs). This abstract specification makes use of the concepts defined by ISO 19156:2011 to

facilitate the association of observations/spatial data to a DGGs cell(s). Some DGGs/polyhedron choices are more efficient for sampling than others (e.g. DGGs based on an icosahedron).

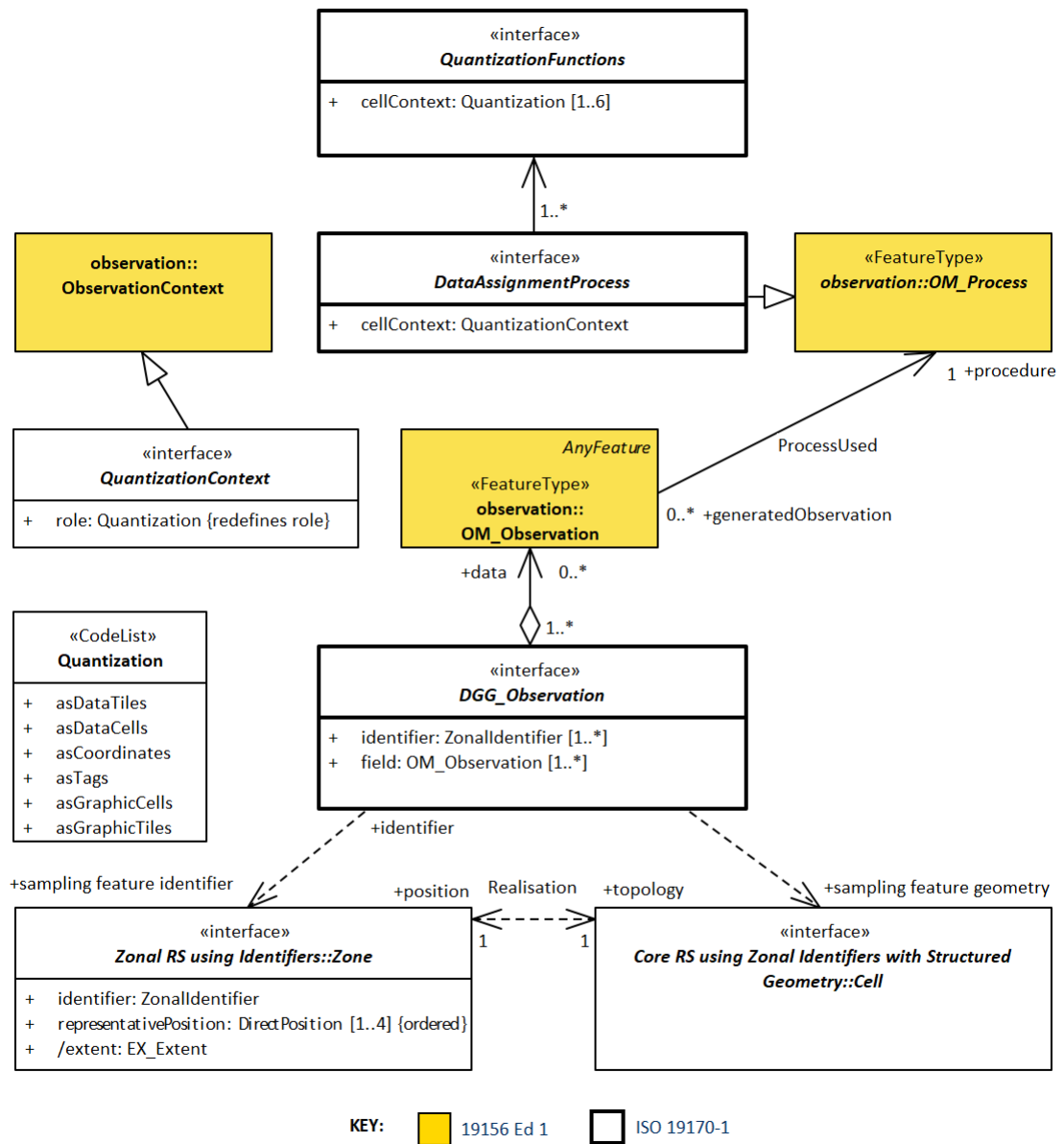
Multiple observation contexts are recognized for quantization, each corresponding to a distinct role for DGGs cells to play. In any particular DGGs specification, one or more (and potentially all) roles may be described for either internal or external use to support interoperability, as follows:

1. **asDataTiles:** In data tile quantization, spatial feature/observations (e.g. point clouds, images, vectors, etc.) are aggregated and clipped to cell boundaries and stored in tiles without any changes made to the feature type parameters. The cells of the DGGs provide a multi-, or single-resolution tiling schema with the cell index used as the identifier in the tile naming convention. In the context of “Big Data Analytics”, ‘asDataTile’ support will likely be the most efficient type of granularity for job submission on HPC/HPD or Cloud ICT infrastructure, particularly for dominantly embarrassingly parallel analyses. It is also likely to be the most efficient granularity for many data transfer requests.
2. **asDataCells:** In data cell quantization, the spatial features/observations (e.g. point clouds, images, vectors, etc.) are sampled to each DGGs cell by assignment of data value(s) using the cell’s geometry to govern the quantization operation.
3. **asCoordinates:** In coordinate quantization, each coordinate tuple from the spatial feature/observation is converted to a cell index of an appropriate level of precision. The cell data package will include appropriate vector topology to preserve the structure of the spatial feature in the context of the DGGs.
4. **asTags:** In tag quantization, cell index values are “tagged” to data objects in a similar fashion to social media records. The refinement level of the cell index is indicative of the precision with which the location of a spatial feature/ observation and/or its spatial extent are known. This can be thought of as a convex hull with the same geometry of the DGGs cell surrounding the objects to be assigned to that cell.
5. **asGraphicCells:** In graphic cell quantization, data is rendered to cells, and refinement levels are leveraged to support corresponding levels of detail or zoom levels.
6. **asGraphicTiles:** In graphic tile quantization, graphic cells are tiled, and often cached for delivery to a display system. As with data tiles, the cell index is used as the identifier in the tile naming convention.

The extent of the data assigned at any time to a particular DGGs implementation, defines the DGGs’s extent. The extent is likely to vary over the DGGs’s lifecycle as the extent of data assigned to it changes. The domain of the DGGs, is however, always fixed and always defined over the entire surface model of the DGGs’s globe.

Figure 14 shows the key elements required to perform data quantization operations in a DGGs specification.

Figure 14 – Components of the Core Quantization Functions module



### 8.3.2. Defining tables for Core Quantization Functions sub-package

- a) Table 47 – Elements of Core Quantization Functions::DataAssignmentProcess
- b) Table 48 – Elements of Core Quantization Functions::DGG\_Observation
- c) Table 49 – Elements of Core Quantization Functions::QuantizationContext
- d) Table 50 – Elements of Core Quantization Functions::QuantizationFunctions

e) Table 51 – Elements of Core Quantization Functions::Quantisation

**Table 47 – Elements of Core Quantization Functions::DataAssignmentProcess**

<b>Name:</b>	DataAssignmentProcess					
<b>Definition:</b>	The class DataAssignmentProcess is a generalization of OM_Process, which in turn is an instance of the «metaclass» GF_FeatureType (ISO 19109:2007), which therefore represents a feature type. DataAssignmentProcess is abstract, and has no attributes, operations or associations. It serves as the base class for DataAssignment processes. The purpose of a data assignment process is to generate an assignment result. An instance of DataAssignmentProcess is often a data import function to import data from a pre-existing spatial dataset, but as in OM_Process “it may also be an instrument or sensor, a human observer, a simulator, or a process or algorithm applied to more primitive results used as inputs”. [Source ISO 19156:2011, clause 7.2.3]					
<b>Stereotype:</b>	Feature					
<b>Inheritance from:</b>	OM_Process					
<b>Abstract:</b>	true					
<b>Associations:</b>	(none)					
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	cellContext	Roll that cell provides in this DataAssignment Process.		M	1	QuantizationContext
<b>Constraints:</b>	(none)					

**Table 48 – Elements of Core Quantization Functions::DGG\_Observation**

<b>Name:</b>	DGG_Observation					
<b>Definition:</b>	DGG_Observation is an abstract class holding ZonalIdentifier, OM_Observation tuples. In the context of Quantization, DGG_Observation holds records of Observations made with DataAssignmentProcess in assigning values to cells.					
<b>Stereotype:</b>	Interface					
<b>Abstract:</b>	true					

<b>Associations:</b>	<i>Association with:</i>			<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	OM_Observation (feature type)			C	*	data
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	field	Values that were observed.		M	*	OM_Observation (feature type)
	identifier	Cells that were observed.		M	*	ZonalIdentifier (data type)
<b>Constraints:</b>	(none)					

**Table 49 – Elements of Core Quantization Functions::QuantizationContext**

<b>Name:</b>	QuantizationContext					
<b>Definition:</b>	ObservationContext for this DataAssignmentProcess.					
<b>Stereotype:</b>	Interface					
<b>Inheritance from:</b>	ObservationContext					
<b>Abstract:</b>	true					
<b>Associations:</b>	(none)					
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	role	Defines the role or roles that cells play in the quantization.		M	1	Quantization (code list)
<b>Constraints:</b>	(none)					

**Table 50 – Elements of Core Quantization Functions::QuantizationFunctions**

**Name:** QuantizationFunctions

<b>Definition:</b>	Process for quantizing external data by using a cell's geometry to sample external data and assign the results to zonal identifiers. The quantization code list identifies different potential roles for the cell geometry in the quantization process.												
<b>Stereotype:</b>	Interface												
<b>Abstract:</b>	true												
<b>Associations:</b>	(none)												
<b>Public attributes:</b>	<table border="1"> <thead> <tr> <th>Name</th> <th>Definition</th> <th>Derived</th> <th>Obligation</th> <th>Maximum occurrence</th> <th>Data type</th> </tr> </thead> <tbody> <tr> <td>cellContext</td> <td>List of roles that cell geometry is to be used in associated Data AssignmentProcesses.</td> <td></td> <td>M</td> <td>6</td> <td>Quantization (code list)</td> </tr> </tbody> </table>	Name	Definition	Derived	Obligation	Maximum occurrence	Data type	cellContext	List of roles that cell geometry is to be used in associated Data AssignmentProcesses.		M	6	Quantization (code list)
Name	Definition	Derived	Obligation	Maximum occurrence	Data type								
cellContext	List of roles that cell geometry is to be used in associated Data AssignmentProcesses.		M	6	Quantization (code list)								
<b>Constraints:</b>	(none)												

**Table 51 – Elements of Core Quantization Functions::Quantization**

<b>Name:</b>	Quantisation								
<b>Definition:</b>	CodeList for roles that cell geometries may play in a DataAssignmentProcess.								
<b>Stereotype:</b>	CodeList								
<b>Abstract:</b>	true								
<b>Associations:</b>	(none)								
<b>Values:</b>	<table border="1"> <thead> <tr> <th>Name</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>asDataTiles</td> <td>Zone assigned features clipped to the boundary of the zone's cell <b>NOTE</b> Features are clipped to the cell boundary and stored as a feature tile. <b>NOTE</b> The zone's zonal identifier may be used in the naming convention for data tiles.</td> </tr> <tr> <td>asDataCells</td> <td>Zone assigned a data value, either resampled or remapped, from a feature based on the geometry of the cell.</td> </tr> <tr> <td>asCoordinates</td> <td>Each coordinate tuple in a vector feature's geometry data is replaced by a zonal identifier.</td> </tr> </tbody> </table>	Name	Definition	asDataTiles	Zone assigned features clipped to the boundary of the zone's cell <b>NOTE</b> Features are clipped to the cell boundary and stored as a feature tile. <b>NOTE</b> The zone's zonal identifier may be used in the naming convention for data tiles.	asDataCells	Zone assigned a data value, either resampled or remapped, from a feature based on the geometry of the cell.	asCoordinates	Each coordinate tuple in a vector feature's geometry data is replaced by a zonal identifier.
Name	Definition								
asDataTiles	Zone assigned features clipped to the boundary of the zone's cell <b>NOTE</b> Features are clipped to the cell boundary and stored as a feature tile. <b>NOTE</b> The zone's zonal identifier may be used in the naming convention for data tiles.								
asDataCells	Zone assigned a data value, either resampled or remapped, from a feature based on the geometry of the cell.								
asCoordinates	Each coordinate tuple in a vector feature's geometry data is replaced by a zonal identifier.								

		<p>The size of zone is chosen to represent the uncertainty in the knowledge of the position represented by the coordinate tuple.</p>
	<p>asTags</p>	<p>Minimal set of zonal identifiers applied to an object.  <b>NOTE</b> The zone operates in this context as a minimum bounding container (similar to a minimum bounding rectangle in 2D) where the boundary of the zone wholly encloses a set of features assigned to that zone.  <b>NOTE</b> The refinement level of a zone index used to tag a feature (or set of features) provides an indication of the level of precision and/or the spatial extents of the feature.</p>
	<p>asGraphicCells</p>	<p>Zone assigned the color of a pixel in a map image ready for delivery to a map display system, with the color representing an attribute value, either sampled or mapped, from a feature.  <b>NOTE</b> Refinement levels may be aligned with zoom levels or scales in a map display system.</p>
	<p>asGraphicTiles</p>	<p>Tiling scheme used for a map display system using cells to define the tile boundaries.  <b>NOTE</b> Graphic tiles may be cached or stored and sent to the display system as a map tile.</p>

The following requirement applies:

#### REQUIREMENT 16

[www.opengis.net/spec/DGGS/2.0/req/core/functions/quantization](http://www.opengis.net/spec/DGGS/2.0/req/core/functions/quantization)

*A DGGS specification shall define quantization operations for assigning data from external sources to DGGS cells that conform to the data model in Figure 14 and definitions in tables Table 47–Table 51.*

### 8.3.3. Core Topological Query Functions module

#### 8.3.3.1. Overview of topological query functions

The Topological Query Functions module implements the DE9IM [5] functionality defined in ISO 19107:2019 Geometry::Query2D and Geometry::Query3D for zonal topology. This



is achieved through a single interface called ZoneQuery (Figure 16, Table 53). By default ZoneQuery operates at the dimensionality of the DGGS's zones. Changes in dimensionality are controlled with an optional parameter *projectTo* that constrains ZoneQuery to a specified reference direction, surface or volume.

Two additional operations are provided that are based on the temporal concept of relative position. These are called *relatePosition* and *relativePosition*. These are generalized for use on any single reference direction specified by *projectTo*, not just the temporal direction.

Six additional operations are provided in ZoneQuery that leverage the ZoneClass hierarchy. These are called *parent*, *child* and *sibling* and *parentOf*, *childOf* and *siblingOf*.

ZoneQuery operations are defined in Figure 16 and Clause 8.3.4. The following parameters are shared by many of the operations as specified in <Table 53

<i>another</i>	type ZonalIdentifier, mandatory. Specifies the target region for the query. In zonal query a zone's identifier provides sufficient description of its topology. ZonalIdentifier therefore takes the place of the geometry data used in Query2D and Query3D for both the source and the target.
<i>inheritID</i>	type Boolean, optional, default $\Leftarrow$ <i>False</i> When <i>inheritID</i> has a value of <i>True</i> the result «set» only contains cells for which the IDs have shared inheritance, and a value of <i>False</i> indicates that inheritance is ignored.
<i>projectTo</i>	type directPosition[4], optional, default $\Leftarrow$ (0,0,0,1) for <i>relatePosition</i> and <i>relativePosition</i> , otherwise (1,1,1,1) <i>projectTo</i> specifies an optional reference direction, surface or volume for an operation. Allowed values for each direction are 0 and 1, and spatial directions may also have a value of <i>n</i> . <i>projectTo</i> defines a vector whose starting point is inferred as the point with each <i>projectTo</i> direction whose value is 1 set to 0. It takes one of three forms. In its one-dimensional form for specifying a reference direction, one direction has a value of 1. For example (0,0,0,1) projects to the temporal axis, and (0,0,1,0) projects to the vertical axis. In its two-dimensional form for specifying a reference surface, two directions have a value of 1. For example a surface at height <i>n</i> is specified by a <i>projectTo</i> value of (1,1, <i>n</i> ,0) representing the vector [ (0,0, <i>n</i> ,0), (1,1, <i>n</i> ,0) ]. In its three-dimensional form for specifying a reference volume, three directions have a value of 1. For example (1,1,1,0) projects to a spatial volume without reference to time, and (1,1, <i>n</i> ,1) projects to a surface spanning all time at height <i>n</i> . Only the one-D\dimensional form is supported by <i>relativePosition</i> and <i>relatePosition</i> .

<i>rangeRefine</i>	<p>While this construct could be used to implement more complex spatio-temporal queries, that isn't the intent of Query2D, and isn't specified for ZoneQuery either.</p> <p>type refinementLevelRange, optional, default <math>\Leftarrow</math> [ min(source.refinementLevel,target.refinementLevel) : max(source.refinementLevel,target.refinementLevel) ].</p> <p>Specifies the range of refinement levels to include in the return «set». The lower and upper bounds in the refinementLevelRange datatype are both included in the range.</p>
<i>levels</i>	<p>type Integer, optional, default <math>\Leftarrow</math> 1</p> <p>Levels indicates the relative number of levels in the hierarchy to be traversed in assembling the result «set» Figure 15 illustrates the parent, child and sibling suite of functions through examples.</p>

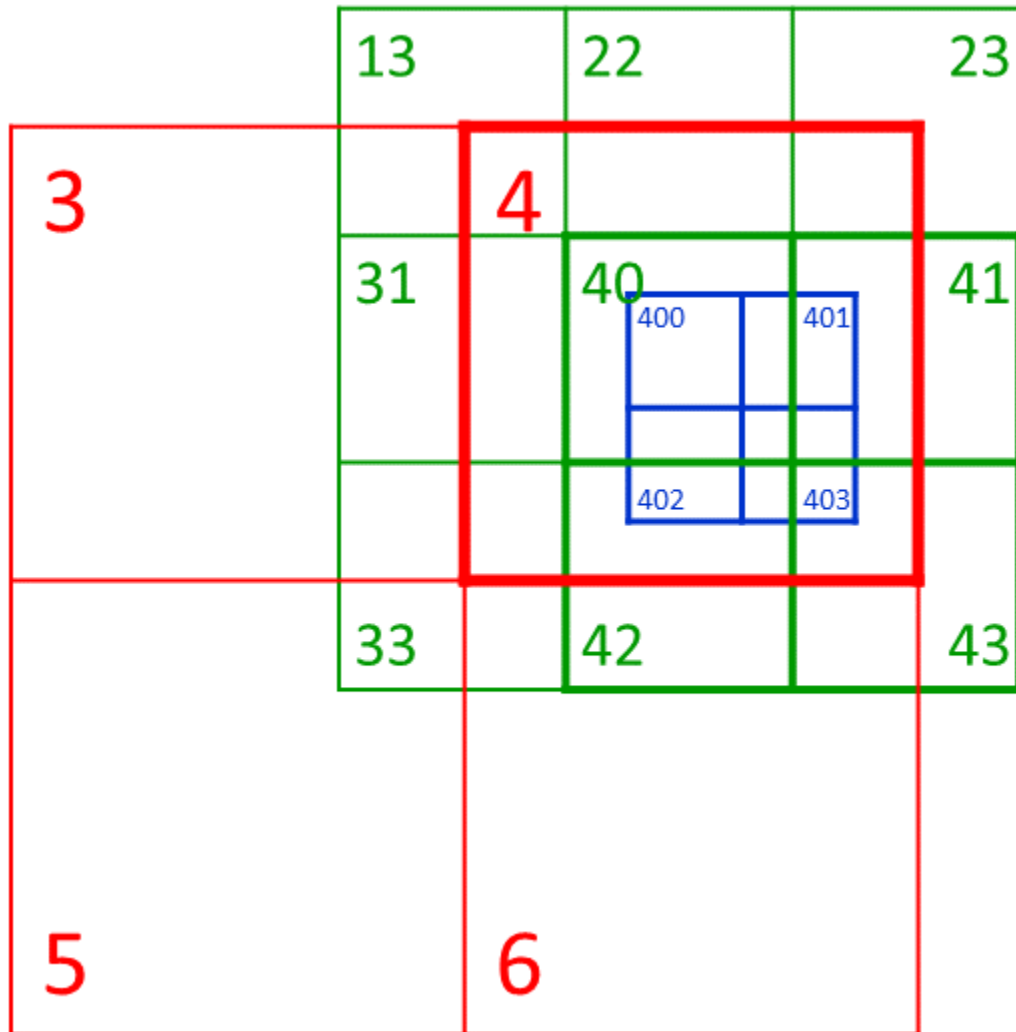
### 8.3.3.2. Summary of operations in ZoneQuery

The following operations have the same topological meaning as their equivalent operations in ISO 19107:2019 Geometry::Query2D and Geometry::Query3D: distance, contains, crosses, disjoint, equals, intersects, overlaps, touches, within, withinDistance, difference, intersection, symDifference, union, and relate.

1. **relativePosition:** *A.relativePosition(B, projectTo)*, returns the relativePosition enumerator that describes *B*'s relative position to *A* with respect to the direction defined by *projectTo*.
2. **relatePosition:** *A.relatePosition(B, relation, projectTo)* returns whether *B* has the relative position to *A* given by *relation* with respect to the direction *projectTo*
3. **parentOf:** *A.parentOf(B, inheritID, projectTo)*, returns whether *A* is a parent cell of *B*, optionally filtered by *inheritID* and *projectTo*.
4. **childOf:** *A.childOf(B, inheritID, projectTo)*, returns whether *A* is a child cell of *B*, optionally filtered by *inheritID* and *projectTo*.
5. **siblingOf:** *A.siblingOf(B, inheritID, projectTo)*, returns whether *A* is a sibling cell of *B*, optionally filtered by *inheritID* and *projectTo*.
6. **parent:** *A.parent(inheritID, projectTo, levels)*, returns the unique «set» of zoneIdentifiers for zones that satisfy *A.parentOf(B, inheritID, projectTo)* applied recursively *levels* times up the parent hierarchy. The «set» will have at most one member from each level of the hierarchy if *inheritID* is *True*, and may have more than one if the cell refinementStrategy is not nested.
7. **child:** *A.child(inheritID, projectTo, levels)*, returns the unique «set» of zoneIdentifiers for zones that satisfy *A.parentOf(B, inheritID, projectTo)* applied recursively *levels* times down the parent hierarchy.

8. **sibling:** *A.sibling(inheritID, projectTo, levels)*, returns the unique «set» of zoneIdentifiers for zones that satisfy *A.parentOf(B, inheritID, projectTo)* applied recursively *levels* times outward on zones at the same refinement level. Multiple levels of sibling can also be thought of as the children of its parent(s) the specified number of levels up the hierarchy.

Figure 15 – Examples of parent, child, and sibling query operations



EXAMPLES

*Parent, sibling and child queries for zone 40:*

*40.parent()* returns {4}

*40.sibling(inheritID=true)* returns {40, 41, 42, 43}

*40.sibling()* returns {13, 22, 23, 31, 40, 41, 42, 43}

*40.child(inheritID=true)* returns {400, 401, 402, 403}

*Parent, sibling and child tests for zone 40:*

*40.parentOf(400)* returns true

*40.siblingOf(41)* returns true

*40.siblingOf(31, inheritID=true)* returns false

*40.siblingOf(41)* returns true

**40**.childOf(4) returns *true*

*Parent and child tests for zones 31 and 33 with multiple parents:*

**31**.childOf(4, *inheritID=true*) returns *false*

**31**.childOf(4) returns *true*

**31**.parent(*inheritID=true*) returns {**3**}

**31**.parent() returns {**3, 4**}

**33**.parent() returns {**3, 4, 5, 6**}

*Parent and child queries with levels set to a value greater than 1:*

**400**.parent(*inheritID=true, level=2*) returns {**40, 4**}

**4**.child(*inheritID=true, 2*) returns {**40, 41, 42, 43, 400, 401, 402, 403, 410, 411, 412, 413, 420, 421, 422, 423, 430, 431, 432, 433**}

**400**.sibling(*inheritID=true, levels=2*) returns {**400, 401, 402, 403, 410, 411, 412, 413, 420, 421, 422, 423, 430, 431, 432, 433**}

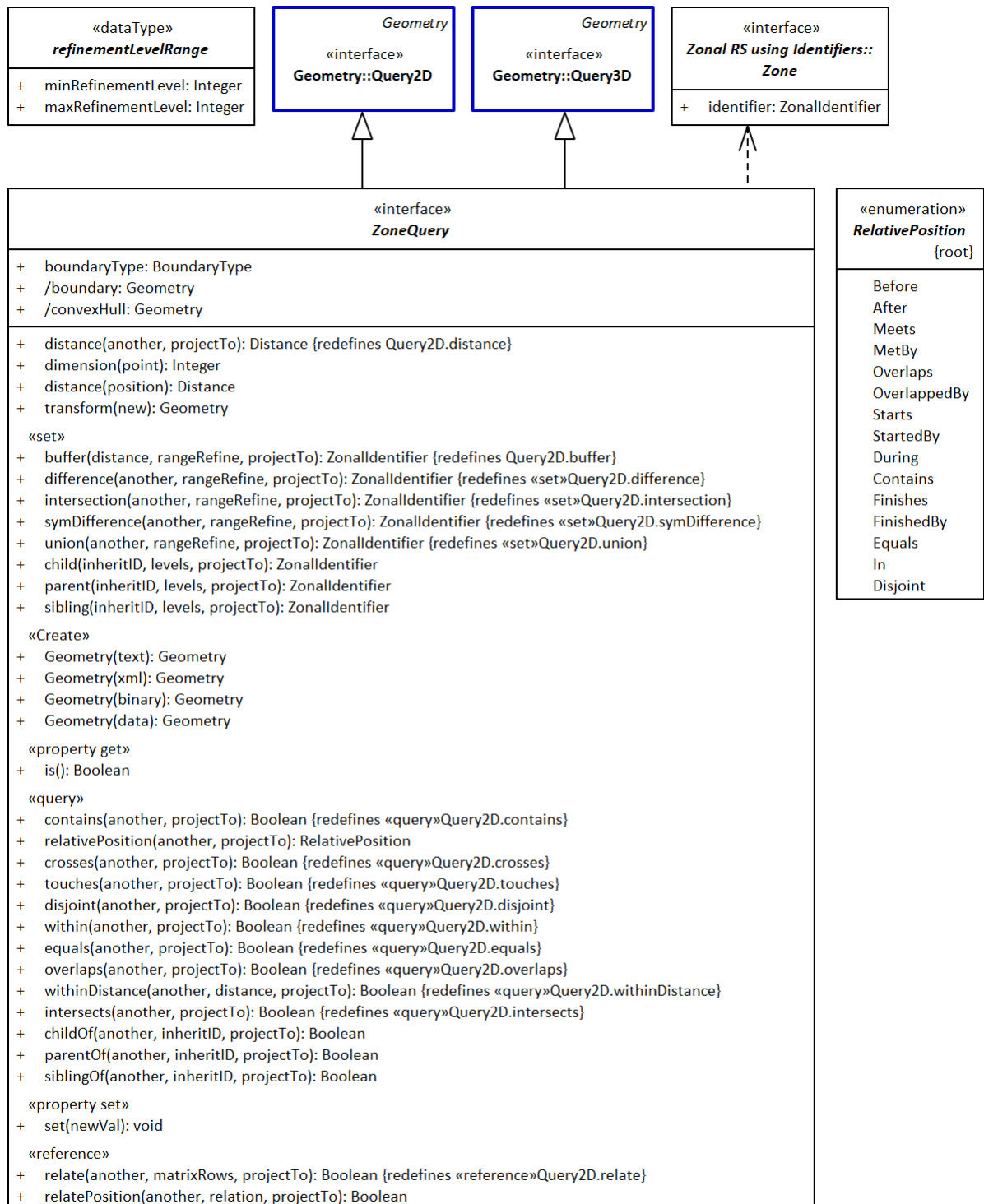
While some of these results extend to zones that are not drawn in the figure, the location indicated by their zonal identifier should be readily apparent from the pattern.

NOTE:

In all examples the optional parameters *inheritID* and *levels* take their default values of *false* and *1* respectively, unless they are specified. Since these are two-dimensional examples without any depth or time, *projectTo* has no influence.

Further query and analysis functions may then be applied to the returned data through additional software bindings. This abstract specification does not specify any requirements for the binding or implementation of further, extension, query or analytic functions.

Figure 16 – Components of Topological Zonal Query Functions module



KEY:  ISO 19107 Ed 2  ISO 19170-1

### 8.3.4. Defining tables

- a) Table 52 – Elements of Core Topological Query Functions::refinementLevelRange
- b) Table 53 – Elements of Core Topological Query Functions::ZoneQuery
- c) Table 54 – Elements of Core Query Functions::RelativePosition

**Table 52 – Elements of Core Topological Query Functions::refinementLevelRange**

**Name:** refinementLevelRange

**Definition:** Datatype to define a range of refinement levels, specified through a lower- and an upper-bound. Both bounds are included within the range. The range acts as a filter on the ZoneClass's refinementLevel attribute.

**Stereotype:** DataType

**Abstract:** true

**Associations:** (none)

	Name	Definition	Derived	Obligation	Maximum occurrence	Data type
<b>Public attributes:</b>	maxRefinementLevel	Upper-bound of the refinement level range. <b>NOTE</b> Corresponds to the most refined, and therefore smallest zones in the range.		M	1	Integer
	minRefinementLevel	Lower-bound of the refinement level range. <b>NOTE</b> Corresponds to the least refined, and therefore largest zones in the range.		M	1	Integer

Constraints: (none)

**Table 53 – Elements of Core Topological Query Functions::ZoneQuery**

Name: ZoneQuery

**Definition:** ZoneQuery redefines the DE9IM operations in Geometry::Query2D, Geometry::Query3D and provides relativePosition and relatePosition operations for the topology of zones.

**Stereotype:** Interface

**Inheritance from:** Query2D, Query3D

**Abstract:** true

**Associations:** (none)

	Name	Definition	Derived	Obligation	Maximum occurrence	Data type
<b>Public attributes:</b>	boundary	Boundary of the combined spatial geometries of the zones in the query.	true	M	1	Geometry
	boundaryType	Boundary type of the combined spatial geometries of the zones in the query.		M	1	Boundary Type (code list)
	convexHull	Convex hull of the combined spatial geometries of the zones in the query.	true	M	1	Geometry
<b>Operations:</b>	distance	(another:ZonalIdentifier, projectTo:Direct Position[4])				Distance
	«query» (1D) relativePosition	(another:ZonalIdentifier, projectTo:Direct Position[4])				RelativePosition
	«query» contains	(another:ZonalIdentifier, projectTo:Direct Position[4])				Boolean

	crosses	(another:ZonalIdentifier, projectTo:Direct Position[4])	Boolean	A.crosses(B)
	disjoint	(another:ZonalIdentifier, projectTo:Direct Position[4])	Boolean	A.disjoint(B) $\Leftrightarrow$ $A \cap B = 0$
	equals	(another:ZonalIdentifier, projectTo:Direct Position[4])	Boolean	A.equals(B) $\Leftrightarrow$ $A = B$
	intersects	(another:ZonalIdentifier, projectTo:Direct Position[4])	Boolean	A.intersects(B) $\Leftrightarrow$ $A \cap B \neq 0$
	overlaps	(another:ZonalIdentifier, projectTo:Direct Position[4])	Boolean	A.overlaps(B)
	touches	(another:ZonalIdentifier, projectTo:Direct Position[4])	Boolean	A.touches(B)
	within	(another:ZonalIdentifier, projectTo:Direct Position[4])	Boolean	A.within(B) $\Leftrightarrow$ B.contains(A)
	withinDistance	(another:ZonalIdentifier, dist:Distance, projectTo: DirectPosition[4])	Boolean	A.withinDistance(B) $\Leftrightarrow$ A. distance(B) < dist
	parentOf	(another:ZonalIdentifier, inheritID:Boolean, projectTo:Direct Position[4])	Boolean	A.parentOf(B)
	childOf	(another:ZonalIdentifier, inheritID:Boolean, projectTo:Direct Position[4])	Boolean	A.childOf(B)
	siblingOf	(another:ZonalIdentifier, inheritID:Boolean, projectTo:Direct Position[4])	Boolean	A.siblingOf(B)
	buffer	(dist:Distance, project To:DirectPosition[4])	ZonalIdentifier	A.buffer(dist)
«set»	difference	(another:ZonalIdentifier, rangeRefine:refinement LevelRange, projectTo: DirectPosition[4])	ZonalIdentifier	A.difference(B) $\Leftrightarrow$ A-B



	intersection	(another:ZonalIdentifier, rangeRefine:refinement LevelRange, projectTo: DirectPosition[4])	ZonalIdentifier	$A.intersection(B) \Leftrightarrow A \cap B$
	symDifference	(another:ZonalIdentifier, rangeRefine:refinement LevelRange, projectTo: DirectPosition[4])	ZonalIdentifier	$A.symDifference(B) \Leftrightarrow (A-B) \cup (B-A)$
	union	(another:ZonalIdentifier, rangeRefine:refinement LevelRange, projectTo: DirectPosition[4])	ZonalIdentifier	$A.union(B) \Leftrightarrow A \cup B$
	parent	(inheritID:Boolean, levels:Integer, project To:DirectPosition[4])	ZonalIdentifier	A.parent(B)
	child	(inheritID:Boolean, levels:Integer, project To:DirectPosition[4])	ZonalIdentifier	A.child(B)
	sibling	(inheritID:Boolean, levels:Integer, project To:DirectPosition[4])	ZonalIdentifier	A.sibling(B)
«reference» (1D)	relatePosition	(another:ZonalIdentifier, relate:RelativePosition, projectTo:Direct Position[4])	Boolean	A.relatePosition(B,enum,(0,0,1,0))
«reference»	relate	(another:ZonalIdentifier, matrix:Character String, projectTo:Direct Position[4])	Boolean	A.relate(B,matrix)
<b>Constraints:</b> (none)				

**Table 54 – Elements of Core Topological Query Functions::RelativePosition enumeration**

**Name:** RelativePosition

**Definition:** Enumeration for the relative position of two geometries projected to a single uni-directional dimension, e.g. time.  
**NOTE** In this document the relative position names follow those jointly adopted by W3D and OGC OGC 16-071r3, which are more recent than ISO 19108:2002

**Stereotype:** Enumeration

<b>Abstract:</b>	true																																
<b>Associations:</b>	(none)																																
	<p><i>Definition:</i> self and another are two Periods (or 1D projected Zone Geometries).</p>																																
	<table border="1"> <tr> <td>Name</td> <td></td> </tr> <tr> <td>Before</td> <td>self.end &lt; another.begin</td> </tr> <tr> <td>After</td> <td>self.begin &gt; another.end</td> </tr> <tr> <td>Meets</td> <td>self.end = another.begin</td> </tr> <tr> <td>MetBy</td> <td>self.begin = another.end</td> </tr> <tr> <td>Overlaps</td> <td>self.begin &lt; another.begin AND self.end &gt; another.begin AND self.end &lt; another.end</td> </tr> <tr> <td>OverlappedBy</td> <td>self.begin &lt; another.end AND self.end &gt; another.end</td> </tr> <tr> <td>Starts</td> <td>self.begin = another.begin AND self.end &lt; another.end</td> </tr> <tr> <td>StartedBy</td> <td>self.begin = another.begin AND self.end &gt; another.end</td> </tr> <tr> <td>During</td> <td>self.begin &gt; another.begin AND self.end &lt; another.end</td> </tr> <tr> <td>Contains</td> <td>self.begin &lt; another.begin AND self.end &gt; another.end</td> </tr> <tr> <td>Finishes</td> <td>self.end = another.end AND self.begin &gt; another.begin</td> </tr> <tr> <td>FinishedBy</td> <td>self.begin &gt; another.begin AND self.end = another.end</td> </tr> <tr> <td>Equals</td> <td>self.begin = another.begin AND self.end = another.end</td> </tr> <tr> <td>In</td> <td>self.relativePosion(another) IN [Starts, During, Finishes]</td> </tr> <tr> <td>Disjoint</td> <td>self.relativePosion(another) IN [Before, After]</td> </tr> </table>	Name		Before	self.end < another.begin	After	self.begin > another.end	Meets	self.end = another.begin	MetBy	self.begin = another.end	Overlaps	self.begin < another.begin AND self.end > another.begin AND self.end < another.end	OverlappedBy	self.begin < another.end AND self.end > another.end	Starts	self.begin = another.begin AND self.end < another.end	StartedBy	self.begin = another.begin AND self.end > another.end	During	self.begin > another.begin AND self.end < another.end	Contains	self.begin < another.begin AND self.end > another.end	Finishes	self.end = another.end AND self.begin > another.begin	FinishedBy	self.begin > another.begin AND self.end = another.end	Equals	self.begin = another.begin AND self.end = another.end	In	self.relativePosion(another) IN [Starts, During, Finishes]	Disjoint	self.relativePosion(another) IN [Before, After]
Name																																	
Before	self.end < another.begin																																
After	self.begin > another.end																																
Meets	self.end = another.begin																																
MetBy	self.begin = another.end																																
Overlaps	self.begin < another.begin AND self.end > another.begin AND self.end < another.end																																
OverlappedBy	self.begin < another.end AND self.end > another.end																																
Starts	self.begin = another.begin AND self.end < another.end																																
StartedBy	self.begin = another.begin AND self.end > another.end																																
During	self.begin > another.begin AND self.end < another.end																																
Contains	self.begin < another.begin AND self.end > another.end																																
Finishes	self.end = another.end AND self.begin > another.begin																																
FinishedBy	self.begin > another.begin AND self.end = another.end																																
Equals	self.begin = another.begin AND self.end = another.end																																
In	self.relativePosion(another) IN [Starts, During, Finishes]																																
Disjoint	self.relativePosion(another) IN [Before, After]																																
<b>Values:</b>																																	

The following requirement applies:

### REQUIREMENT 17

[www.opengis.net/spec/DGGS/2.0/req/core/functions/query/zonequery](http://www.opengis.net/spec/DGGS/2.0/req/core/functions/query/zonequery)

## REQUIREMENT 17

A DGGS specification shall implement query operations that conform to the data model in Figure 16 and classes defined in Table 52–Table 54, across its entire domain.

### 8.3.5. Core Interoperation Functions module

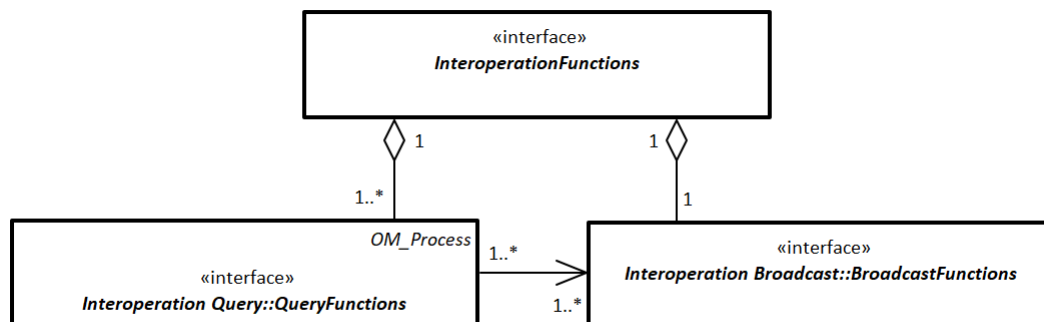
#### 8.3.5.1. Overview of interoperation functions

While the quantization and topological query functions enable a DGGS implementation to successfully operate internally, in order to facilitate connectivity with other spatial data infrastructures additional interoperation functions are required. As shown in Figure 17, the interoperation functions are split into two modules:

- a) Interoperation Query: Interpret and translate external data queries sent to the DGGS implementation; and
- b) Interoperation Broadcast: Convert the result set returned from a DGGS query operation from internal data format(s) (optimized for that DGGS implementation) to format(s) suitable for external data delivery.

This document does not specify the specific interface protocol encodings required to connect a DGGS implementation to an external client and to facilitate the transfer of information into and out of a DGGS. This abstract specification makes use of the tools available in ISO 19156:2011 to facilitate the linkage between external query operations and the data/observations assigned to the DGGS zone(s) of interest. Specific interface encodings are anticipated to be elaborated as extensions to this abstract specification.

Figure 17 – Components of Core Interoperation Functions module



### 8.3.5.2. Defining tables

- a) Table 55 – Elements of Interoperation::InteroperationFunctions

**Table 55 – Elements of Interoperation::InteroperationFunctions**

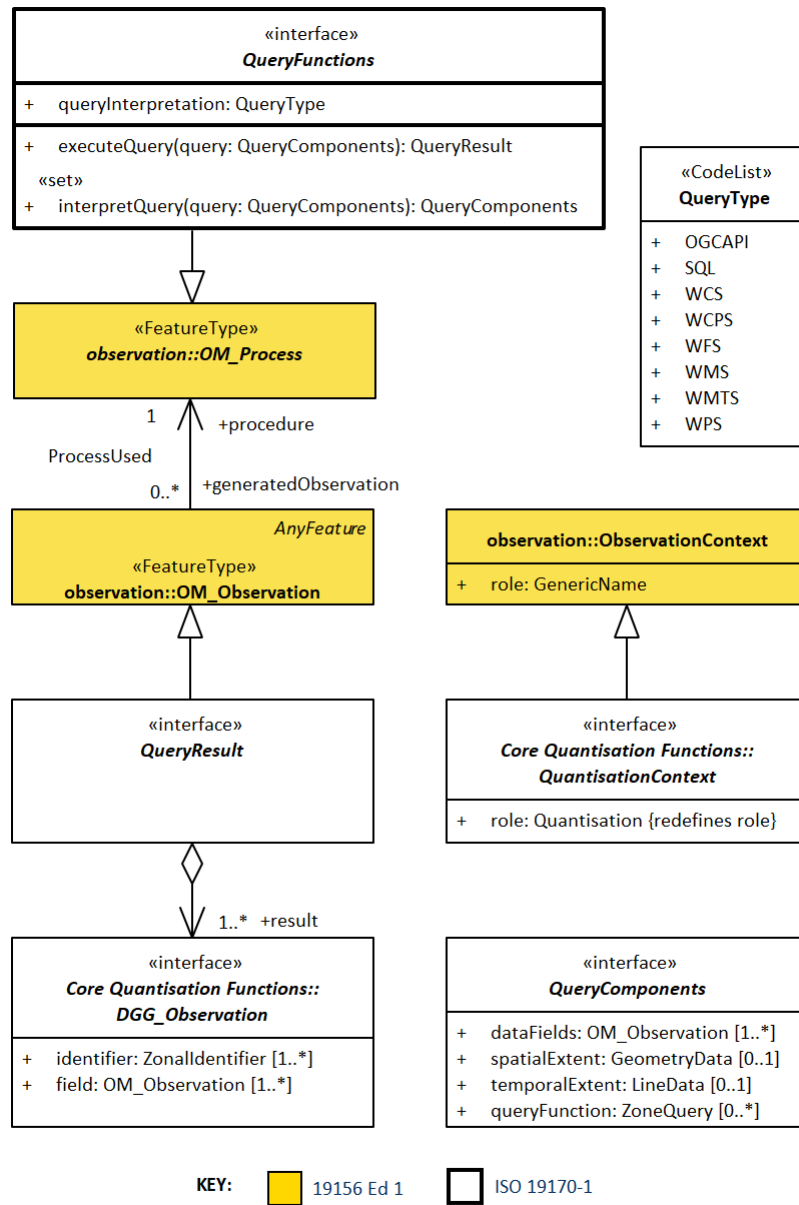
Name:	InteroperationFunctions
Definition:	Interoperation is modelled as receipt of a query from an external service and broadcast of results.
Stereotype:	Interface
Abstract:	true
Associations:	(none)
Public attributes:	(none)
Constraints:	(none)

### 8.3.5.3. Interoperation Query module

External queries may originate from an external client application and range in syntax from natural language queries (e.g. 'Where are the gas pipelines in Western Canada located?', or 'How has the Murray-Darling Basin in Australia changed over the past 27 years?', or 'Compute the watershed area of the Kowarau Catchment in New Zealand'), to an OWS 'GetCapabilities' or similar type of query, to an SQL (or similar) statement. To support interoperability, a DGGS specification shall define methods to receive, interpret and translate an external data query (or process) request into a form that can be processed by the internal DGGS data retrieval and query functions.

Figure 18 shows the key functional elements required for DGGS to translate and execute an external query or process operations.

Figure 18 – Components of Interoperation Query module



### 8.3.5.4. Defining tables

- a) Table 56 – Elements of Interoperation Query::QueryComponents
- b) Table 57 – Elements of Interoperation Query::QueryFunctions
- c) Table 58 – Elements of Interoperation Query::QueryResult
- d) Table 59 – Elements of Interoperation Query::QueryType

**Table 56 – Elements of Interoperation Query::QueryComponents**

**Name:** QueryComponents

**Definition:** Structure to hold parameters for a query.

**Stereotype:** Interface

**Abstract:** true

**Associations:** (none)

	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
<b>Public attributes:</b>	dataFields	Descriptor of the non-spatial information that is requested from the DGGs.		M	*	OM_ Observation (feature type)
	queryFunction	Spatio-temporal extent of the region of interest for the query, expressed as a ZoneQuery expression.		C	*	ZoneQuery
	spatialExtent	Spatial extent of the region of interest for the query, expressed as GeometryData.		C	1	GeometryData (data type)
	temporalExtent	Temporal extent of the region of interest for the query, expressed as temporal geometry.		C	1	LineData (data type)

**Constraints:** (none)

**Table 57 – Elements of Interoperation Query::QueryFunctions**

<b>Name:</b>	QueryFunctions					
<b>Definition:</b>	QueryFunctions is an interface to receive, interpret, and execute queries from external services. Queries for broadcast are modelled as OM_Process, that makes an OM_Observation of a set of Cells and their associated DGG_Observations to create a QueryResult. The ObservationContext:role is selected from the Quantisation CodeList.					
<b>Stereotype:</b>	Feature					
<b>Inheritance from:</b>	OM_Process					
<b>Abstract:</b>	true					
<b>Associations:</b>	(none)					
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	queryInterpretation	Identifies the language type for query.		M	1	Query Type (code list)
<b>Operations:</b>	<i>Name</i>	<i>Parameters: ParameterType</i>	<i>Return type</i>	<i>Definition</i>		
	interpretQuery	(query:Query Components)	Query Components	Transform query components from an external structure to a «set» of one or more query components structured for execution by the DGGs.		
	executeQuery	(query:Query Components)	QueryResult	Execute query to generate a result.		
<b>Constraints:</b>	(none)					

**Table 58 – Elements of Interoperation Query::QueryResult**

<b>Name:</b>	QueryResult
<b>Definition:</b>	Abstract placeholder for a query result.
<b>Stereotype:</b>	Feature
<b>Inheritance from:</b>	OM_Observation

**Abstract:** true

<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	DGG_Observation (feature type)	M	*	result
<b>Public attributes:</b>	(none)			

**Constraints:** (none)

**Table 59 – Elements of Interoperation Query::QueryType**

**Name:** QueryType

**Definition:** CodeList for the structure of an interoperation query.

**Stereotype:** CodeList

**Abstract:** true

**Associations:** (none)

	<i>Name</i>	<i>Definition</i>
<b>Values:</b>	OGCAPI	OGC API query
	SQL	Structured Query Language query.
	WCS	Web Coverage Service query.
	WCPS	Web Coverage Processing Service query.
	WFS	Web Feature Service query.
	WMS	Web Map Processing Service query.
	WMTS	Web Map Tile Service query.
	WPS	Web Processing Service query.



The following requirement applies:

### REQUIREMENT 18

[www.opengis.net/spec/DGGS/2.0/req/core/functions/interoperation/query](http://www.opengis.net/spec/DGGS/2.0/req/core/functions/interoperation/query)

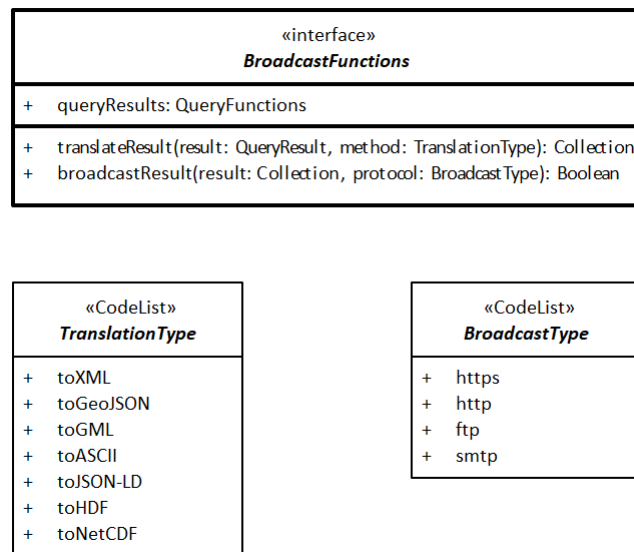
*A DGGS specification shall implement operations to read, interpret and execute external data queries that conform to the data model in Figure 17 and Figure 18 and definitions in Table 55–Table 59.*

### 8.3.5.5. Interoperation Broadcast module

Just as it is necessary for DGGS to be able to interpret and execute external data queries, DGGS shall also define methods to broadcast results from data queries to external client(s) or data infrastructure(s). External clients are anticipated to be web-based client(s), software client(s) on the same ICT infrastructure as the DGGS, or other DGGS.

Figure 19 shows basic elements required to translate the result set(s) returned from a DGGS data query into a suitable data format for transfer and broadcast the reformatted result set via one or a number of data or information transfer protocols.

Figure 19 – Interoperation Broadcast



### 8.3.5.6. Defining tables

- Table 60 – Elements of Interoperation Broadcast::BroadcastFunctions
- Table 61 – Elements of Interoperation Broadcast::TranslationType

c) Table 62 – Elements of Interoperation Broadcast::TranslationType

**Table 60 – Elements of Interoperation Broadcast::BroadcastFunctions**

**Name:** BroadcastFunctions

**Definition:** An interface to translate the results from an Interoperation Query::QueryFunctions into the requested format and return it to the client.

**Stereotype:** Interface

**Abstract:** true

**Associations:** (none)

	Name	Definition	Derived	Obligation	Maximum occurrence	Data type
<b>Public attributes:</b>	queryResults	Query result received from the Query Functions ready for processing for export.		M	1	QueryFunctions (feature type)
	Name	Parameters: ParameterType	Return type	Definition		
<b>Operations:</b>	translateResult	(result:Query Result)	Collection	Reformat query result from internal structure to requested format for broadcast.		
	broadcastResult	(result:Collection)	Boolean	Broadcast reformatted result to client using designated protocol, and return acknowledgement of success.		

**Constraints:** (none)

**Table 61 – Elements of Interoperation Broadcast::BroadcastType**

**Name:** BroadcastType

**Definition:** CodeList for DGGs interoperation data broadcast protocols

**Stereotype:** CodeList

**Abstract:** true

**Associations:** (none)

	<i>Name</i>	<i>Definition</i>
<b>Values:</b>	https	Broadcast over Hypertext transfer secure protocol.
	http	Broadcast over Hypertext transfer protocol.
	ftp	Broadcast over File transfer protocol.

**Table 62 – Elements of Interoperation Broadcast::TranslationType**

**Name:** TranslationType

**Definition:** CodeList for DGGS interoperation data broadcast translation formats.

**Stereotype:** CodeList

**Abstract:** true

**Associations:** (none)

	<i>Name</i>	<i>Definition</i>
<b>Values:</b>	toASCII	Translate to ASCII format.
	toGeoJSON	Translate to GeoJSON format.
	toGML	Translate to GML format.
	toHDF	Translate to HDF format.
	toJSON-LD	Translate to JSON-LD format.
	toNetCDF	Translate to NetCDF format.
	toXML	Translate to XML format.

The following requirement applies:

## REQUIREMENT 19

[www.opengis.net/spec/DGGS/2.0/req/core/functions/interoperation/broadcast](http://www.opengis.net/spec/DGGS/2.0/req/core/functions/interoperation/broadcast)

---

*A DGGS specification shall implement operations to translate data results from interoperation queries to standard data formats and broadcast the reformatted result set that conform to the data model in Figure 17 and Figure 19, and definitions in Table 55 and Table 60–Table 62.*

---

9

# DGGS EQUAL AREA EARTH REFERENCE SYSTEM PACKAGE

---

# DGGS EQUAL AREA EARTH REFERENCE SYSTEM PACKAGE

---

## 9.1. EQUAL-AREA EARTH DGGS PACKAGE

---

### 9.1.1. Equal-Area Earth RS module

This clause specifies the Equal-Area Earth RS module.

For a DGGS RS that is conformant with the DGGS Core to be an EAERS, it shall satisfy all the requirements for the Equal-Area Earth DGGS package.

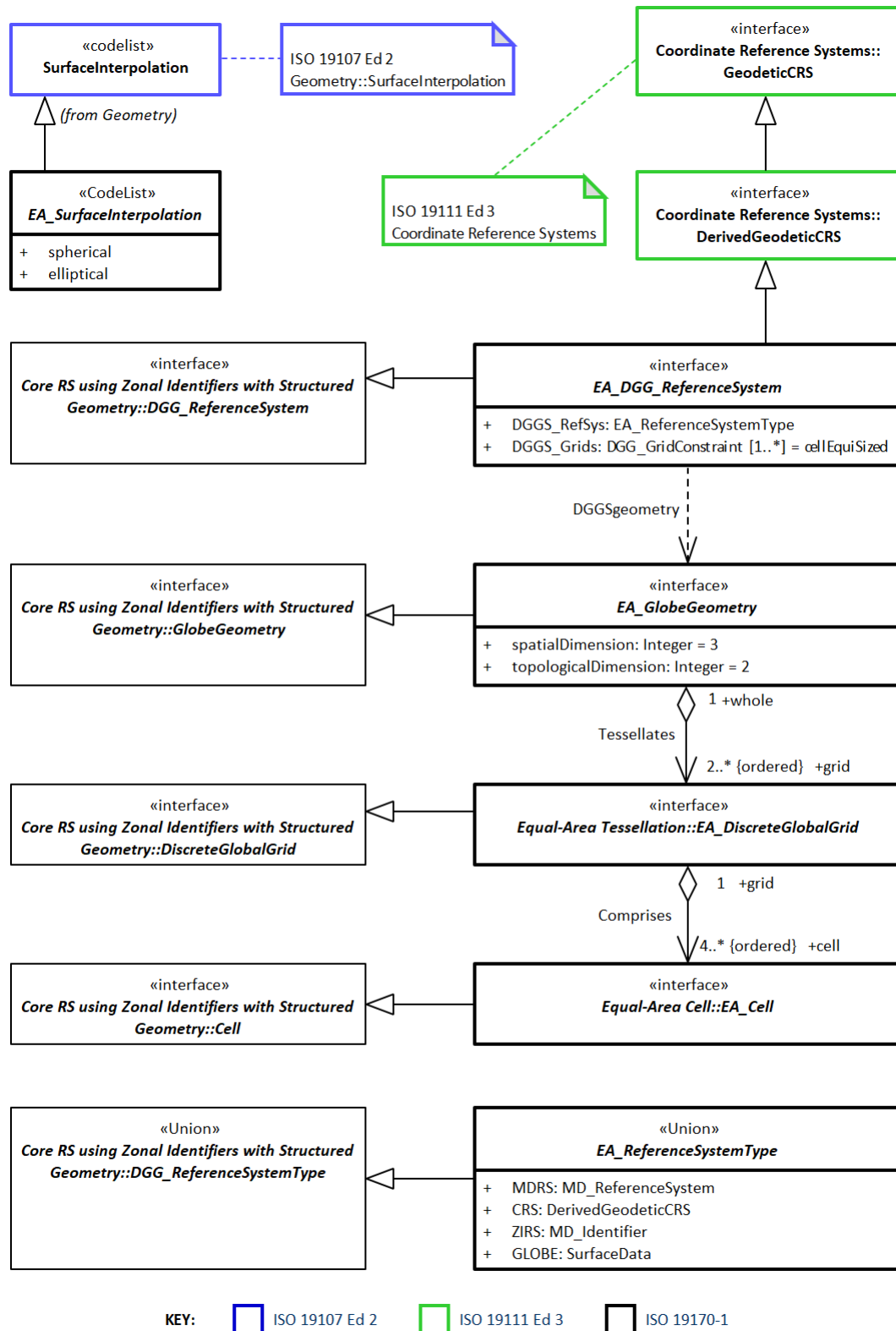
The data model supports DGGS EAERS based on either *static* or *dynamic* datums. Care shall be taken when implementing DGGS with static datums. Static CRS are by their nature only intended for use on one tectonic plate. However the definition for a DGGS always has a global domain. This apparent conundrum is resolved in two ways:

1. **Orientation:** Noting that some static CRS have one or more points on the Earth's surface where the underlying mathematics is poorly behaved, orient the EA\_BaseUnitPolyhedron so that areas, centroids, vertices, and edges can be computed; and
2. **Precision:** On the tectonic plate where the reference frame is static choose the level of DGGS precision to suit the intended use of the DGGS (typically in the range of millimetres to 10s of metres), and in areas of the Earth on plates that are moving with respect to the static reference frame choose the level of DGGS precision to reflect the larger of the constraints imposed by the mathematics and by plate tectonics (typically in the of range 10s to 100 000s of metres).

Dynamic datums present problems of a different form. The most frequently used is WGS 84, which is a datum ensemble. Care shall be taken to ensure that the error budget for EA\_Cell's area is set at a value that caters to the level of approximation in the WGS 84 definition.

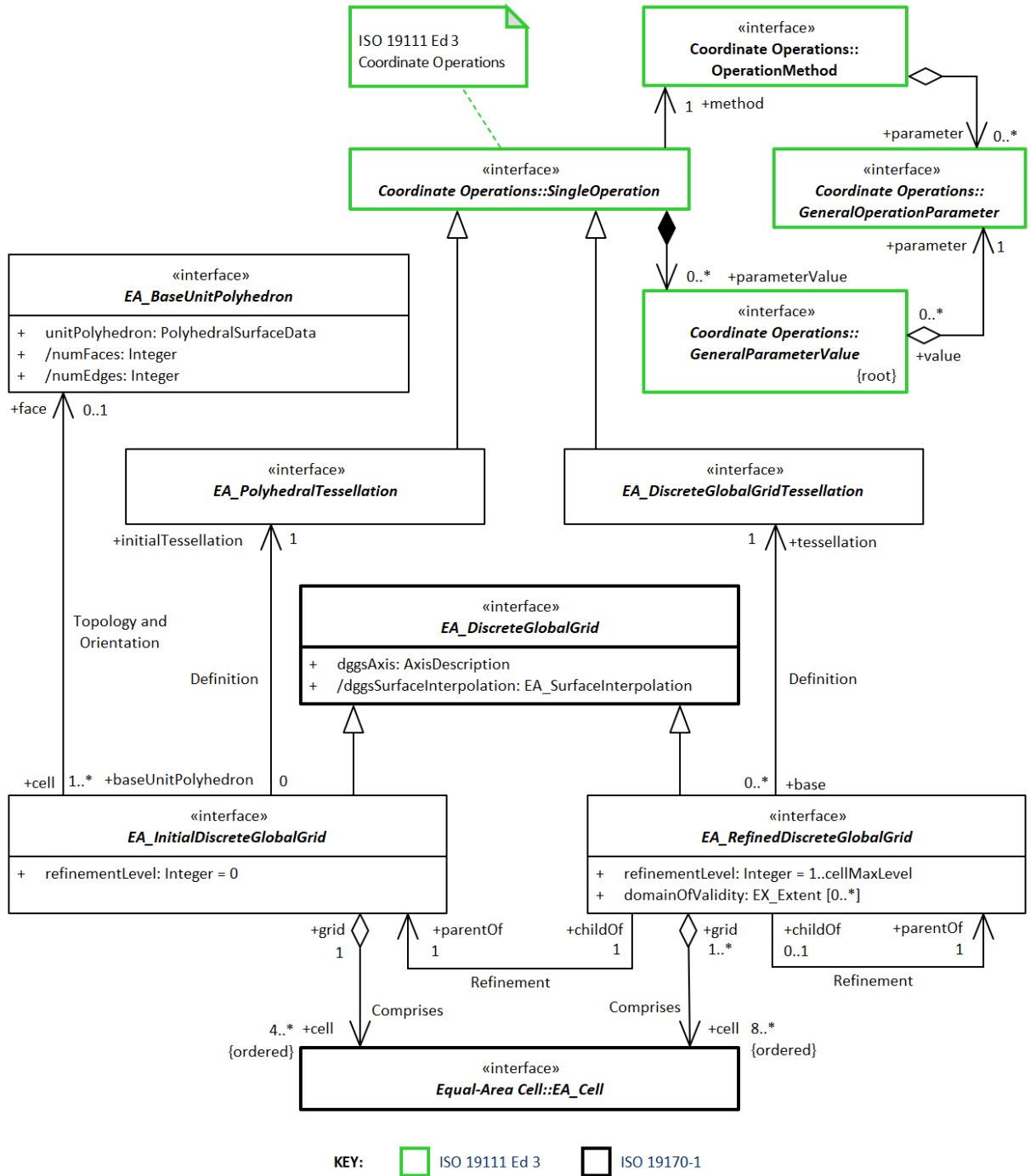
Figure 20, Figure 21 and Figure 22 show the data models for modules in the Equal-Area Earth DGGS package.

Figure 20 – Components of Equal-Area Earth RS module



See Clause 9.1.2 for definitions of elements in this Figure.

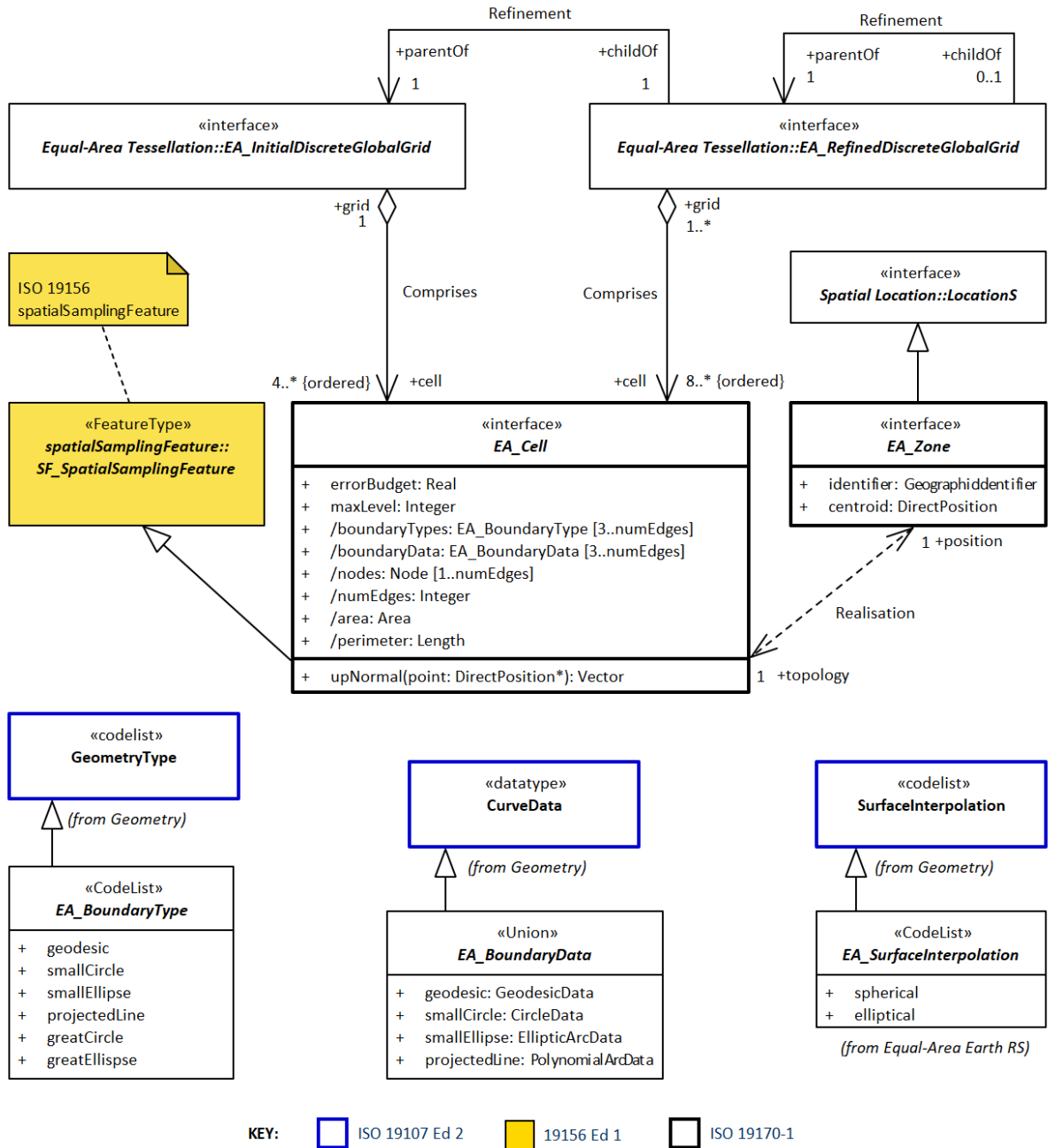
Figure 21 – Components of Equal-Area Tessellation module



See Clause 9.1.4.5 for definitions of elements in this Figure.



Figure 22 – Components of Equal-Area Cell module



See Clause 9.1.5.4 for definitions of elements in this Figure.

### 9.1.2. Defining tables

- a) Table 63 – Elements of Equal-Area Earth RS::EA\_DGG\_ReferenceSystem

- b) Table 64 – Elements of Equal-Area Earth RS::EA\_GlobeGeometry
- c) Table 65 – Elements of Equal-Area Earth RS::EA\_ReferenceSystemType
- d) Table 66 – Elements of Equal-Area Earth RS::EA\_SurfaceInterpolation

**Table 63 – Elements of Equal-Area Earth RS::EA\_DGG\_ReferenceSystem**

<b>Name:</b>	EA_DGG_ReferenceSystem					
<b>Definition:</b>	Defining characteristics of a RS using Zonal Identifiers with Structured Geometry for an Equal-Area Earth DGGs.					
<b>Stereotype:</b>	Interface					
<b>Inheritance from:</b>	DerivedGeodeticCRS, DerivedGeodeticCRS, DGG_ReferenceSystem					
<b>Abstract:</b>	true					
<b>Associations:</b>	(none)					
	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
<b>Public attributes:</b>	DGGS_Grids	List of characteristics that constrain the grid cells in this DGGS in decreasing order of priority. cell EquiSized shall be the first value.		M	*	DGG_GridConstraint (code list)
	DGGS_RefSys	Reference system metadata,		M	1	EA_ReferenceSystemType (union data type)
<b>Constraints:</b>	(none)					

**Table 64 – Elements of Equal-Area Earth RS::EA\_GlobeGeometry**

<b>Name:</b>	EA_GlobeGeometry					
<b>Definition:</b>	Parent geometry specifying the geometry, dimensionality and domain of the globe for this DGGs. Geometry of the surface of an Earth reference model.					
<b>Stereotype:</b>	Interface					

<b>Inheritance from:</b>	GlobeGeometry					
<b>Abstract:</b>	true					
<b>Associations:</b>	<i>Association with:</i>			<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>
	EA_DiscreteGlobalGrid (feature type)			M	*	grid
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	spatialDimension	EA_GlobeGeometry shall have a spatialDimension of 3.		M	1	Integer
	topologicalDimension	EA_GlobeGeometry shall have a topological Dimension of 2, corresponding to the surface of the Earth.		M	1	Integer
<b>Constraints:</b>	(none)					

**Table 65 – Elements of Equal-Area Earth RS::EA\_ReferenceSystemType**

<b>Name:</b>	EA_ReferenceSystemType					
<b>Definition:</b>	Defining metadata elements of the base CRS for a DGGs Equal-Area Earth RS.					
<b>Stereotype:</b>	Union					
<b>Inheritance from:</b>	DGG_ReferenceSystemType					
<b>Abstract:</b>	true					
<b>Associations:</b>	(none)					
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	CRS	Metadata required to reference coordinates, Includes CRS ID and coordinate Epoch for dynamic CRS.		M	1	DerivedGeodeticCRS

GLOBE	SurfaceData for the chosen EA_GlobeGeometry that specifies geometry, spatial, and topological dimensionality and domain of the globe for this DGGS.	M	1	SurfaceData (data type)
MDRS	Reference system information describing this whole DGGS.	M	1	MD_ReferenceSystem
ZIRS	Identifier for the spatial RSUsingZonalIdentifiers used by the DGGS.	M	1	MD_Identifier (data type)
<b>Constraints:</b>	(none)			

**Table 66 – Elements of Equal-Area Earth RS::EA\_SurfaceInterpolation class**

<b>Name:</b>	EA_SurfaceInterpolation						
<b>Definition:</b>	Subset of Geometry::SurfaceInterpolation (code list) providing permitted interpolation methods for EA_Cell						
<b>Stereotype:</b>	CodeList						
<b>Abstract:</b>	true						
<b>Associations:</b>	(none)						
<b>Values:</b>	<table border="1"> <thead> <tr> <th>Name</th> <th>Definition</th> </tr> </thead> <tbody> <tr> <td>spherical</td> <td>The EA_Cell surface is a section of a spherical surface. <b>NOTE</b> For EA_Cells of sufficiently small extent, linear interpolation is sufficient to meet the EA_Cell.errorBudget for area.</td> </tr> <tr> <td>elliptical</td> <td>The EA_Cell surface is a section of an elliptical surface. <b>NOTE</b> for EA_Cells of sufficiently small extent, spherical interpolation is sufficient to meet the EA_Cell.errorBudget for area. <b>NOTE</b> for EA_Cells of sufficiently small extent, linear interpolation is sufficient to meet the EA_Cell.errorBudget for area.</td> </tr> </tbody> </table>	Name	Definition	spherical	The EA_Cell surface is a section of a spherical surface. <b>NOTE</b> For EA_Cells of sufficiently small extent, linear interpolation is sufficient to meet the EA_Cell.errorBudget for area.	elliptical	The EA_Cell surface is a section of an elliptical surface. <b>NOTE</b> for EA_Cells of sufficiently small extent, spherical interpolation is sufficient to meet the EA_Cell.errorBudget for area. <b>NOTE</b> for EA_Cells of sufficiently small extent, linear interpolation is sufficient to meet the EA_Cell.errorBudget for area.
Name	Definition						
spherical	The EA_Cell surface is a section of a spherical surface. <b>NOTE</b> For EA_Cells of sufficiently small extent, linear interpolation is sufficient to meet the EA_Cell.errorBudget for area.						
elliptical	The EA_Cell surface is a section of an elliptical surface. <b>NOTE</b> for EA_Cells of sufficiently small extent, spherical interpolation is sufficient to meet the EA_Cell.errorBudget for area. <b>NOTE</b> for EA_Cells of sufficiently small extent, linear interpolation is sufficient to meet the EA_Cell.errorBudget for area.						

The following requirement applies:

#### REQUIREMENT 20

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/harmonized\\_model](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/harmonized_model)

---

*An EAERS specification shall comply with the data model in Figure 20, Figure 21 and Figure 22 and definitions in Table 63–Table 76.*

---

### 9.1.3. Global domain

For a DGGS RS to be an EAERS, the domain shall be defined as the entire surface of the Earth.

The following requirement applies:

#### REQUIREMENT 21

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/global\\_domain](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/global_domain)

---

*EAERS global domain – the domain of an EAERS shall be the whole surface of the reference frame's Earth model and the DGG\_ReferenceSystem shall specify cellEqualSized as one of the DGGS\_Grids constraint values.*

---

### 9.1.4. Equal-Area Tessellation module

#### 9.1.4.1. Tessellation overview

A multiresolution hierarchical tessellation of cells is created by constructing a sequence of discrete global grids, each with successively finer cell resolutions. First an initial discrete global grid is constructed as described in subclause Clause 9.1.4.2. The cells of this initial tessellation are then iteratively refined by application of cell refinement method(s) [2] to create finer resolution child cells. The initial tessellation, the cell shape, the refinement methods and indexing methods may all vary for different DGGSs.

#### 9.1.4.2. Initial tessellation

The entire surface of the Earth is partitioned to a finite/discrete set of regions. Most methods initially approximate the Earth's surface using a simple base unit polyhedron which is scaled so that all vertices are located on the surface model of the Earth, and the edges are warped so they also lie on the surface model. The resulting edges may be any of the types listed in EA\_CurveType. These include geodesics, small circles, small ellipses and lines that project to a

straight line on a plane. These all result from intersections of a plane and an ellipsoid. The initial discrete global grid tessellation has the same form as the base unit polyhedron. Each EA\_Cell of the initial tessellation represents one face of the chosen base unit polyhedron mapped to the chosen surface model of the Earth. This document refers to the initial tessellation as a “polyhedral tessellation”. The most common choices for an initial base unit polyhedron are discussed in subclause Annex C.4 [2].

The following requirement applies:

#### REQUIREMENT 22

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/initial](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/initial)

---

*An EAERS specification shall include an initial tessellation that is defined by equal area cells produced by mapping the faces of a base unit polyhedron to the surface model of the Earth.*

---

#### 9.1.4.3. Tessellation sequence

DGGS EAERS comprise a discrete sequence of global grids formed from recursive application of a tessellation method to refine the grid cells, so that each global grid has a progressively finer spatial resolution.

The following requirements apply:

#### REQUIREMENT 23

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/sequence](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/sequence)

---

*An EAERS specification shall have tessellation operations that generate a sequence of discrete global grids with progressively smaller cells.*

---

#### REQUIREMENT 24

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/sequence/max](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/sequence/max)

---

*An EAERS specification shall specify a limit to the number of iterations in its sequence of discrete global grids, to ensure that the error budget for EA\_Cell's area is not exceeded.*

---

#### 9.1.4.4. Global area preservation

Preservation of total surface area throughout the range of hierarchical tessellations is a necessary property of DGGSs in order to represent information consistently at successive

resolutions. This requirement ensures that each level of grid refinement completely covers the Earth's surface without cell overlaps.

The following requirement applies:

### REQUIREMENT 25

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/global\\_area\\_preservation](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/global_area_preservation)

*For each successive discrete global grid, an EAERS specification one of the following two statements shall apply:*

Dynamic Datum	<i>If the base CRS has a dynamic datum the sequence of discrete global grids shall preserve global domain completeness and position uniqueness throughout the sequence.</i>
Static Datum	<i>If the base CRS has a static datum the sequence of discrete global grid shall preserve domain completeness and position uniqueness throughout the sequence for all EA_Cells within their respective discrete global grids.</i>

#### 9.1.4.5. Defining tables

- a) Table 67 – Elements of Equal-Area Tessellation::EA\_BaseUnitPolyhedron
- b) Table 68 – Elements of Equal-Area Tessellation::EA\_DiscreteGlobalGrid
- c) Table 69 – Elements of Equal-Area Tessellation::EA\_DiscreteGlobalGridTessellation
- d) Table 70 – Elements of Equal-Area Tessellation::EA\_InitialDiscreteGlobalGrid
- e) Table 71 – Elements of Equal-Area Tessellation::EA\_PolyhedralTessellation
- f) Table 72 – Elements of Equal-Area Tessellation::EA\_RefinedDiscreteGlobalGrid

**Table 67 – Elements of Equal-Area Tessellation::EA\_BaseUnitPolyhedron**

**Name:** EA\_BaseUnitPolyhedron

**Definition:** EA\_BaseUnitPolyhedron  
Polyhedron with circumsphere radius of one, specified by the number and faces and edges and the collection of boundary curves that make up each polygonal face.

**Stereotype:** Interface

**Abstract:** true

Associations:		Association with:	Obligation	Maximum occurrence	Provides:		
		SC_ProjectedCRS (feature type)	M	*	singleProjCRS		
Public attributes:		Name	Definition	Derived	Obligation	Maximum occurrence	Data type
		numEdges	Number of edges in the EA_BaseUnitPolyhedron.	true	M	1	Integer
		numFaces	Number of faces on the EA_BaseUnitPolyhedron, corresponds to the number of EA_Cells in the EA_InitialDiscreteGlobalGrid.	true	M	1	Integer
		unitPolyhedron	PolygonData for each of the unit Polyhedron's segment Polygons, expressed in spherical coordinates (theta, phi) with unit radius.		M	1	PolyhedralSurfaceData (data type)

**Constraints:** self.vertex.r = 1

**Table 68 – Elements of Equal-Area Tessellation::EA\_DiscreteGlobalGrid**

**Name:** EA\_DiscreteGlobalGrid

**Definition:** Super class for EA\_InitialDiscreteGlobalGrid and EA\_RefinedDiscreteGlobalGrid

**Stereotype:** Interface

**Inheritance from:** DiscreteGlobalGrid

**Generalization of:** EA\_InitialDiscreteGlobalGrid, EA\_RefinedDiscreteGlobalGrid

**Abstract:** true

Associations:		Association with:	Obligation	Maximum occurrence	Provides:		
		EA_Cell (feature type)	M	*	cell		
Public attributes:		Name	Definition	Derived	Obligation	Maximum occurrence	Data type



dggsAxis	A dggsAxis typically follows a space-filling curve designed to recursively traverse all cells in each tessellation, Cell::identifiers are ordered along the path of the dggsAxis.	M	1	AxisDescription (data type)	
dggsSurfaceInterpolation	EA_Surface Interpolation corresponding to the form of the specified EA_GlobeGeometry.	true	M	1	EA_Surface Interpolation (code list)

**Constraints:** (none)

**Table 69 – Elements of Equal-Area Tessellation::EA\_DiscreteGlobalGridTessellation**

**Name:** EA\_DiscreteGlobalGridTessellation

**Definition:** The EA\_DiscreteGlobalGridTessellation method implements the DGGS\_Grids constraint, DGGS\_Refinement strategy and DGGS\_RefinementRatio, to create a child EA\_RefinedDiscreteGlobalGrid from a parent EA\_DiscreteGlobalGrid.

**Stereotype:** Interface

**Inheritance from:** SingleOperation, CC\_SingleOperation, SingleOperation

**Abstract:** true

**Associations:** (none)

**Public attributes:** (none)

**Constraints:** (none)

**Table 70 – Elements of Equal-Area Tessellation::EA\_InitialDiscreteGlobalGrid**

**Name:** EA\_InitialDiscreteGlobalGrid

<b>Definition:</b>	The EA_InitialDiscreteGlobalGrid is formed by applying the EA_PolyhedralTessellation method to a EA_BaseUnitPolyhedron.					
<b>Stereotype:</b>	Interface					
<b>Inheritance from:</b>	EA_DiscreteGlobalGrid					
<b>Abstract:</b>	true					
<b>Associations:</b>	<i>Association with:</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Provides:</i>		
	EA_Cell (feature type)	M	*	cell		
	EA_BaseUnitPolyhedron (feature type)	C	1	face		
	EA_PolyhedralTessellation (feature type)	M	1	initialTessellation		
<b>Public attributes:</b>	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
	refinementLevel	EA_InitialDiscreteGlobalGrid has a refinementlevel of 0.		M	1	Integer
<b>Constraints:</b>	(none)					

**Table 71 – Elements of Equal-Area Tessellation::EA\_PolyhedralTessellation**

<b>Name:</b>	EA_PolyhedralTessellation
<b>Definition:</b>	<p>The EA_PolyhedralTessellation method transforms the EA_BaseUnit Polyhedron in such a way that:</p> <ol style="list-style-type: none"> <li>1) the surface area of each cell belonging to the EA_InitialDiscreteGlobalGrid is the same,</li> <li>2) domain completeness is preserved by the EA_InitialDiscreteGlobalGrid, and</li> <li>3) location uniqueness is preserved by the EA_InitialDiscreteGlobalGrid.</li> </ol>
<b>Stereotype:</b>	Interface
<b>Inheritance from:</b>	SingleOperation, CC_SingleOperation, SingleOperation
<b>Abstract:</b>	true
<b>Associations:</b>	(none)

Public attributes: (none)

Constraints: (none)

**Table 72 – Elements of Equal-Area Tessellation::EA\_RefinedDiscreteGlobalGrid**

Name: EA\_RefinedDiscreteGlobalGrid

Definition: EA\_DiscreteGlobalGrid formed by an EA\_DiscreteGlobalGridTessellation, is a EA\_RefinedDiscreteGlobalGrid.

Stereotype: Interface

Inheritance from: EA\_DiscreteGlobalGrid

Abstract: true

	Association with:	Obligation	Maximum occurrence	Provides:
<b>Associations:</b>	EA_Cell (feature type)	M	*	cell
	EA_RefinedDiscreteGlobalGrid (feature type)	M	1	parentOf
	EA_InitialDiscreteGlobalGrid (feature type)	M	1	parentOf
	EA_DiscreteGlobalGridTessellation (feature type)	M	1	tessellation

	Name	Definition	Derived	Obligation	Maximum occurrence	Data type
<b>Public attributes:</b>	domainOfValidity	<p>For DGGs with a dynamic datum the domainOfValidity is always global.</p> <p>For DGGs with a static datum, the domainOfValidity is the extent of the continent or plate that the datum refers to.</p> <p><b>NOTE</b> Inside the domain OfValidity each cell's representative position is guaranteed to be inside the boundary of the cell. Outside the domainOfValidity plate tectonics move the earth</p>		C	*	EX_Extent (data type)

	through cells, and there is no guarantee that cells have meaningful representative positions.			
refinementLevel	The child EA_RefinedDiscrete GlobalGrid has a refinement Level of one greater than its parent.	M	1	Integer
<b>Constraints:</b> (none)				

## 9.1.5. Equal-Area Cell module

### 9.1.5.1. Cells are simple polygons

EA\_Cells are simple polygons formed from an ordered set of boundary curves adhering to the geometric reference surface of the Earth model. EA\_Cells are defined in three-dimensional space and are topologically two-dimensional. Different cell shapes can be used, typically triangle, quadrilateral, pentagon, hexagon, and octagon. Cells formed from faces of the five (5) Platonic solids (tetrahedron-triangle, cube-quadrilateral, octahedron-triangles, dodecahedron-pentagons, icosahedron-triangles) all satisfy the requirements for simple cells. Larger numbers of faces and therefore cells at the top of the hierarchy of tessellations, increases the uniformity of the cell shape. The truncated icosahedron, with thirty-two (32) faces is often used for DGGs. It produces mostly hexagons augmented by twelve (12) pentagons in each tessellation. Each cell shape has its own advantages and disadvantages [2] and it is usually desirable for each refined discrete global grid to have a majority of cells with the same shape [8],[9]. Triangular, quadrilateral and hexagonal cells are common choices used in DGGs on the surface of the Earth. These shapes provide regular tiling of the plane [8], which can be mapped to a curved surface such as a spherical or ellipsoidal Earth surface model.

The cell structures in each successive level of cell refinement are constrained by the properties of the initial tessellation, but do not necessarily have the same geometry as the initial tessellation. Simple two-dimensional cells have the following properties:

1. edges meet only at the vertices;
2. exactly two edges meet at each vertex;
3. exactly the same number of edges and vertices; and,
4. enclose a region which always has a measurable area.

The following requirement applies:

## REQUIREMENT 26

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/simple/2d\\_polygon](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/simple/2d_polygon)

---

*For each successive discrete global grid, an EAERS specification shall define EA\_Cells that are simple polygons.*

---

### 9.1.5.2. Cells referenced at their centroid

Each EA\_Cell shall be referenced at its centroid. This is because the centroid is the only location that provides a representative point that behaves consistently with shape and is invariant under orientation. The representative point for an EA\_Cell shall lie on the surface of the cell. For curved surfaces, the “centre of gravity” computed in three-dimensions cannot actually lie on the surface (see 6.4.4.8). If, however, the “centre of gravity” is computed using distances computed along geodesic arcs on the curved surface, then the centroid is constrained to lie on the surface. This is a centroid calculated as the “geodesic centre of surface area” of an EA\_Cell.

The centroid enables a dual representation of a DGGS tessellation as both two-dimensional areal cell grids and as point-based lattices of cell reference locations.

The following requirement applies:

## REQUIREMENT 27

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/direct\\_position/centroid](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/direct_position/centroid)

---

*An EAERS specification shall define the DirectPosition of an EA\_Cell to be the centroid of the EA\_cell, computed as the geodesic centre of surface area.*

---

### 9.1.5.3. Cells of equal area

This document defines an EAERS based on a hierarchy of equal area tessellations. Cells of equal area provide global grids with spatial units that (at multiple resolutions) have an equal probability of contributing to an analysis. Equal area cells also help to minimize the confounding effects of area variations in spatial analyses where the curved surface of the earth is the fundamental reference frame.

In an EAERS the unit of measure is a unit of area, not length as in a cartesian XYZ coordinate system or angle as in an ellipsoidal coordinate system. For angular measures the degree is defined as one 360th integer fraction of a full circle, and by analogy the areal unit of measure of a DGGS coordinate reference system is an integer fraction of the surface area of the chosen ellipsoidal surface model for the earth. For DGGSs the integer fraction is the number of cells in the initial discrete global grid. Continuing the angular analogy with minutes being 1/60th of a

degree and seconds being 1/60th of a minute, for DGGs, recursive use of the refinement ratio generates a sequence of smaller areal units, one for each refined discrete global grid.

Standard units of measure are defined through a reference measure with defined precision. The meter unit of length has been redefined a number of times as technology has improved. For example:

1. **From 1793:** 1/10 000 000 of the meridian through Paris between the North Pole and the Equator (+/-  $\sim 10^{-4}$  m); and
2. **From 1983:** The length of the path travelled by light in a vacuum in 1/299 792 458<sup>th</sup> of a second (+/-  $10^{-10}$  m).

To address the need for transparency of the precision in its units of area, this document uses the concept of a specified error budget. The error budget takes into account all sources of variation that are relevant to the particular DGGs, its derivation, and its expected use. For most day to day uses, most of these sources contribute errors that are significantly smaller than the precision required of the DGGs. However, as technology pushes the limits of available precision, use-cases demand combining data from highly disparate sources, and users push their expectations, it becomes increasingly important to know when limits of use are reached.

The sources considered for inclusion in the error budget can include:

1. **spatial variation:** whether the available precision of each source of error is uniform or variable across the DGGs's domain;
2. **datum:** precision of the underlying measurements for the datum;
3. **static datum:** precision of the assertion that the coordinates are static both on the tectonic plate the datum is tied to, and away from the static plate;
4. **earth model:** differences between the chosen ellipsoidal or spherical earth model and the real earth's surface. Spheres lead to faster and more precise computations for the tessellations but have a less precise fit to the real earth's surface.

While implementation issues are beyond the scope of this document, the following sources may contribute to increases in error budgets for an implementation:

1. **spherical maths:** exact solutions are available for solving the required mathematics on the surface of a sphere;
2. **ellipsoidal maths:** solving the required mathematics on the surface of an ellipsoid requires the use of iterative numerical methods which result in approximate solutions with an uncertainty determined by both the method and the number of iterations used to perform the computation;
3. **data-type precision:** most will probably use 64-bit double precision for location, anything less will have a significant impact on the error budget;

4. **cell edge type:** geodesics, small circles, small ellipses, and projected lines each have different solutions with different consequences for their precision and performance. At finer resolutions it may be possible to implement a simpler edge type and still stay within the implementation error budget;
5. **error propagation:** the choice of strategy for implementing the sequence of tessellations.

The error budget puts a limit on the number of iterations in the sequence of refined discrete global grids. This is acknowledged in the requirements.

DGGs may validly comprise more than one cell geometry. This most typically arises for systems based on truncated polyhedra such as the cuboctahedron – with both square and triangular faces, and the truncated icosahedron – with pentagonal and hexagonal faces. In these situations, equal area is interpreted to mean that all of the cells of a particular geometry are equal area, and that the ratio of the areas of the two geometries is preserved through the tessellations. For example, in the truncated icosahedron used by ISEA the ratio of pentagonal to hexagonal areas within a tessellation level is always 5/6.

The following requirements apply:

#### REQUIREMENT 28

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/equal\\_area/error\\_budget](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/equal_area/error_budget)

*An EAERS shall specify error budget(s) for cell area of 1% or less that represents the maximum ratio of a cell's area to the theoretical average cell area of all cells in a discrete global grid computed from the number of cells in the discrete global grid and the discrete global grid's surface area. One of the following two statements shall apply:*

Dynamic Datum

*If the base CRS has a dynamic datum, for each successive level of grid refinement, an EAERS specification shall specify one error budget value.*

Static Datum

*If the base CRS has a static datum, for each successive level of grid refinement, an EAERS specification shall specify at least one error budget value for the area of the globe that it is static, and at least one error budget value for the area of the globe that is not static.*

#### REQUIREMENT 29

[www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/equal\\_area](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/equal_area)

*For each discrete global grid, and for each cell geometry, an EAERS specification shall define EA\_Cells that are equal area within the specified error budget.*

#### 9.1.5.4. Defining tables

- a) Table 73 – Elements of Equal-Area Cell::EA\_BoundaryData
- b) Table 74 – Elements of Equal-Area Cell::EA\_Cell
- c) Table 75 – Elements of Equal-Area Cell::EA\_Zone
- d) Table 76 – Elements of Equal-Area Cell::EA\_BoundaryType

**Table 73 – Elements of Equal-Area Cell::EA\_BoundaryData**

**Name:** EA\_BoundaryData

**Definition:** Curve data for the permitted boundary types. greatCircle and greatEllipse boundary types use the same data types as smallCircle and smallEllipse respectively.

**Stereotype:** Union

**Inheritance from:** CurveData

**Abstract:** true

**Associations:** (none)

	<i>Name</i>	<i>Definition</i>	<i>Derived</i>	<i>Obligation</i>	<i>Maximum occurrence</i>	<i>Data type</i>
<b>Public attributes:</b>	geodesic	Curve data for a geodesic boundary.		M	1	GeodesicData (data type)
	projectedLine	Curve data for a projected line boundary.		M	1	PolynomialArcData (data type)
	smallCircle	Curve data for a small circle boundary.		M	1	CircleData (data type)
	smallEllipse	Curve data for a small ellipse boundary.		M	1	EllipticArcData (data type)

**Constraints:** (none)



**Table 74 – Elements of Equal-Area Cell::EA\_Cell**

**Name:** EA\_Cell

**Definition:** Reference system unit of geometry associated with an EA\_Zone. As part of EA\_GlobeGeometry, it has the same spatial, temporal and topological dimensionality as GlobeGeometry.  
**NOTE** As an EA\_Cell on the surface of the Earth, its topologicalDimension is 2.

**Stereotype:** Interface

**Inheritance from:** SF\_SpatialSamplingFeature, Cell

**Abstract:** true

**Associations:** (none)

	Name	Definition	Derived	Obligation	Maximum occurrence	Data type
	area	Area of the cell's surface.	true	M	1	Area
<b>Public attributes:</b>	boundaryData	List of EA_BoundaryData that make up the cells' boundary, starting with the curve connecting node_0 to node_1, and continuing in a clockwise sequence.	true	M	1	EA_Boundary Data (union data type)
	boundaryTypes	List of EA_CurveTypes that make up the cells' boundary, starting with the curve connecting node_0 to node_1, and continuing in clockwise sequence.	true	M	1	EA_Boundary Type (code list)
	errorBudget	For DGGs referencing a dynamic datum, cellEqualArea Precision will typically be a single value for each tessellation, and therefore most efficiently realised as an attribute of each RefinedDiscreteGlobalGrid. For DGGs referencing a static datum, cellAreaPrecision will have different values in different locations, primarily dependent on whether the location is on the static tectonic plate or not.		M	1	Real

Under these circumstances the cellEqualAreaPrecision may be realised as an attribute of a parent cell one or more refinement levels above the cell.

maxLevel	<p>For DGGs referencing a dynamic datum, cellMaxLevel will typically be a single value for each tessellation, and therefore most efficiently realised as an attribute of each Refined DiscreteGlobalGrid.</p> <p>For DGGs referencing a static datum, cellMaxLevel will have different values in different locations, primarily dependent on whether the location is on or off the tectonic plate the datum is tied to. Under these circumstances the cellMaxLevel may be realised as an attribute of a parent cell one or more refinement levels above the cell.</p>		M	1	Integer
nodes	Ordered sequence of vertices, clockwise round the cell's boundary.	true	M	1	Node
numEdges	Number of edges that make up the cell's boundary.	true	M	1	Integer
perimeter	Length of the cell's boundary.	true	M	1	Length
<b>Constraints:</b>	(none)				

**Table 75 – Elements of Equal-Area Cell::EA\_Zone**

**Name:** EA\_Zone

**Definition:** An EA\_Cell's location is an EA\_Zone.

**Stereotype:** Interface

**Inheritance from:** LocationS

**Abstract:** true

**Associations:** (none)

	Name	Definition	Derived	Obligation	Maximum occurrence	Data type
<b>Public attributes:</b>	centroid	<p>Cell's representative position calculated as the cell's geodesic centre of surface area, held by the cell's LocationS.</p> <p>An EA_Zone's centroid is:</p> <ul style="list-style-type: none"><li>a) The representative Position from LocationS, and</li><li>b) The controlPoint of ISO19111 Surface.</li></ul>		M	1	DirectPosition (data type)
	identifier	<p>EA_Zone's unique GeographicIdentifier. Commonly known as the CellID.</p> <p><b>NOTE</b> A zone's position along the path of the dggsAxis is represented by it's identifier, which is analogous to ISO 19107 coordinates. Zone identifiers are typically a single structured string or binary value constructed from a concatenated sequence of alpha-numeric or binary digits, corresponding to the sequence of tessellations which formed the associated cell. Each tessellation provides one digit representing the relative location of the child cell to its parent in the tessellation. Thus Cell IDs carry resolution, area, position, and topology information in their structure. CellIDs are derived from cell's ordered position along the path</p>		M	1	GeographicIdentifier (data type)

of the `dggsAxis`, which in turn is a function of the `refinementlevel` of the `EA_Cell`'s `EA_DiscreteGlobalGrid`.

**Constraints:** (none)

**Table 76 – Elements of Equal-Area Cell::EA\_BoundaryType**

**Name:** EA\_BoundaryType

**Definition:** CodeList for permitted curve types as constructors of an `EA_Cell` boundary. **NOTE** All `EA_BoundaryType` curves can be constructed as an intersection of a plane and an ellipsoid. So each of these curve types is a straight line on a designated plane.

**Stereotype:** CodeList

**Abstract:** true

**Associations:** (none)

	<i>Name</i>	<i>Definition</i>
<b>Values:</b>	geodesic	Shortest path between two points on the spherical or ellipsoidal DGGS's Earth reference model. Geodesic curves are defined using the <code>GeodesicData</code> constructor.
	smallCircle	Curves that are the intersection of a plane and an ellipsoid, where the plane is perpendicular to the axis of rotation of an ellipsoid are in small circles, see <code>greatCircle</code> . Lines of latitude are small circles.
	greatCircle	The great circle is a special case of small circle where the two focal points of the ellipse are also on the defining plane. On a non-spherical ellipsoid this is the only circle that is also a geodesic.
	smallEllipse	Curves that are the intersection of a plane and an ellipsoid, where the plane is parallel to the axis of rotation of the

	ellipsoid, are small ellipses, see great Ellipse.
greatEllipse	A greatEllipse is a special case of small Ellipse where the axis of rotation is on the defining plane. GreatEllipses are geodesics. Lines of longitude are great Ellipses.
projectedLine	A curve on the surface of an ellipsoid, whose equal-area projection on the projection's reference plane is a straight line.



# ANNEX A (NORMATIVE) CONFORMANCE CLASS ABSTRACT TEST SUITE

---



# ANNEX A (NORMATIVE) CONFORMANCE CLASS ABSTRACT TEST SUITE

---

This annex specifies Abstract test suites for the conformance categories in this document. The requirement tests associated with each conformance category are listed in Table A.1–Table A.3. Tests identifiers below are relative to <http://www.opengis.net/spec/dggs/2.0/>

## A.1. COMMON SPATIO-TEMPORAL CLASSES PACKAGE CONFORMANCE CATEGORIES

---

The requirement tests that apply to each conformance category in the Common Spatio-temporal Classes package are listed in Table A.1

**Table A.1 – Conformance classes for Common Spatio-temporal Classes package**

Conformance class	Requirement tests
Temporal geometry and topology	Annex A.1, Requirement test A.1
Temporal reference systems using period identifiers	Annex A.1, Requirement test A.4
Spatial zone geometry and topology	Annex A.1, Requirement test A.2
Spatial reference systems using zonal identifiers	Annex A.1, Requirement test A.3, & Annex A.1, Requirement test A.5
Spatio-temporal zone geometry and topology	Annex A.1, Requirement test A.1, & Annex A.1, Requirement test A.2
Spatio-temporal reference systems using zonal identifiers	Annex A.1, Requirement test

A.3, Annex A.1,  
Requirement test  
A.4, & Annex A.1,  
Requirement test A.5

### REQUIREMENT TEST A.1: COMMON SPATIO-TEMPORAL CLASSES – TEMPORAL – GEOMETRY

Abbreviation	conf/cc/temporal/geometry
Type	Basic
Requirement	Clause 7.2.1.2, Requirement 1: <a href="http://www.opengis.net/spec/DGGS/2.0/req/cc/temporal/geometry">http://www.opengis.net/spec/DGGS/2.0/req/cc/temporal/geometry</a>
Reference subclause	Clause 7.2.1
Test purpose	To verify the common classes for temporal geometry and topology conform to the data model in Figure 3 and Figure 4, and defining tables in Table 5–Table 16.
Test method	Inspect documentation of the DGGS specification.

### REQUIREMENT TEST A.2: COMMON SPATIO-TEMPORAL CLASSES – ZONE – GEOMETRY

Abbreviation	conf/cc/zone/geometry
Type	Basic
Requirement	Clause 7.2.2.2, Requirement 2: <a href="http://www.opengis.net/spec/DGGS/2.0/req/cc/zone/geometry">http://www.opengis.net/spec/DGGS/2.0/req/cc/zone/geometry</a>
Reference subclause	Clause 7.2.2
Test purpose	To verify the common classes for zonal geometry and topology conform to the data model in Figure 5 and defining tables in Table 17–Table 22.
Test method	Inspect documentation of the DGGS specification.

### REQUIREMENT TEST A.3: COMMON SPATIO-TEMPORAL CLASSES – SPATIAL – LOCATION

Abbreviation	conf/cc/spatial/location
Type	Basic



### REQUIREMENT TEST A.3: COMMON SPATIO-TEMPORAL CLASSES – SPATIAL – LOCATION

Requirement	Clause 7.3.1.2, Requirement 3: <a href="http://www.opengis.net/spec/DGGS/2.0/req/cc/spatial/location">http://www.opengis.net/spec/DGGS/2.0/req/cc/spatial/location</a>
Reference subclause	Clause 7.3.1
Test purpose	To verify the common classes for spatial location conform to the data model in Figure 6 and defining tables in Table 23.
Test method	Inspect documentation of the DGGS specification.

### REQUIREMENT TEST A.4: COMMON SPATIO-TEMPORAL CLASSES – TEMPORAL – REFERENCE SYSTEM USING PERIOD IDENTIFIERS

Abbreviation	conf/cc/temporal/rsupi
Type	Basic
Requirement	Clause 7.3.2.2, Requirement 4: <a href="http://www.opengis.net/spec/DGGS/2.0/req/cc/temporal/rsupi">http://www.opengis.net/spec/DGGS/2.0/req/cc/temporal/rsupi</a>
Reference subclause	Clause 7.3.2
Test purpose	To verify the common classes for reference systems using period identifiers conform to the data model in Figure 6 and defining tables in Table 24–Table 27.
Test method	Inspect documentation of the DGGS specification.

### REQUIREMENT TEST A.5: COMMON SPATIO-TEMPORAL CLASSES – ZONE – REFERENCE SYSTEM USING ZONAL IDENTIFIERS

Abbreviation	conf/cc/zone/rsuzi
Type	Basic
Requirement	Clause 7.3.3.2, Requirement 5: <a href="http://www.opengis.net/spec/DGGS/2.0/req/cc/zone/rsuzi">http://www.opengis.net/spec/DGGS/2.0/req/cc/zone/rsuzi</a>
Reference subclause	Clause 7.3.3
Test purpose	To verify the common classes for reference systems using zonal identifiers conform to the data model in Figure 8–Figure 12 and defining tables in Table 28–Table 39.
Test method	Inspect documentation of the DGGS specification.

## A.2. DGGS CORE PACKAGE CONFORMANCE CATEGORIES

The requirement tests that apply to each conformance category in the DGGS Core package are listed in Table A.2

Table A.2 – Conformance classes for DGGS Core package

Conformance class	Requirement tests
Spatial DGGS Core	Annex A.1, Requirement test A.2, Annex A.1, Requirement test A.3, Annex A.1, Requirement test A.5, and all Core requirements in Annex A.2.1, Requirement test A.6–Annex A.2.2, Requirement test A.19.
Spatio-temporal DGGS Core	Annex A.1, Requirement test A.1, Annex A.1, Requirement test A.2, Annex A.1, Requirement test A.3, Annex A.1, Requirement test A.4, Annex A.1, Requirement test A.5, and all Core requirement tests in Annex A.2.1, Requirement test A.6–Annex A.2.2, Requirement test A.19.

## A.2.1. Core – Reference System

### REQUIREMENT TEST A.6: CORE – REFERENCE SYSTEM – HARMONIZED MODEL

Abbreviation	conf/core/rs/harmonized_model
Type	Basic
Requirement	Clause 8.2.2, Requirement 6: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/rs/harmonized_model">http://www.opengis.net/spec/DGGS/2.0/req/core/rs/harmonized_model</a>
Reference subclause	Clause 8.2
Test purpose	To verify that all DGGS RS classes comply with classes in the DGGS Core RS data model in Figure 13 and definitions in Table 40–Table 46.
Test method	Inspect documentation of the DGGS specification.

### REQUIREMENT TEST A.7: CORE – REFERENCE SYSTEM – CRS

Abbreviation	conf/core/rs/crs
Type	Basic
Requirement	Clause 8.2.2, Requirement 7: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/rs/crs">http://www.opengis.net/spec/DGGS/2.0/req/core/rs/crs</a>
Reference subclause	Clause 8.2
Test purpose	To verify that the DGGS RS defines a CRS and complies with coordinate epoch requirements as specified for MD_Reference System.
Test method	Inspect documentation of the DGGS specification.

### REQUIREMENT TEST A.8: CORE – REFERENCE SYSTEM – GLOBAL DOMAIN

Abbreviation	conf/core/rs/global_domain
Type	Basic
Requirement	Clause 8.2.3, Requirement 8: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain">http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain</a>

## REQUIREMENT TEST A.8: CORE – REFERENCE SYSTEM – GLOBAL DOMAIN

Reference subclause	Clause 8.2.3
Test purpose	To verify a DGGS RS specifies a global domain, and its spatial, temporal and topological dimensionality.
Test method	Inspect documentation of the DGGS specification.

## REQUIREMENT TEST A.9: CORE – REFERENCE SYSTEM – GLOBAL DOMAIN – COMPLETE

Abbreviation	conf/core/rs/global_domain/complete
Type	Basic
Requirement	Clause 8.2.3, Requirement 9: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain/complete">http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain/complete</a>
Reference subclause	Clause 8.2.3
Test purpose	To verify reference system domain completeness, with the level zero discrete global grid covering the entire global domain.
Test method	Inspect documentation of the DGGS specification.

## REQUIREMENT TEST A.10: CORE – REFERENCE SYSTEM – GLOBAL DOMAIN – UNIQUE

Abbreviation	conf/core/rs/global_domain/unique
Type	Basic
Requirement	Clause 8.2.3, Requirement 10: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain/unique">http://www.opengis.net/spec/DGGS/2.0/req/core/rs/global_domain/unique</a>
Reference subclause	Clause 8.2.3
Test purpose	To verify reference system domain uniqueness, with every location in the domain of the DGGS located in exactly one cell of the level zero discrete global grid.
Test method	Inspect documentation of the DGGS specification.

## REQUIREMENT TEST A.11: CORE – REFERENCE SYSTEM – CELL – SIMPLE

Abbreviation	conf/core/rs/cell/simple
--------------	--------------------------

## REQUIREMENT TEST A.11: CORE – REFERENCE SYSTEM – CELL – SIMPLE

Type	Basic
Requirement	Clause 8.2.4.1, Requirement 11: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/cell/simple">http://www.opengis.net/spec/DGGS/2.0/req/core/cell/simple</a>
Reference subclause	Clause 8.2.4.1
Test purpose	To verify that all the cells of a DGGS specification have shapes that are simple geometry, where the geometry of the cell: does not self-intersect; is topologically the same as the equivalent shape of a circle with the cell's dimensionality; and encloses a region that is measurable in the cell's metric.
Test method	Inspect documentation of the DGGS specification.

## REQUIREMENT TEST A.12: CORE – REFERENCE SYSTEM – CELL – DIRECT POSITION

Abbreviation	conf/core/rs/cell/direct_position
Type	Basic
Requirement	Clause 8.2.4.2, Requirement 12: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/cell/direct_position">http://www.opengis.net/spec/DGGS/2.0/req/core/cell/direct_position</a>
Reference subclause	Clause 8.2.4.2
Test purpose	To verify that all zones in each discrete global grid are assigned a direct position that is within the zone's boundary.
Test method	Inspect documentation of the DGGS specification.

## REQUIREMENT TEST A.13: CORE – REFERENCE SYSTEM – CELL – ADDRESS

Abbreviation	conf/core/rs/cell/address
Type	Basic
Requirement	Clause 8.2.4.3, Requirement 13: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/cell/address">http://www.opengis.net/spec/DGGS/2.0/req/core/cell/address</a>
Reference subclause	Clause 8.2.4.3
Test purpose	To verify that all zones in all discrete global grids have a globally unique zonal identifier (or cell index) that provides a spatio-temporal reference.

### REQUIREMENT TEST A.13: CORE – REFERENCE SYSTEM – CELL – ADDRESS

Test method	Inspect documentation of the DGGs specification.
-------------	--

### REQUIREMENT TEST A.14: CORE – REFERENCE SYSTEM – DISCRETE GLOBAL GRID

Abbreviation	conf/core/rs/discrete_global_grid
--------------	-----------------------------------

Type	Basic
------	-------

Requirement	Clause 8.2.5, Requirement 14: <a href="http://www.opengis.net/spec/DGGs/2.0/req/core/rs/discrete_global_grid">http://www.opengis.net/spec/DGGs/2.0/req/core/rs/discrete_global_grid</a>
-------------	---

Reference subclause	Clause 8.2.5
---------------------	--------------

Test purpose	To verify that a DGGs RS defines discrete global grids as aggregations of all the cells at the same level in the hierarchy.
--------------	---

Test method	Inspect documentation of the DGGs specification.
-------------	--

### REQUIREMENT TEST A.15: CORE – REFERENCE SYSTEM – DISCRETE GLOBAL GRID – SEQUENCE

Abbreviation	conf/core/rs/discrete_global_grid/sequence
--------------	--

Type	Basic
------	-------

Requirement	Clause 8.2.5, Requirement 15: <a href="http://www.opengis.net/spec/DGGs/2.0/req/core/rs/discrete_global_grid/sequence">http://www.opengis.net/spec/DGGs/2.0/req/core/rs/discrete_global_grid/sequence</a>
-------------	---

Reference subclause	Clause 8.2.5
---------------------	--------------

Test purpose	To verify a DGGs RS orders its discrete global grids in a sequence of increasing refinement level.
--------------	--

Test method	Inspect documentation of the DGGs specification.
-------------	--

## A.2.2. Core – Functions

### REQUIREMENT TEST A.16: CORE – FUNCTIONS – QUANTIZATION

Abbreviation	conf/core/functions/quantization
--------------	----------------------------------

## REQUIREMENT TEST A.16: CORE – FUNCTIONS – QUANTIZATION

Type	Basic
Requirement	Clause 8.3.2, Requirement 16: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/functions/quantization">http://www.opengis.net/spec/DGGS/2.0/req/core/functions/quantization</a>
Reference subclause	Clause 8.3.1
Test purpose	To verify that a DGGS specification has operations to assign data from external sources to cells that conform to the data model in Figure 14 and definitions in Table 47–Table 51.
Test method	Inspect documentation of the DGGS specification.

## REQUIREMENT TEST A.17: CORE – FUNCTIONS – QUERY – ZONEQUERY

Abbreviation	conf/core/functions/query/zonequery
Type	Basic
Requirement	Clause 8.3.4, Requirement 17: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/functions/query/zonequery">http://www.opengis.net/spec/DGGS/2.0/req/core/functions/query/zonequery</a>
Reference subclause	Clause 8.3.3
Test purpose	To verify that a DGGS specification implements query operations that conform to the data model in Figure 16 and definitions in Table 52–Table 54, across its entire domain.
Test method	Inspect documentation of the DGGS specification.

## REQUIREMENT TEST A.18: CORE – FUNCTIONS – INTEROPERATION – QUERY

Abbreviation	conf/core/functions/interoperation/query
Type	Basic
Requirement	Clause 8.3.5.4, Requirement 18: <a href="http://www.opengis.net/spec/DGGS/2.0/req/core/functions/interoperation/query">http://www.opengis.net/spec/DGGS/2.0/req/core/functions/interoperation/query</a>
Reference subclause	Clause 8.3.5.3
Test purpose	To verify that a DGGS specification implements operations to read, interpret and execute external data queries that conform to the data model Figure 17 and Figure 18 and definitions in Table 55–Table 59.

## REQUIREMENT TEST A.18: CORE – FUNCTIONS – INTEROPERATION – QUERY

Test method	Inspect documentation of the DGGs specification.
-------------	--

## REQUIREMENT TEST A.19: CORE – FUNCTIONS – INTEROPERATION – BROADCAST

Abbreviation	conf/core/functions/interoperation/broadcast
--------------	--

Type	Basic
------	-------

Requirement	Clause 8.3.5.6, Requirement 19: <a href="http://www.opengis.net/spec/DGGs/2.0/req/core/functions/interoperation/broadcast">http://www.opengis.net/spec/DGGs/2.0/req/core/functions/interoperation/broadcast</a>
-------------	---

Reference subclause	Clause 8.3.5.5
---------------------	----------------

Test purpose	To verify that a DGGs specification implements operations to translate data results from interoperation queries to standard data formats and broadcast the reformatted result set that conform to the data model in Figure 17, Figure 19, Table 55, and definitions in Table 60...Table 62_.
--------------	--

Test method	Inspect documentation of the DGGs specification.
-------------	--

## A.3. EQUAL-AREA EARTH DGGs PACKAGE CONFORMANCE CATEGORIES

The requirement tests that apply to each conformance category in the Equal-Area Earth DGGs package are listed in Table A.3

Table A.3 – Categories of conformance for Equal-Area Earth DGGs package

Conformance class	Requirement tests
Equal-Area Earth DGGs	Annex A.1, Requirement test A.2, Annex A.1, Requirement test A.3, & Annex A.1, Requirement test A.5, and



all Core requirement tests in Annex A.2.1, Requirement test A.6–Annex A.2.2, Requirement test A.19, and all EAERS requirement tests in Annex A.3, Requirement test A.20–Annex A.3, Requirement test A.29.

### REQUIREMENT TEST A.20: EQUAL AREA – EARTH REFERENCE SYSTEM – HARMONIZED MODEL

Abbreviation	conf/ea/ers/harmonized_model
Type	Basic
Requirement	Clause 9.1.2, Requirement 20: <a href="http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/harmonized_model">http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/harmonized_model</a>
Reference subclause	Clause 9.1
Test purpose	To verify an EAERS specification complies with the data model in Figure 20, Figure 21 and Figure 22 and definitions in Table 63–Table 76
Test method	Inspect documentation for the EAERS specification.

### REQUIREMENT TEST A.21: EQUAL AREA – EARTH REFERENCE SYSTEM – GLOBAL DOMAIN

Abbreviation	conf/ea/ers/global_domain
Type	Basic
Requirement	Clause 9.1.3, Requirement 21: <a href="http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/global_domain">http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/global_domain</a>
Reference subclause	Clause 9.1.3
Test purpose	To verify the domain of an EAERS is the whole surface of the reference frame's earth model and the DGG_ReferenceSystem cellEqualSized is specified as a DGGS_Grids constraint.

## REQUIREMENT TEST A.21: EQUAL AREA – EARTH REFERENCE SYSTEM – GLOBAL DOMAIN

Test method	Inspect documentation of the EAERS specification.
-------------	---

## REQUIREMENT TEST A.22: EQUAL AREA – EARTH REFERENCE SYSTEM – TESSELLATION – INITIAL

Abbreviation	conf/ea/ers/tessellation/initial
--------------	----------------------------------

Type	Basic
------	-------

Requirement	Clause 9.1.4.2, Requirement 22 : <a href="http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/initial">http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/initial</a>
-------------	--

Reference subclause	Clause 9.1.4.2
---------------------	----------------

Test purpose	To verify that an EAERS specification has an initial tessellation comprising equal area cells produced by mapping the faces of a base unit polyhedron to a surface model of the Earth.
--------------	--

Test method	Inspect documentation of the EAERS specification.
-------------	---

## REQUIREMENT TEST A.23: EQUAL AREA – EARTH REFERENCE SYSTEM – TESSELLATION – SEQUENCE

Abbreviation	conf/ea/ers/tessellation/sequence
--------------	-----------------------------------

Type	Basic
------	-------

Requirement	Clause 9.1.4.3, Requirement 23 : <a href="http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/sequence">http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/sequence</a>
-------------	--

Reference subclause	Clause 9.1.4.3
---------------------	----------------

Test purpose	To verify an EAERS defines tessellation operations that generate a sequence of discrete global grids with progressively smaller cells.
--------------	--

Test method	Inspect documentation of the EAERS specification.
-------------	---

## REQUIREMENT TEST A.24: EQUAL AREA – EARTH REFERENCE SYSTEM – TESSELLATION – SEQUENCE – MAX

Abbreviation	conf/ea/ers/tessellation/sequence/max
--------------	---------------------------------------

## REQUIREMENT TEST A.24: EQUAL AREA – EARTH REFERENCE SYSTEM – TESSELLATION – SEQUENCE – MAX

Type	Basic
Requirement	Clause 9.1.4.3, Requirement 24 : <a href="http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/sequence/max">http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/sequence/max</a>
Reference subclause	Clause 9.1.4.3
Test purpose	To verify that an EAERS specifies a limit in its sequence of discrete global grids that ensures the error budget for EA_Cell area cannot be exceeded.
Test method	Inspect documentation of the EAERS specification.

## REQUIREMENT TEST A.25: EQUAL AREA – EARTH REFERENCE SYSTEM – TESSELLATION – GLOBAL AREA PRESERVATION

Abbreviation	conf/ea/ers/tessellation/global_area_preservation
Type	Basic
Requirement	Clause 9.1.4.4, Requirement 25 : <a href="http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/global_area_preservation">http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/tessellation/global_area_preservation</a>
Reference subclause	Clause 9.1.4.4
Test purpose	To verify for EA_Cells in each successive discrete global grid, that domain completeness and location uniqueness are preserved, with total area of those EA_Cells the same as the initial global grid and no overlapping DGGS cells.
Test method	Inspect documentation of the EAERS specification.

## REQUIREMENT TEST A.26: EQUAL AREA – EARTH REFERENCE SYSTEM – CELL – SIMPLE – 2D POLYGON

Abbreviation	conf/ea/ers/cell/simple/2d_polygon
Type	Basic
Requirement	Clause 9.1.5.1, Requirement 26: <a href="http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/simple/2d_polygon">http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/simple/2d_polygon</a>
Reference subclause	Clause 9.1.5.1
Test purpose	To verify that all of the cells of an EAERS specification have shapes that are simple polygons on the surface model of the

## REQUIREMENT TEST A.26: EQUAL AREA – EARTH REFERENCE SYSTEM – CELL – SIMPLE – 2D POLYGON

Earth where: edges only meet at vertices; exactly two edges meet at each vertex; the number of edges and vertices are the same; and the region enclosed always has a measurable area.

**Test method** Inspect documentation of the EAERS specification.

## REQUIREMENT TEST A.27: EQUAL AREA – EARTH REFERENCE SYSTEM – CELL – DIRECT POSITION – CENTROID

**Abbreviation** conf/ea/ers/cell/direct\_position/centroid

**Type** Basic

**Requirement** Clause 9.1.5.2, Requirement 27: [http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/direct\\_position/centroid](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/direct_position/centroid)

**Reference subclause** Clause 9.1.5.2

**Test purpose** To verify that the cells defined by an EAERS are referenced at the centroid of each cell. This test requires verification that the attribute EA\_Zone(centroid) returns a direct position on the surface of the cell.

**Test method** Inspect documentation of the EAERS specification.

## REQUIREMENT TEST A.28: EQUAL AREA – EARTH REFERENCE SYSTEM – CELL – EQUAL AREA – ERROR BUDGET

**Abbreviation** conf/ea/ers/cell/equal\_area/error\_budget

**Type** Basic

**Requirement** Clause 9.1.5.3, Requirement 28: [http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/equal\\_area/error\\_budget](http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/equal_area/error_budget)

**Reference subclause** Clause 9.1.5.3

**Test purpose** To verify that an EAERS specification defines error budget(s) of 1% or less that represents the maximum ratio of a cell's area to the theoretical cell area computed from the number of cells and the surface area of the Earth model.

**Test method** Inspect documentation of the EAERS specification.

## REQUIREMENT TEST A.29: EQUAL AREA – EARTH REFERENCE SYSTEM – CELL – EQUAL AREA

<b>Abbreviation</b>	conf/ea/ers/cell/equal_area
<b>Type</b>	Basic
<b>Requirement</b>	Clause 9.1.5.3, Requirement 29: <a href="http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/equal_area">http://www.opengis.net/spec/DGGS/2.0/req/ea/ers/cell/equal_area</a>
<b>Reference subclause</b>	Clause 9.1.5.3
<b>Test purpose</b>	To verify that the cells in each discrete global grid, and of each cell geometry in a DGGS specification have equal area within the specified error budget.
<b>Test method</b>	Inspect documentation of the EAERS specification, and for each discrete global grid, compare the area of each cell defined by its geometry with the average area of all cells of the same geometry (triangle, quadrangle, pentagon, or hexagon).



B

# ANNEX B (INFORMATIVE) EQUAL AREA DGGS THEORY

---

## B

## ANNEX B (INFORMATIVE) EQUAL AREA DGGS THEORY

---

All Discrete Global Grid Systems (DGGS) are structured for information as distinct from conventional coordinate reference systems originally designed for navigation. For a grid based global spatial information system to operate effectively as an analytical system it should be constructed using cells that represent the surface of the Earth in a uniform way. This ensures that, at each resolution, all cells have the same probability of contributing to an analysis. An Equal Area DGGS is a spatial reference system that uses a hierarchy of equal area tessellations to partition the surface of the Earth into grid cells or their analogous lattice points. In this way, information recorded about phenomena at a location can be easily referenced to the explicit area of the associated cell, integrated with other cell values, and provides statistically valid summaries based on any chosen selection of cells. With equal area partitioning, spatial analysis can be replicated consistently anywhere on the Earth independent of resolution or scale.

Equal Area Earth DGGS reference systems are polyhedral reference systems on the surface of a base unit polyhedron's circumscribed ellipsoid. The base unit polyhedron's location and orientation is defined in Earth Centered (EC) coordinates. The initial equal area tessellation of the chosen ellipsoidal Earth model is constructed by scaling a unit polyhedron of defined orientation until its vertices all touch the ellipsoid and connecting adjoining vertices with arcs selected from the set of permitted arcs, the simplest of which are geodesic, small circle or small ellipse arcs. Appropriate differential scaling is applied to the unit polyhedron to ensure an equal area initial tessellation. For the simple case of regular polyhedra and geodesic (i.e. great circle) arcs on its circumscribed spheroid, the scaling is uniform. Figure B.1 illustrates their simplest form using a regular spherical polyhedron with a spheroidal circumscribing ellipsoid and geodesic arcs. Small circle arcs are typically used to construct arcs along lines of latitude for both ellipsoids and spheroids. Both small circle and small ellipse arcs are formed from the intersection of a defined plane with the ellipsoid, and in that sense they can be considered equivalent to the "straight" lines of two-dimensional cell boundaries. More complex forms of straight line, such as arcs that project to a straight line in an equal area projection are also allowed.

Figure B.1 – Regular polyhedra (top) and their corresponding initial equal area tessellation (bottom)

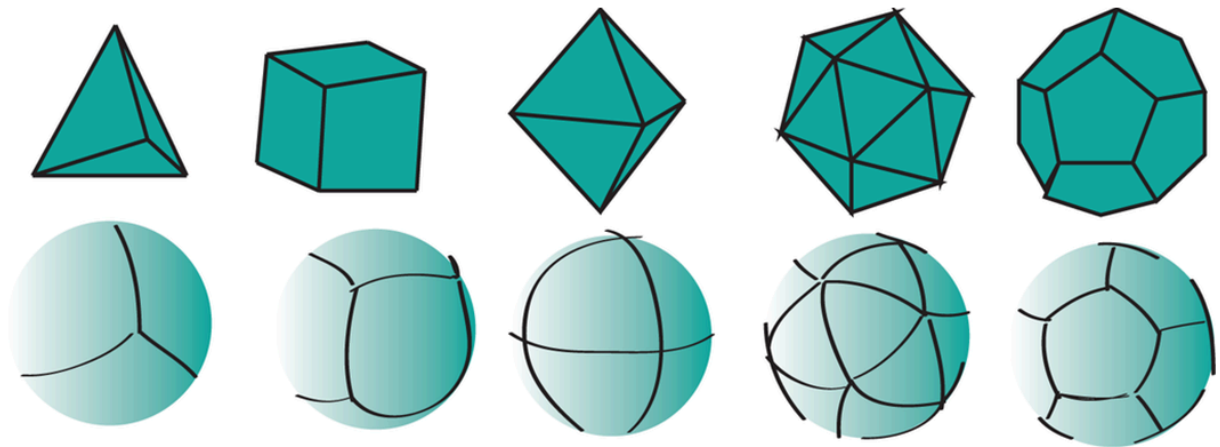


Table B.1

a) tetrahedron    b) cube    c) octahedron    d) icosahedron    e) dodecahedron

NOTE: Based on Figure 2 in [1]

There is a gap between conventional coordinate reference systems and the reference system needed to define DGGs. This document fills the gap in existing ISO standard reference systems and establishes requirements for globally interoperable equal-area cell- or lattice-based information frameworks.

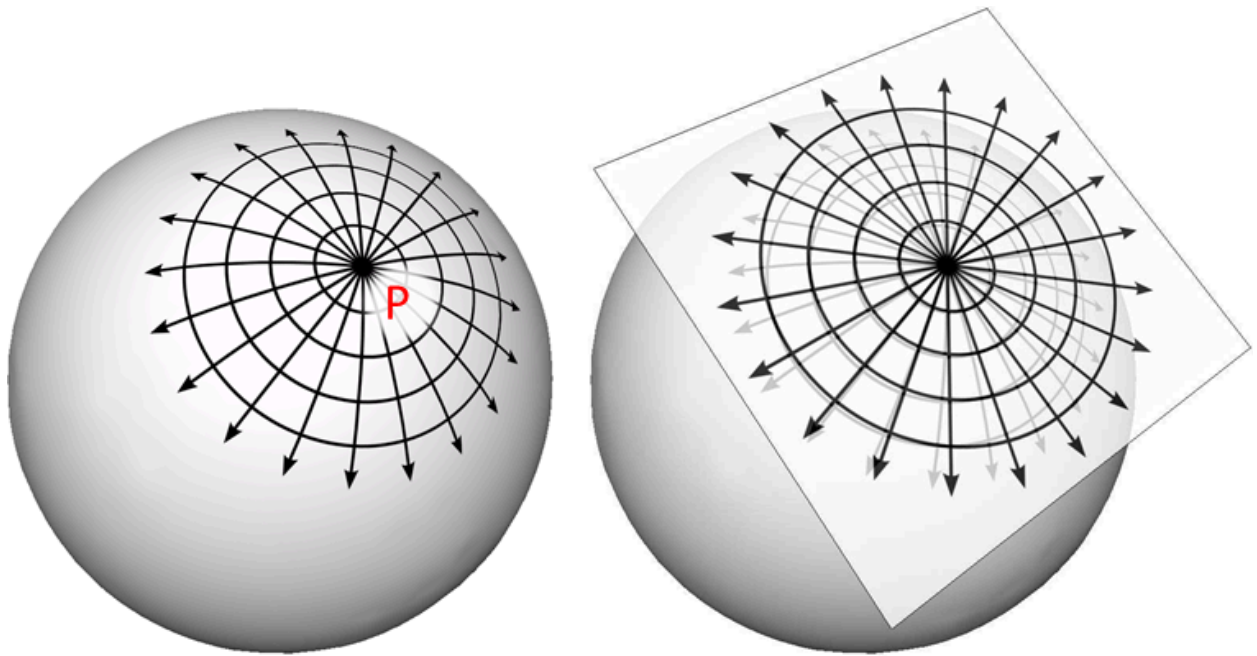
Existing spatial CRS (e.g. ECEF [Earth Centered Earth Fixed], WGS 84 or Web Mercator) build grids from projected Cartesian or ellipsoidal coordinate axes. Rectangular planar grids are typically formed by establishing a set of regular ticks on a pair of linear axes with grids cells being formed by the intersection of straight lines drawn normal to the ticks on each axis. Analogous construction techniques can be used to create triangular or hexagonal grids. The properties of grids built this way arise from the premise of planar geometry and not the curved geometry of the surface of a sphere or ellipsoid. While these properties hold true at local scales, in curved geometries they increasingly fail at progressively larger regions of interest (see Figure B.2). Take for example the assumption that a grid cell's geometric properties are independent of its size or resolution, which is implicit in constructing sets of planar aligned (or 'nested') 10 m, 30 m and 90 m grids. As shown in Figure B.3, a 90 m square cell formed from nine 30 m square child cells has the following properties:

1. It is also square;
2. Its edges are three times the edge length of its 30 m child cells, which in turn all are three times the edge length of their 10 m child cells;
3. Its interior angles are all right angles and identical to the interior angles of all of the child cells;



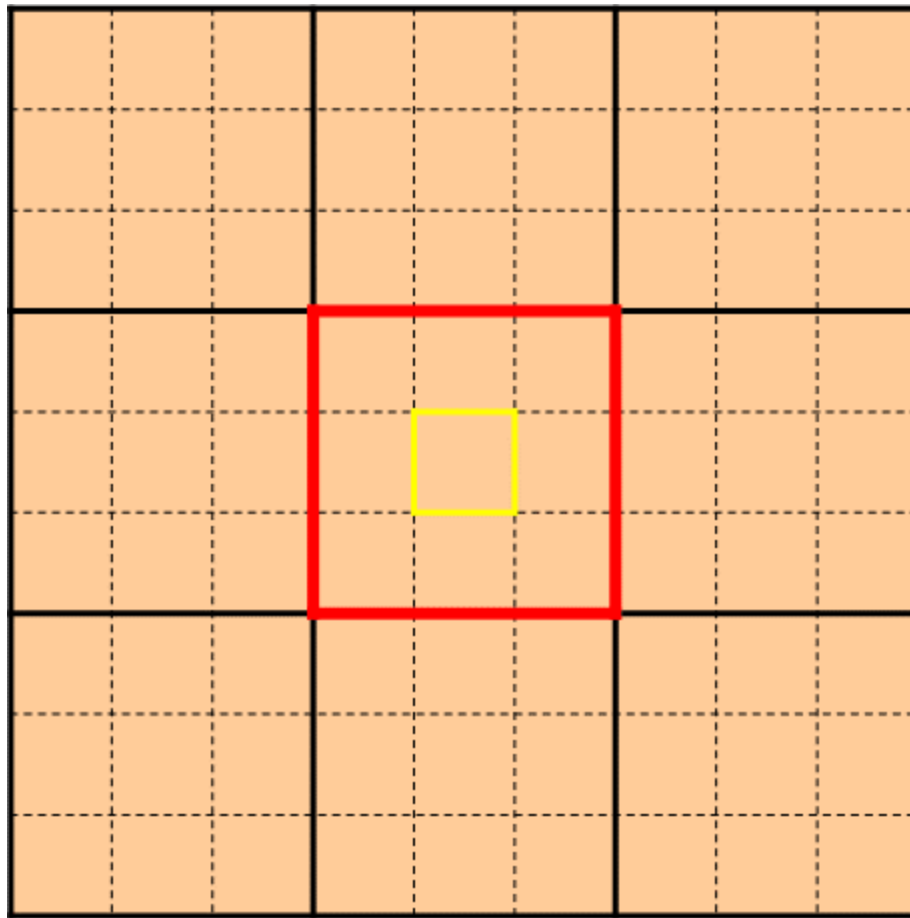
4. Its edges follow the shortest linear path between neighbouring cell vertices; and,
5. The angles or bearings from centroid to centroid between cells are preserved irrespective of the direction of travel.

**Figure B.2 – Comparison of a grid (in this case radial) represented on both (a) curved and (b) planar surfaces.**



Note With increasing distance away from the point P there is an increasing deviation between the two representations of the grid [Based on Figure 15 in [2]; and Figure 3 in [44]].

Figure B.3 – Planar square grid with nested child cells.



Note            The red and yellow cells have identical geometry, and in each case the geometry is also shared with all other cells of the same size.

On a curved surface, however, this is never the case. Yet we often make the same assumption; that all cells are geometrically identical in, for example, a country- or continental-wide mosaic comprising many satellite images. Consequently, under this paradigm assumption, choosing a fixed cell size for a global grid whose cells represent equal areas and seamlessly fit the earth's surface is problematic. When this is required, conventional spatial standards enforce latitude-longitude axes to be used and these grids are therefore described in these spherical coordinates. But the cells of these types of (equal-angular) grids do not have the same properties of planar grids. Figure B.4 shows a similar consideration to that of Figure B.3, only the grids are constructed using spherical instead of planar Cartesian coordinates. In this scenario, the largest (parent) cell does not necessarily have the same shape or internal angles as the child cells. Also, its edges do not follow the shortest linear path from corner to corner. Bearing directions between cell centroids, however, are preserved in both planar and curved geometry spaces.

Figure B.4 – Geometries and spatial properties of cells on curved surfaces.

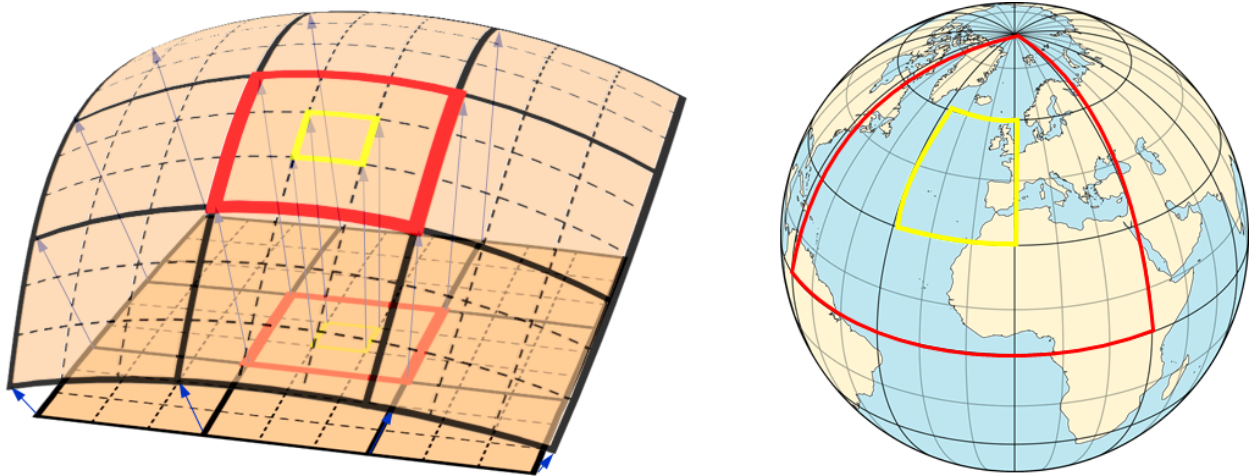


Table B.2

a) Square grid with nested child cells on a portion of a sphere (projected from the planar grid shown in Figure B.3)

b) Latitude-longitude (equal angular) grid, the red cell is  $90^\circ \times 90^\circ$  and has nine  $30^\circ \times 30^\circ$  child cells (the central child cell is shown in yellow)

In an attempt to address this dichotomy, conventional spatial standards therefore support either small, local, well-behaved planar grids or global grids that preserve bearings and angular lengths, and do not preserve area, but not both at the same time. This document fills this gap by providing a formal specification for area preserving reference systems based on the surface model of the Earth that respect the accuracy and precision of spatial data at all scales from local to global. These systems use a hierarchical tessellation of the entire Earth to produce equal-area grids. Figure B.5 shows two examples. We anticipate that future extensions of the specifications in this document will support higher dimensions, such as the volume of the Earth and its atmosphere, and the Earth through time.

Figure B.5 – Tessellations of the Earth to equal-area cells.

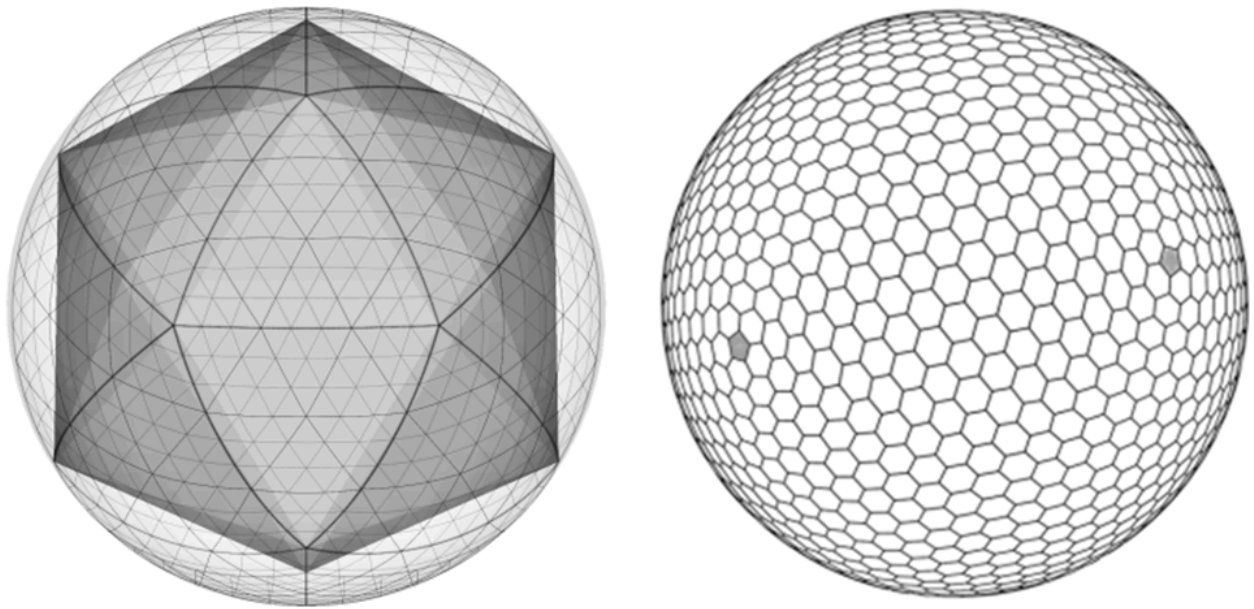


Table B.3

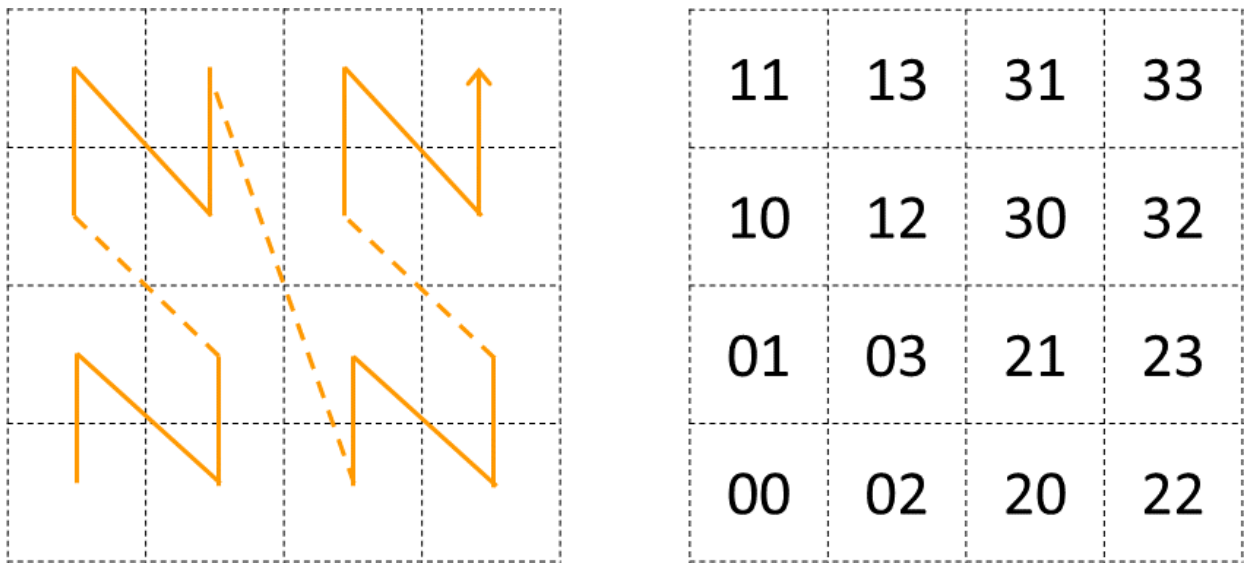
- |   |   |
|---|---|
| a) Triangular cells starting with an icosahedron. Then refining each cell recursively into 9 smaller cells. Shown after two iterations. | b) Hexagonal cells with twelve pentagonal cells at the vertices of the initial tessellation |
|---|---|

The language and foundations of current geospatial standards are deeply rooted in planar thinking, so while this document leverages as much as it can from existing standards, it also introduces new concepts that are subtly and yet fundamentally, different from those described by the standards that it draws from. These subtle differences challenge our thinking. As a consequence, this document is an evolution of both existing raster processing practice and past usage of discrete global grids.

As a specification for an area preserving earth reference system, this document defines more than just grids and lattices. The underlying geometry of the cells and the topological relationships between neighbouring cells can be used to define GUIDs for the cells across all resolution.

Earlier, we noted that planar grids are formed from the pairs of axes each with regular ticks corresponding to the cell dimension, facilitating a simple topological referencing schema for each cell (usually via a matrix style index for each cell along the axes of the grid, i.e. rows and columns for a two-dimensional grid). With DGGs, we introduce a more sophisticated set of cell referencing schemas, such as space filling curves which traverse all of the cells in a manner that is functionally equivalent to the axes. As shown in Figure B.6, cell indices are assigned to cells along the path of the space filling curve. These indices, together with the geometry of the space filling curve, carry the metrics of the curved surface and the topological relationships between neighbouring cells. The cell indices are explicitly treated as GUIDs.

Figure B.6 – Using Morton space filling curve for defining.



Note Labels of 4×4 square cells. (after [Based on Figure 25 in[2]])

The mathematical properties of integers and real numbers on axis pairs in a plane are known implicitly and are therefore not part of any ISO document concerning planar grids. The theoretical basis on which the separate disciplines for space filling curves, GUIDs, grids, spatial topology and discrete global grids are also well founded. However, their roles in DGGs are not implicitly understood. This document therefore defines these roles and relationships explicitly. This is a necessary departure from previous DGGs work that is needed to ensure a robust spatial reference frame standard. A brief history of DGGs is provided in Annex C for reference.



# ANNEX C (INFORMATIVE) BACKGROUND TO DGGS

---

C

## ANNEX C (INFORMATIVE) BACKGROUND TO DGGs

---

### C.1. DGGs AS A DIGITAL INFORMATION MEDIUM

---

Conventional coordinate reference systems address the globe with a continuous field of points suitable for repeatable navigation and analytical geometry. While this continuous field is represented on a computer in a digitized and discrete fashion by tuples of fixed-precision floating point values, it is a non-trivial exercise to relate point observations spatially referenced in this way to areal features on the surface of the Earth. In contrast, a DGGs is designed to be an information grid, not a navigation grid. DGGs provide a fixed areal based geospatial reference frame for the persistent location of measured Earth observations, feature interpretations, and modelled predictions. DGGs address the entire planet by partitioning it into a discrete hierarchical tessellation of progressively finer resolution cells. The geometry and location of the cell is the principle aspect of a DGGs. Data integration, decomposition, and aggregation is optimized in the DGGs's hierarchical structure and can be exploited for efficient multi-source data processing, storage, discovery, transmission, visualization, computation, analysis, and modeling.

### C.2. HISTORY

---

The concept of using polyhedra to model the surface of the Earth is by no means new. In 1509 Lucia Pacioli published “De Divina Proportione” [11] a treatise, illustrated by Leonardo Da Vinci, exploring the mathematical characteristics of the “golden ratio” (also referred to as the “divine proportion”) which included a consideration of the properties of the five platonic solids circumscribed within the sphere and the eminent role of the “golden ratio” in the construction of two of them (the icosahedron and dodecahedron). It is likely that his time studying with Pacioli influenced Da Vinci's later thinking regarding spherical geometry; evidenced in 1515 by his derivation of the octahedral analysis of the volume of a sphere with various forms of segmentation [12]. This work is suggested to have led to the development of the Reuleaux triangular formulation of the world map (or *mappamundi*), attributed to Da Vinci [13], and can be



considered a precursor to differential calculus, formally developed by Newton and Leibniz nearly two centuries later.

In the 1940's, a similar approach was used by R. Buckminster Fuller in the development of the Dymaxion map of the world [14], a physical model of the Earth mapped onto the planar faces of a polyhedron (first presented as a cuboctahedron [15] and then later as an icosahedron [16]). The aim of the Dymaxion map was to depict the spherical world as a flat surface with true scale, true direction and correct configuration, all at the same time. Although a physical model of the Earth, and not strictly a DGGS, it inspired later researchers to produce digital Earth models, which in turn has led to the development of DGGS.

Formal development of DGGS began in the 1950s with the promising value of global analysis coinciding with the increased use of geographic information systems and the availability of global mapping data and positioning systems. Perhaps the first published instance of formalized discrete global grids with application to numerical analysis was described by Vestine et. al. [17] in 1955, where they define and use an equal-area grid based on the mapping of a spherical icosahedron onto the surface of the Earth as a framework to conduct areal based integral and spherical-harmonic analyses of the geomagnetic field. This grid was later used (and re-described) by Sadourny et. al. [18] in 1968 to model equations of atmospheric motion without distortion across the entire globe. Another style of discrete global grid, and perhaps the first application of hierarchical indexing schemas as an analytic cell referencing tool was implemented by Geoffrey Dutton in 1984 [19],[20],[21] at the Laboratory for Computer Graphics and Spatial Analysis at Harvard Graduate School of Design. Dutton's first global grid was designed for assembling and managing global terrain data on a triangular global grid. His global GEM [19],[20],[21],[22] started with a cuboctahedron connected into a rhombic dodecahedron (which is its dual polyhedron where the vertices of one corresponds to the centre of the faces of the other) and recursively divided the initial 12 triangular faces into refinements of 9 partially nested equilateral triangles. Dutton refined GEM to use only an octahedral basis in the QTM DGGS [19],[20],[21],[23]. QTM is a fourfold hierarchical decomposition of facets of an octahedron into triangles whose edges follow small circles. Elevations were assigned to the centroids of child triangles and to the vertices that coincided with them. Waldo Tobler and Zi-tan Chen [24] imagined the primary purpose for a formal discrete global grid standard would be information exchange and storage. Tobler argued that as a generalized information medium "*coverage must be uniform and that every element of area must have an equal probability of entering the system. This suggests that the world should be partitioned into chunks of equal size*" [24]. Tobler's global grid started with a cube as a base unit polyhedron and divided into rectilinear quad-trees to create successive subdivisions with unlimited resolution. Dennis White, Scott Overton, and Jon Kimerling, driven by a need for a statistically valid sampling to integrate aquatic and terrestrial monitoring for the US-EPA, designed a global grid in 1989 using closely packed hexagonal cells that started with a truncated icosahedron as the base unit polyhedron [25].

## C.3. GLOBAL GRID TAXONOMY

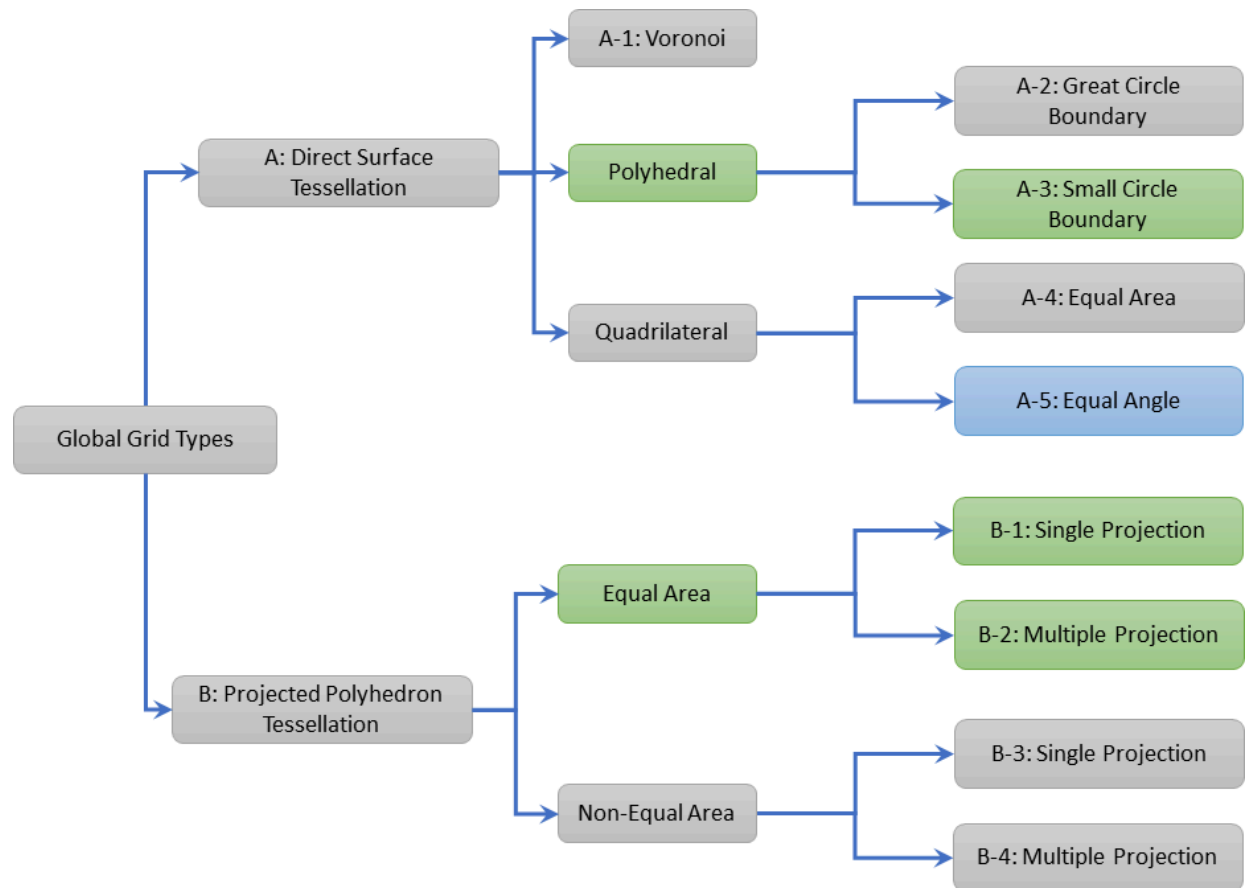
---

There have been numerous methods proposed for achieving a tessellation of the Earth, each with different distortion characteristics [26]. These tessellations can be organized into a limited set of categories that describe a hierarchical taxonomy of global grids Figure C.1, [26],[27]. Only two groups of classes of global grid achieve an equal area solution [26]: those based on equal



area mapping of a base unit polyhedron (such as the ISEA projection [28]) and those based on direct surface tessellations using small circle subdivision (e.g. References [29],[30]).

Figure C.1 – Global Grid Taxonomy (Based on Figure 4 in [2]).



Key

Table C.1

A	DIRECT SURFACE TESSELLATIONS	B	PROJECTED POLYHEDRON TESSELLATIONS
A-1	voronoi – Hipparchus geo-positioning [31]	B-1*	equal area – single projection [24],[36]
A-2	polyhedral – great circle boundary [32]	B-2*	equal area – multiple projection [28],[38],[39]
A-3*	polyhedral – small circle boundary [29],[30],[21]	B-3	non-equal area – single projection [37]
A-4	quadrilateral – "constant" area [33]	B-4	non-equal area – multiple projection [40],[41]
A-5†	quadrilateral – equal angle [34],[35],[45],[46]		

NOTE: Tessellations marked in the key as suitable for use with this document are:

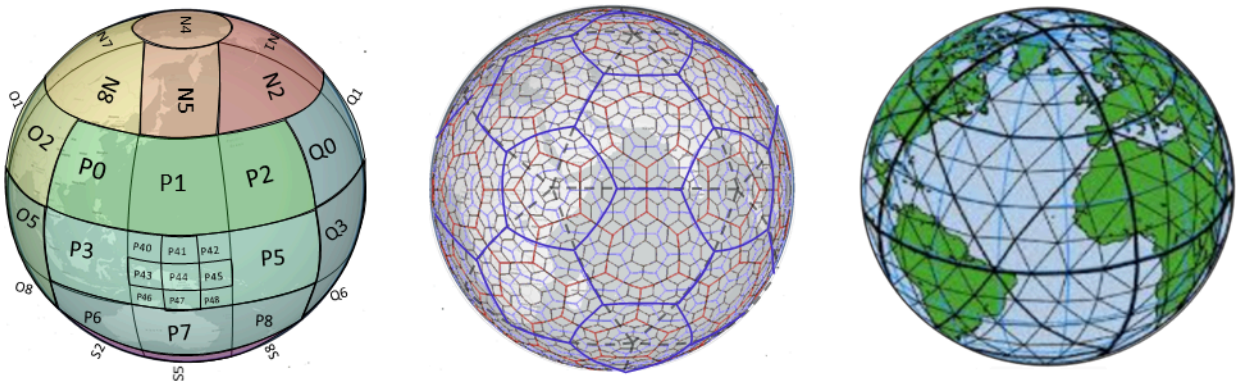
\*: Equal-area DGGs (Clause 9.1 in this document), and  
†: Axis-aligned DGGs (planned additional part to series).

## C.4. CRITERION

There are many possible DGGs, each with their own advantages and disadvantages. Criteria for a discrete global grid are well developed by both Michael F. Goodchild [7] and Jon Kimerling [26], the foremost requirements being a tessellation of cells that exhaustively cover the globe with each cell having an equal area, and a centroid representing a single point. The points and cells of the various resolution grids which constitute the grid system form a hierarchy which displays a high degree of regularity [42]. Choices for an appropriate tessellation include properties of shape, adjacency, connectivity, orientation, self-similarity, decomposability, and packing properties. Cell choices are generally taken from the three shapes that uniformly tile a plane – rectilinear, triangular, and hexagonal cells.

While it is possible to map any polyhedral solid to the surface of the earth, the Platonic solids (tetrahedron, cube, octahedron, dodecahedron and icosahedron; see Figure B.1) are the only polyhedral solids that perfectly partition the surface of a sphere into regular, equal area cells [43]. As a result, the Platonic solids are used in the construction of most equal area DGGs, often via a mapping of the polyhedral faces to the surface model of the Earth (some examples are shown in Figure C.2). This method of mapping the faces of a base unit polyhedron to the surface model of the Earth creates a coordinate reference system that is based on a curved geometric framework. GIS and image analysis packages that assume flat earth geometries need to adapt to support this new construct, which more closely represents the Earth.

**Figure C.2 – Examples of DGGs based on the mapping of the faces of Platonic solids to the surface model of the Earth**



Note

Table C.2

a) Quadrilateral cells on rHEALPIX projected hexahedron, levels 0–2 with refinement ratio of 9.	b) Hexagonal cells on ISEA projected icosahedron, levels 0–4 with refinement ratio of 3.	c) Triangular cells on QTM of an octahedron, levels 0–3 with refinement ratio of 4 (courtesy of Geoffrey Dutton).
---	--	---

Any tessellation of the Earth does not necessarily produce a DGGs. Single resolution computational grids are not sufficient to constitute a DGGs. Spatial data structures used to organize map tiles or optimize rapid spatial search cannot be considered to qualify as a DGGs in and of themselves. Although DGGs often utilize hierarchical indices to identify a cell, the primary feature of the DGGs is the cell geometry, not the optimization of a spatial query. Further, DGGs have data independent geometry – their geometry is not formed to optimize a balanced search like R-Trees or maximal spacing of data as generated by Voronoi diagrams.

## C.5. A DIGITAL SPATIAL REFERENCE SYSTEM

---

One way to understand the important difference between a DGGs and a conventional spatial reference system is to consider that a DGGs provides a digital framework for geospatial information. Geospatial information is essentially a signal, that is a variable (e.g. measurement of phenomena), which changes subject to another independent variable (e.g. spatial location, time, some physical interaction etc.). Conventional geospatial data are analogue signals, as they reference to a continuous space: geographic coordinates on an ellipsoidal datum. Even the discrete pixels of a satellite Earth observation image reference this continuous analogue model of Earth. However, these pixels do not observe precisely the same locational area for successive observations. Spatial reference by geographic identifier is described in ISO 19112. DGGs provide this globally in a structured form which is analogous to the ellipsoidal coordinate system described in ISO 19111, but is based on discrete cells rather than continuous point locations (e.g. Reference [46]).

Sampling and quantization are necessary for a signal to be considered digital. As the name implies, the DGGs provides the regular discrete intervals or cell partitioning to which location information (e.g. signal values) are sampled. A well-designed quantization strategy is also an important component of a DGGs which should maintain the fidelity of the original information in the values assigned to each cell. The discrete data values can be sampled from any geospatial data source, regardless of the original spatial reference, scale, format, type, frequency or time. A DGGs is a discrete “digital” model of the Earth.

## C.6. APPLICATION

---

As each cell in a DGGs is fixed in location, and the location provides an explicit area representation, basic geospatial enquiries, such as “Where is it?”, “What is here?”, and “How has it changed?”, are simplified into set theory operations. As any data values referenced to

a particular DGGs are, by the nature of the grid, aligned, the high costs of integrating data in traditional systems are dramatically reduced.

A DGGs can even be designed for lossless encoding of vector geometry such that cells, and their integer addressing, predictably converge to the real number coordinate pairs of each observation with each successive refinement: an essential property of a conventional coordinate system.

DGGs are designed to eliminate requirements for complex data fusion processes. Reducing the reliance on an intermediary integrator or analyst is a key requirement for distributed participatory digital-Earth information systems. “[Digital-Earth] *“can clearly benefit from developments in discrete global grid, which can provide the georeferencing, the indexing, and the discretization needed for geospatial data sets. They have properties, in particular hierarchical structure, uniqueness, explicit representation of spatial resolution, and consistency, that make them superior to any single alternative. [7].*

A DGGs provides a uniform environment to integrate, aggregate and visualize both vector/ point cloud geometries and raster-based geospatial data sources in much the same way that information within a computer graphics pipeline becomes the pixels on a computer screen. Efficiencies are gained through implementing the Dimensionally Extended nine-Intersection Model (DE-9IM) set of fundamental spatial operations [3]–[5],[6] directly on the DGGs cell structure. This allows for higher order analytics (via bindings to external analytic libraries) to be created on the DGGs structure itself, independent of the data sources.



D

# ANNEX D (INFORMATIVE) TEMPORAL GEOMETRY, TOPOLOGY, AND TEMPORAL REFERENCING BY NAMED PERIODS – CONTEXT FOR MODELLING

---

# D

## ANNEX D (INFORMATIVE) TEMPORAL GEOMETRY, TOPOLOGY, AND TEMPORAL REFERENCING BY NAMED PERIODS – CONTEXT FOR MODELLING

---

### D.1. GENERAL

---

This document builds on the spatio-temporal definitions in ISO 19111:2019 to achieve a broader integration of spatio-temporal concepts. In addressing alignment with ISO 19108:2002, the reader is directed to the discussion in ISO 19111:2019, Annex D as a starting point which is adopted in full in this document.

### D.2. GEOMETRY AND TOPOLOGY

---

Time can be represented as an ordered, one dimensional geometry with topology. The only significant difference between one-dimensional spatial and one-dimensional temporal is the importance of direction, and the topological relationships that explicitly address direction. ISO 19107:2019 and ISO 19108:2002 define terms for geometric and topological primitives and their complexes that are functionally equivalent but different in structure. To achieve spatio-temporal uniformity this document aligns its definitions with ISO 19107:2019. When compared with equivalent terms in ISO 19108:2002 this primarily results in differences in class inheritance rather than in underlying meaning. Comparing ISO 19108:2002 terms to this document's terms, while putting the structural differences aside:

- TM\_GeometricPrimitive and TM\_TopologicalPrimitive map to this document's TemporalGeometricPrimitive, TemporalTopologicalPrimitive respectively;
- TM\_TopologicalComplex maps to this document's TemporalTopologicalComplex;
- this document introduces TemporalGeometricComplex and TemporalGeometricCollection;

- the super classes of TM\_Primitive, TM\_Complex and TM\_Object are not implemented and instead this document introduces TemporalGeometry and TemporalTopology;
- TM\_Instant and TM\_Period map to this document's Instant and Interval respectively;
- TM\_Node and TM\_Edge map to this document's NodeT and EdgeT respectively;
- TM\_Duration, and its subclasses TM\_PeriodLength and TM\_IntervalLength map to this document's Duration, which encompasses any valid unit on a TemporalCS (ISO 19111:2019); and
- TM\_RelativePosition maps to this document's RelativePosition, which is defined to reflect modern usage by SDWIG and OGC as defined in OWL-time.

## D.3. REFERENCING BY NAMED PERIODS

---

ISO 19108:2002 includes the concepts of a named Ordinal Era, and Ordinal reference systems which address the needs of geological time. While the topology of Ordinal Era are known, their extent is in general ill-defined. To achieve spatio-temporal uniformity this document aligns itself with ISO 19112:2019. Named Periods are introduced as the temporal equivalent of named Locations (ISO 19112:2019). Periods have well-defined extents as well as well-defined topology. Comparing the terms of ISO 19108:2002 with this document's terms, while acknowledging this difference:

- TM\_OrdinalEra maps to both Period and PeriodClass in this document depending on context; and
- TM\_OrdinalReferenceSystems maps to TemporalReferenceSystemUsingPeriodIdentifiers in this document.



# ANNEX E (INFORMATIVE) REQUIREMENTS CROSSWALK FROM TOPIC 21 V1.0 TO V2.0



## E

# ANNEX E

## (INFORMATIVE)

### REQUIREMENTS CROSSWALK FROM TOPIC 21 V1.0 TO V2.0

---

Topic 21 v1.0 has only one conformance class – Equal Area Earth DGGs, Topic 21 v2.0 extends the 2D surface only data model to a potentially 3D+T data model by establishing Common Spatio-temporal Classes, and a dimension agnostic DGGs Core Reference system and dimensionally agnostic Functions for Quantisation, Query, and Interoperability. The conformance for Equal Area Earth DGGs is addressed in terms of conformance to the DGGs Core Functions for the 2D Earth surface and conformance to the Equal Area Earth Reference systems class built on the DGGs Core Reference system.

Requirements in Topic 21 v1.0 are addressed in a number of different ways in Topic 21 v2.0:

- **1:1** with a DGGs Core Requirement,
- **1:1** with an Equal Area Earth DGGs Reference system Requirement,
- **many:1** where multiple requirements have been consolidated into a single all inclusive requirement,
- **1:many** for example where conformance to the data model was addressed as a single requirement in v1.0 but this is now specifically incorporated into other requirements, and
- **0:1** with a new requirement, either in the Core or in the Equal Area Earth DGGs Reference system.

Even if the correspondence is 1:1, a new number has been assigned in v2.0 since the underlying data model is substantially different, or the definitions of terms used in the requirement have been rephrased, or the scope of the test has been broadened, for instance to include relevant parts of the data model. Requirements introduced in v2.0 appear with their number in brackets in the following list, grouped with other related requirements, and requirements that are only a partial match have their requirement number in italics.

Table E.1 – Requirements crosswalk from Topic 21 v1.0 to v2.0

15-104r5	Topic 21 v1.0	20-040	Topic 21 v2.0
#	Requirement	#	Requirement

		38	Equal Area – Earth Reference System – Harmonized Model
		34	Core – Functions – Quantization
1	Core Data Model	35	Core – Functions – Query – Zone Query
		36	Core – Functions – Interoperation – Query
		37	Core – Functions – Interoperation – Broadcast
		39	Equal Area – Earth Reference System – Global Domain
2	Reference Frame – Global Domain – Surface Area Equivalence	26	Core – Reference system – Global Domain
		27	Core – Reference system – Global Domain – Complete
3	Reference Frame – Global Domain – Cell Boundary Overlap	28	Core – Reference system – Global Domain – Unique
		41	Equal Area – Earth Reference System – Tessellation – Sequence
4	Reference Frame – Tessellation Sequence	42	Equal Area – Earth Reference System – Tessellation – Sequence – Max
		(33)	Core – Reference system – Discrete Global Grid – Sequence
5	Reference Frame – Global Area Preservation	43	Equal Area – Earth Reference System – Tessellation – Global Area Preservation
6	Reference Frame – Cell Shape	44	Equal Area – Earth Reference System – Cell – Simple – 2D Polygon
7	Reference Frame – Equal Area Precision	46	Equal Area – Earth Reference System – Cell – Equal Area – Error Budget
8	Reference Frame – Equal Area Cells	47	Equal Area – Earth Reference System – Cell – Equal Area
9	Reference Frame – Initial Tessellation	40	Equal Area – Earth Reference System – Tessellation – Initial
10	Reference Frame – Cell Refinement	(33)	Core – Reference system – Discrete Global Grid – Sequence
11	Reference Frame – Cell Addressing		
		31	Core – Reference system – Cell – Address
12	Reference Frame – Spatial Reference		

13	Reference Frame – Spatial Reference – Cells Referenced at their Centroid	45	Equal Area – Earth Reference System – Cell – Direct Position – Centroid
14	Functional Algorithms – Quantization Operations	34	Core – Functions – Quantization
15	Functional Algorithms – Algebraic Processes – Cell Navigation	35	Core – Functions – Query – Zone Query
16	Functional Algorithms – Algebraic Processes – Spatial Analysis		
17	Functional Algorithms – Interoperability Query Operations	36	Core – Functions – Interoperation – Query
18	Functional Algorithms – Interoperability Broadcast Operations	37	Core – Functions – Interoperation – Broadcast

**Table E.2 – Requirements crosswalk from Topic 21 v2.0 to v1.0**

20-040 Topic 21 v2.0		15-104r5	Topic 21 v1.0
#	Requirement	#	Requirement
19-25	Common Classes	-	-
26	Core – Reference system – Global Domain	2	Reference Frame – Global Domain – Surface Area Equivalence
27	Core – Reference system – Global Domain – Complete		
28	Core – Reference system – Global Domain – Unique	3	Reference Frame – Global Domain – Cell Boundary Overlap
29	Core – Reference system – Cells – Simple	-	(this is more general than 6, since it is spatio-temporal)
30	Core – Reference system – Cell – Direct Position	-	(this is both more general than 13, since it is spatio-temporal and less stringent since it doesn't enforce the centroid)
31	Core – Reference system – Cell – Address	11	Reference Frame – Cell Addressing
		12	Reference Frame – Spatial Reference
32	Core – Reference system – Discrete Global Grid	-	(this is less stringent than 4)
33	Core – Reference system – Discrete Global Grid – Sequence	4	Reference Frame – Tessellation Sequence
		10	Reference Frame – Cell Refinement

34	Core – Functions – Quantization	14	Functional Algorithms – Quantization Operations
		1	Core Data Model
35	Core – Functions – Query – Zone Query	15	Functional Algorithms – Algebraic Processes – Cell Navigation
		16	Functional Algorithms – Algebraic Processes – Spatial Analysis
		1	Core Data Model
36	Core – Functions – Interoperation – Query	17	Functional Algorithms – Interoperability Query Operations
		1	Core Data Model
37	Core – Functions – Interoperation – Broadcast	18	Functional Algorithms – Interoperability Broadcast Operations
		1	Core Data Model
38	Equal Area – Earth Reference System – Harmonized Model	1	Core Data Model
39	Equal Area – Earth Reference System – Global Domain	2	Reference Frame – Global Domain – Surface Area Equivalence
40	Equal Area – Earth Reference System – Tessellation – Initial	9	Reference Frame – Initial Tessellation
41	Equal Area – Earth Reference System – Tessellation – Sequence	4	Reference Frame – Tessellation Sequence
42	Equal Area – Earth Reference System – Tessellation – Sequence – Max		
43	Equal Area – Earth Reference System – Tessellation – Global Area Preservation	5	Reference Frame – Global Area Preservation
44	Equal Area – Earth Reference System – Cell – Simple – 2D Polygon	6	Reference Frame – Cell Shape
45	Equal Area – Earth Reference System – Cell – Direct Position – Centroid	13	Reference Frame – Spatial Reference – Cells Referenced at their Centroid
46	Equal Area – Earth Reference System – Cell – Equal Area – Error Budget	7	Reference Frame – Equal Area Precision
47	Equal Area – Earth Reference System – Cell – Equal Area	8	Reference Frame – Equal Area Cells



# ANNEX F (NORMATIVE) REVISION HISTORY

---

F

# ANNEX F (NORMATIVE) REVISION HISTORY

Table F.1

Date	Release	Author	Paragraph modified	Description
2018-12-05	1.0.0	Matt Purss	All	Copied OGC Abstract Specification Topic 21 [OGC 15-104r5] into the ISO document template
2019-02-09	1.0.1	Robert Gibb	3.23, Note 1	Corrected 'circule' → 'circle', and changed '-' and '-' → ' – ', consistent with usage elsewhere in this document
			5.2.4.2	Corrected 'rato' to 'ratio'
			5.3.3.1	Corrected 'left hand' to 'left-hand'
			5.3.3.2	Corrected 'right hand' to 'right-hand'
			Annex A	Expanded width of all tables to the page margins
Annex B.3, caption for Figure B.1	Annex B.3, caption for Figure B.1	Inserted reference [48] Corrected use of hyphen to dash, consistent with other dashes in this caption		
	Annex B.5	Corrected hyphen to dash between 'ISO 19111' and 'spatial' Inserted reference [49]		
Annex D Bibliography	Annex D Bibliography	Annex D Bibliography	[3] Corrected 'Forth' to 'Fourth'	
			[10] Inserted missing DOI hyperlink	
			[26] Corrected underlining and blue text and inserted missing DOI hyperlink	
			[27] Corrected 'Chen, Z.-t.' → 'Chen, Z.'	
			[41] Inserted DOI missing hyperlink	
			[42] Inserted DOI missing hyperlink	
			[43] Inserted DOI missing hyperlink	
Added reference [48]				

2019-02-12	1.0.2	Matt Purss	Section 5	Updated width of tables and figures to fill the full width of the page.
2019-02-16	1.0.2	Robert Gibb		Submitted to ISO TC211 secretariat for CD vote
2019-05-13	1.0.3	Robert Gibb		Received from ISO TC211 secretariat as N5025 for preparation as DIS
2019-05-20	1.0.4	Robert Gibb	3.16, 3.21	<b>TMG042, TMG045</b> Added abbreviations to the terms
2019-05-28	1.0.4	Robert Gibb		<b>DK003, DK004, DK014, DK017, TMG019, DK030, DK031, TMG041, DK080, DK081, JP084, DK086, FR087, FR105, FR106, DK109</b> removed the artifacts of word to pdf conversion referred to in the comments
2019-05-28	1.0.4	Robert Gibb	Title page	<b>CN002</b> Changed title to ' Discrete Global Grid Systems – Part 1 Equal Area Earth Reference System', this clarifies that purpose of Part 1 of the DGGs Standard, and makes the equal area explicit.
2019-05-28	1.0.5	Robert Gibb	Introduction, Scope, Annex B, Annex C, All	<b>DK010, GB011, GB021</b> Moved Introduction to a new Annex B – Discrete Global Grid System Theory, and moved all but the second paragraph of the Scope into the Introduction. Incremented document version from 1.0.4 to 1.0.5 because many section numbers have changed.
2019-05-29	1.0.5	Robert Gibb	Annex B, last para	<b>DK015</b> Expanded DGGs abbreviation to 'discrete global grids' and made other changes to the paragraph to further clarify the distinction between historic DGGs, & DGG work and this standard.
2019-05-29	1.0.5	Robert Gibb	Introduction, Scope	<b>DK022</b> Changed the first instances of DGGs to Discrete Global Grid System – Equal Area Earth Reference System (DGGs) in Introduction and Scope, this also reflects the change in title.
2019-05-29	1.0.5	Robert Gibb	Introduction, Scope, Bibliography	<b>DK024, DK025</b> Now in Introduction not Scope, so added references to OWS, WCS, and WCPS. 'DGGs core Abstract Specification' (now in Introduction) refers to this document ISO19170. Have made this explicit.
2019-05-29	1.0.5	Robert Gibb	Intro para1, 3.20 Note 1, 5.1, 5.2.1, 5.2.2, 5.2.5.1, 5.2.5.3, 5.2.6, 5.2.6.1, 5.3,	<b>GB020</b> must adhere → adheres, must represent → represents, be compliant → comply, must define → shall define, and provide → to form, must also include → includes, must be defined → shall be defined, must define → shall define, must be → is, must → shall, must → shall, must

				→ shall, must include → includes, must → shall, mechanisms → quantisation methods, however, must always be fixed and be defined → is however, always fixed and always defined, must → shall, must also be able → shall also define methods, must be able to → needs to.
2019-05-29	1.0.5	Robert Gibb	Foreward	<b>TMG018</b> Changed the references to ISO/IEC Directives, Part 2, Revision 8, 2018, and added collaboration with OGC.
2019-05-29	1.0.5	Robert Gibb	3.4, 3.11, 3.26, 3.41, 3.56, 3.57	<b>TMG029, TMG039, TMG048, TMG062, TMG077, TMG078</b> Consolidated instances of sources with notes from a different source or modified notes, as recommended by TMG
2019-05-29	1.0.5	Robert Gibb	3.6, 3.19, 3.30, 3.31, 3.32, 3.38, 3.40, 3.46, 3.47, 3.49	<b>TMG033, TMG043, TMG051, TMG053, TMG054, TMG061, TMG061, TMG068, TMG069, TMG072</b> Updated to conform with revision of ISO19107:2019
2019-05-29	1.0.5	Robert Gibb	3.7, 3.8, 3.12, 3.51	<b>TMG034, TMG035, TMG040, TMG073</b> Updated to conform with revision of ISO19111:2019
2019-05-29	1.0.5	Robert Gibb	3.20, 3.23, 3.243.28, 3.29, 3.33, 3.34, 3.36, 3.43, 3.45, 3.48, 3.55	<b>TMG044, TMG046, TMG047, TMG049, TMG050, TMG055, TMG057, TMG059, TMG065, TMG067, TMG071, TMG076</b> Added <DGGs> domain to the definition
2019-05-29	1.0.5	Robert Gibb	3.2 Para2	<b>DK027</b> Changed 'arenas' → 'subject domains'
2019-05-29	1.0.5	Robert Gibb	3.3 Note2	<b>DK028</b> Changed 'all have unique parents or child cells that may share parents' to 'each have a single parent or child cells that have multiple parents'
2019-05-29	1.0.5	Robert Gibb	3.9, 3.10, 3.33, 3.34	<b>TMG036, DK037, TMG038, DK056</b> Added <DGGs> domain to the definition, and deleted the word 'role' from the definitions
2019-05-29	1.0.5	Robert Gibb	3.35	<b>TMG058</b> removed reference to 19136, moved source 19123 to after the note
2019-05-29	1.0.5	Robert Gibb	Introduction, 3.45, 3.48	<b>DK025, TMG066, TMG070</b> replaced references to this 'Abstract Specification' to this 'document'
2019-05-30	1.0.5	Robert Gibb	Foreward, 2, 3, 4.2	<b>TMG018, DK079</b> Acknowledge the role of ISO19104 in the formulation of this document and the associated UML model.
2019-05-30	1.0.5	Robert Gibb	5.2.4.2, Annex C	Fix a few ISO Style issues e.g. 1m – 1 m, 30m 30 m



2019-05-30	1.0.5	Robert Gibb	Intro, Scope, 5, Annex A, Annex B	<b>CN002, DK104, FR107</b> Aligned the document and URI structures, and ensured that terminology was consistent across everything that looks like a function, including replacing Core with EA-ERS.
2019-05-30	1.0.5	Robert Gibb	5.2.1	<b>DK089</b> changed 'defined' → 'expressed'
2019-05-30	1.0.5	Robert Gibb	Introduction, 5.2.4.2	<b>DK096</b> Inserted a reference to ISO 19112:2019 and clarified use of geodetic identifier to enforce the distinction between the two.
2019-05-30	1.0.5	Robert Gibb	5.2.1	<b>DK088</b> Added a sentence to elaborate the meaning, and recast the Requirements in terms of completeness and uniqueness.
2019-05-31	1.0.5	Robert Gibb	Annex C.4	<b>DK110, DK111</b> Changed Criterium → Criteria, and 'representing a single point' → 'whose centroid represents a single point'
2019-05-31	1.0.5	Robert Gibb	Abbreviated terms	<b>DK092, DK103</b> Added new terms to the Abbreviated Terms list
2019-06-03	1.0.5	Robert Gibb	Fig 8	<b>DK083, DK085</b> Split Spatial Reference figure into three figures and update all figures to reflect UML in newer editions of 19107 & 19111, implement HMMG advice.
2019-06-03	1.0.5	Robert Gibb	TermDef	Reviewed Terms and Definitions wrt updates in 19107, 19111, 19112, 19115, 19161-2, introduced internal T&C links, and removed retired terms that retain their previous meaning: curve, object, point, sequence, set, solid, surface
2019-07-22	1.0.5	Robert Gibb	Title page, NormRefs, TermDef	Rename as Part 1 as discussed at June ISO & OGC/TC meetings.
2019-10-18	1.0.6	Robert Gibb	5, Annex A, All	<b>CN002</b> As discussed at June ISO & OGC/TC meetings, reordered and restructured Sections 5 and A. Conformance to split into Core and Equal Area Earth Reference System modules, and added specific support for DGGS on static and dynamic datums. Incremented document version from 1.0.5 to 1.0.6 because many section numbers have changed.
2019-10-26	1.0.6	Robert Gibb	5.2.2, Annex A	Elaborated the way cellQuery2D differs from Query2D and the described the behaviour of parent, child and sibling functions
2019-10-27	1.0.6	Robert Gibb	5.3.2.2, Annex A	Introduced error budget as an improved way of deriving and describing equal area precision, changed to the wording of the

equal area precision requirement, and added a new requirement for limiting the number of tessellations

2019-10-28	1.0.6	Robert Gibb	NormRefs, TermDef	Updated 19107-FDIS to 19107:2019, and resolved source reference for direct position which has been retired as a term but retained as a data type in 19107:2019
2019-10-28	1.0.6	Robert Gibb	Annex B Fig B.5	<b>DK013</b> Fig B.5 (was Fig 5), left hand example replaced with monochrome one.
2019-10-28	1.0.6	Robert Gibb	Annex B para 1	<b>DK016</b> removed 'framework' from annex B, para 1
2019-10-28	1.0.6	Robert Gibb	TermDef	<b>TMG0443, TMG074, TMG075</b> , terms 'direct position', 'spatial reference system' and 'surface' deleted
2019-10-28	1.0.6	Robert Gibb	5	<b>DK082, DK083</b> all UML diagrams are now monochrome.
2019-10-30	1.0.7	Robert Gibb	Title page, Introduction, Scope, Abbr, 5, 5.1, 5.2, 6 (was 5.3)	Added Reference System to the Core as a new section 5.2.1, and separated it out from 5.3. EAERS is a new conformance class section 6 (from 5.3) Inserted new Fig 2 and 3, increasing numbers of all following figures in Sect 5 by 2. <b>DK093, DK094, DK095, DK097, DK098, DK099, DK100, DK101</b> clarified by new wording Incremented document version from 1.0.6 to 1.0.7 because many section numbers have changed.
2019-10-30	1.0.7	Robert Gibb	Requirements & Conformance	Renumbered and split as per: 1 to 1, 2..5 to 11..14, 6 to 3, 4 & 15, 7 to 5, 8 to 6, 9..11 to 18..20, 12 to 9, 10 & 21, 13..14 to 22..23, 16 to 2, 17 to 17, 18 to 7 & 8. And reworked wording to match revised 5.2.1 & 5.3.
2019-10-30	1.0.7	Robert Gibb	5.2.1.2	<b>DK091</b> a) 'not self-intersect' and b) 'topologically the same as a circle' introduced.
2019-11-06	1.0.7	Robert Gibb	Annex A	Insert 'Requirement n:' in Requirement line of each conformance test.
2019-11-06	1.0.7	Robert Gibb	Abbreviations	Removed DGGs-RS, & HEALPIX (retaining rHEALPIX), changed DGGs-EAERS to EAERS, and made more use of EAERS
2020-03-20	1.0.8	Robert Gibb	All	Created Spatio-temporal Common Class package with conformance classes for geometry and topology, zones and identifiers, and zonal query, their terms and definitions, conformance classes, requirements, defining tables, and abstract tests. Restructured all preamble sections and the

				core to accommodate the new common classes. Inserted a new Annex describing the relationship of the temporal classes to ISO19108.
2020-05-30	1.0.8	Robert Gibb		Submitted for ISO/DIS ballot, published as N5348
2020-06-02	1.0.8	Robert Gibb	Introduction, Annex E	Restructured for OGC template, added summary of v1.0 to v2.0 changes to introduction (OGC only), added Annex E – Requirements v1.0 to v2.0 cross-walk tables (OGC only), published to OGC pending documents as 20-040
2020-06-11	1.0.8	Robert Gibb	Scope, Table 30, Fig 16; Req 47 URI, Fig C.2, Fig B.4, Annex A	changes submitted as DIS comments: Scope bullet 1.1; add description for extent (in EA); Fig 16 remove duplicate 'sibling' (in EA); Requirement 47 insert 'ea/ers/' in URI; Fig C.2 b), c), switch labelling; B.4 b) change 30deg to 90deg and 10deg to 30deg, also superscript deg; Annex A prepend 'conf/' to all abbreviated URI;
2020-12-06	1.0.10	Robert Gibb	All	Align text with changes in response to TC211 comments from both DIS ballot and ISOCS editor
2020-12-07	2.0.0	Robert Gibb	i-vii, 2 (OGC only)	Finalise OGC front material, incl. list of OGC submitters. Add OGC Conformance class tables as section 2, duplicating the summary tables in Abstract Test Suite, publish to OGC pending documents as 20-040r3 Topic 21 v2.0 Part 1
2021-07-16	2.0.0	Robert Gibb	All	Bring changes made during ISO pre-publication proof phase to OGC document.



# BIBLIOGRAPHY





## BIBLIOGRAPHY

---

1. Mahdavi-Amiri, A., Samavati, F. F., Peterson, P., 2015, "Categorization and conversions for indexing methods of discrete global grid systems", *ISPRS International Journal of Geo-Information*, 4(1), pp 320–336. <https://dx.doi.org/10.3390/ijgi4010320>DOI:10.3390/ijgi4010320
2. Mahdavi-Amiri, A., Alderson, T., & Samavati, F., 2015, "A Survey of Digital Earth Representation and Visualization", *Computers & Graphics*, Elsevier Ltd., pp. 95–117. <https://dx.doi.org/10.11575/PRISM/30985>DOI:10.11575/PRISM/30985
3. Clementini, E., Di Felice, P., van Oosterom, P., 1993. "A small set of formal topological relationships suitable for end-user interaction". In *Abel, David; Ooi, Beng Chin. Advances in Spatial Databases: Third International Symposium, SSD '93 Singapore, June 23–25, 1993 Proceedings. Lecture Notes in Computer Science. 692/1993. Springer.* pp 277–295. [https://dx.doi.org/10.1007/3-540-56869-7\\_16](https://dx.doi.org/10.1007/3-540-56869-7_16)DOI:10.1007/3-540-56869-7\_16.
4. Clementini, E., Sharma, J., Egenhofer, M. J., 1994, "Modelling topological spatial relations: Strategies for query processing". *Computers & Graphics* 18 (6), pp 815–822. [https://dx.doi.org/10.1016/0097-8493\(94\)90007-8](https://dx.doi.org/10.1016/0097-8493(94)90007-8)DOI:10.1016/0097-8493(94)90007-8.
5. Clementini, E., and Di Felice P. A., 1996, "A Model for Representing Topological Relationships between Complex Geometric Features in Spatial Databases." *Information Sciences*, 90(1), pp 121–136. [http://dx.doi.org/10.1016/0020-0255\(95\)00289-8](http://dx.doi.org/10.1016/0020-0255(95)00289-8)DOI:10.1016/0020-0255(95)00289-8
6. Egenhofer, M.J.; Herring, J.R., 1990, "A Mathematical Framework for the Definition of Topological Relationships"<sup>1</sup>. *Fourth International Symposium on Spatial Data Handling Zürich, Switzerland*, pp 803–813.
7. Goodchild, M. F., 2000, "Discrete global grids for digital earth"<sup>2</sup>, *International Conference on Discrete Global Grids*. Santa Barbara.
8. Peterson, P. R., 2017, "Discrete Global Grid Systems." In *The International Encyclopedia of Geography*, edited by Douglas Richardson. Malden, Oxford: John Wiley and Sons, Ltd. DOI:10.1002/9781118786352.wbieg1050
9. Sahr, K., White, D., Kimerling, A. J., 2003, "Geodesic discrete global grid systems", *Cartography and Geographic Information Science*, 30 (2), pp 121–134. <https://dx.doi.org/10.1559/152304003100011090>DOI:10.1559/152304003100011090
10. Postel, J., 1981, "RFC 791: Internet protocol"<sup>3</sup>, *Darpa Internet Protocol Specification (September 1981)*.
11. Pacioli, L., 1509, "De Davina Proportione" (Illustrated by Leonardo Da Vinci), Venice.
12. da Vinci, L., 1514, "Manuscript E, 24v-25r (1513-14)". Pen and ink on paper, 150 x 99 mm. Bibliothèque de l'Institut de France, Paris.

13. da Vinci, L, 1478–1518, “Codex Atlanticus”. Pen and ink on paper, 61 x 44 cm. (collated by Pompeo Leoni). Biblioteca Ambrosiana, Milan, Italy.
14. “Life Presents R. Buckminster Fuller’s Dymaxion World”. *Life*, 1 March 1943, pp 41–45
15. Fuller, R. B., 1946, Cartography<sup>4</sup>. U.S. Patent 2,393,676.
16. Fuller, R. B. and Sadao, S., 1954, “Dymaxion Airocean World: The Raleigh Edition of Fuller Projection”<sup>5</sup>. *Student Publications of the School of Design*, North Carolina State College, Raleigh, North Carolina, USA.
17. Vestine, E. H. Sibley, W. L. Kern, J. W. and Carlstedt, J. L. “Integral and Spherical-Harmonic Analyses of the Geomagnetic Field for 1955.0, Part 2,” *Journal of Geomagnetism and Geoelectricity*, vol. 15, No. 2, Aug. 1963, pp. 73–89.
18. Sadourny, R.; Arakawa, A.; Mintz, Y. (1968). “Integration of the nondivergent barotropic vorticity equation with an icosahedral-hexagonal grid for the sphere”. *Monthly Weather Review* 96 (6): 351–356.
19. Dutton, G., 1989, “Modelling location uncertainty via hierarchical tessellation”. In: *Accuracy of Spatial Databases*, Goodchild, M. and Gopal, S. (eds.), London: Taylor & Francis, pp 125–140.
20. Dutton, G., 1999, “A hierarchical coordinate system for geoprocessing and cartography”. *Lecture Notes in Earth Science* 79. Berlin: Springer-Verlag. XIX + 231 pp. 97 figs. 12 plates, 16 tabs. ISSN 0930-0317, ISBN 3-540-64980-8.
21. Dutton, G., 1984, “Part 4: Mathematical, Algorithmic and Data Structure Issues: Geodesic Modelling Of Planetary Relief”, *Cartographica: The International Journal for Geographic Information and Geovisualization*, 21(2–3), pp 188–207. <https://dx.doi.org/10.3138/R613-191U-7255-082N>
22. Dutton, G., 1996, “Encoding and handling geospatial data with hierarchical triangular meshes”, *In Proceeding of 7th International symposium on spatial data handling*, 43, Netherlands: Talyor & Francis. DOI:10.1.1.461.5603
23. Dutton, G., 1999, “Scale, Sinuosity, and Point Selection in Digital Line Generalization”, *Cartography and Geographic Information Science*, 26(1), pp 33–54. <https://doi.org/10.1559/152304099782424929>
24. Tobler, W., & Chen, Z.-t., 1986, “A Quadtree for Global Information Storage”, *Geographical Analysis*, 18(4), pp 360–371. <https://dx.doi.org/10.1111/j.1538-4632.1986.tb00108.x>
25. White, D., Kimerling, J., & Overton, W. S., 1992, “Cartographic and geometric components of a global sampling design for environmental monitoring”, *Cartography and Geographic Information Systems*, 19(1), pp 5–22. <https://dx.doi.org/10.1559/152304092783786636>
26. Kimerling, A. J., Sahr, K., White, D., & Song, L., 1999, “Comparing Geometrical Properties of Global Grids”, *Cartography and Geographic*

*Information Systems* vol. 26, no. 4, pp 271–88. <https://dx.doi.org/10.1559/152304099782294186> DOI:10.1559/152304099782294186

27. Dutton, G., 1994, "Geographical grid models for environmental monitoring and analysis across the globe (panel session)". In: Proceedings of GIS/US'94 Conference, Phoenix, Arizona.
28. Snyder, J. P., 1992, "An equal-area map projection for polyhedral globes". *Cartographica*, 29(1), pp.10–21.
29. Song, L., 1997, "Small circle subdivision method for development of global sampling grid", *Unpublished Masters Thesis*, Oregon State University, 176pp.
30. Song, L., Kimerling, A. J., and Sahr, K., 2002, "Developing an Equal Area Global Grid by Small Circle Subdivision"<sup>6</sup>. In M. Goodchild and A. J. Kimerling (Eds.), *Discrete Global Grids: A Web Book*
31. Lukatela, H., 1987, "Hipparchus geopositioning model: An overview", In: *Proceedings of Auto Carto 8 Symposium, Baltimore, Maryland*. pp 87–96.
32. Fekete, G., Treinish, L., 1990, "Sphere quadtree: A new data structure to support the visualization of spherically distributed data". *SPIE: Extracting Meaning from complex data: Processing, display, interaction*. 1259, pp 242–253.
33. Bjorke, J. T., Grytten, J. K., Haeger, M., Nilsen, S., 2003, "A Global Grid Model Based on 'Constant Area' Quadrilaterals", *ScanGIS*, 3, pp 238–250. DOI: 10.1.1.105.2253
34. Amante, C. and Eakins, B. W., 2009, "ETOPO1 1 Arc-Minute Global Relief Model: Procedures, Data Sources and Analysis", *NOAA Technical Memorandum NESDIS NGDC-24*, 19 pp.
35. Rees, T., 2002, "'C-Squares' – a new metadata element for improved spatial querying and representation of spatial dataset coverage in metadata records", in *EOGEO 2002 Proceedings, Ispra, Italy*.
36. Loveland, T. R., Merchant, J. W., Ohlen, D., O., Brown, J. F., 1991, "Development of a land-cover characteristics database for the conterminous U.S.". *Photogrammetric Engineering and Remote Sensing*, 57(11), pp 1453–1463
37. Snyder J. P., 1993, "Flattening the Earth: Two Thousand Years of Map Projections", pp5–8. ISBN 0-226-76747-7)
38. Gibb, R.G., 2016, "The rHEALPix Discrete Global Grid System", Proceedings of the 9<sup>th</sup> Symposium of the International Society for Digital Earth (ISDE), Halifax, Nova Scotia, Canada. *IOP Conf. Ser.: Earth Env. Sci.*, 34, 012012. DOI:10.1088/1755-1315/34/1/012012
39. Ma, T., Zhou, C., Xie, Y., Qin, B., Ou, Y., 2009, "A discrete square global grid system based on the parallels plane projection". *International Journal of Geographical Information Science*. 23(10), pp 1297–1313. <https://doi.org/10.1080/13658810802344150> DOI:10.1080/13658810802344150



40. Fisher, I., 1943, "A World Map on a Regular Icosahedron by Gnomonic Projection", *Geographical Review*, 33(4), pp 605–619, <https://www.jstor.org/stable/209914>DOI:10.2307/209914
41. Fuller, R. B., 1982, "Synergetics", New York.
42. Clarke, K. C., 2000, "Criteria and Measures for the Comparison of Global Geocoding Systems"<sup>7</sup>, *International Conference on Discrete Global Grids. Santa Barbara: University of California, Santa Barbara.*
43. Fitzpatrick, R., 2008. *Euclid's Elements of Geometry. Richard Fitzpatrick.* ISBN 978-0-6151-7984-1
44. Schmidt, R., Grimm, C., Wyvill, B., 2006, Interactive decal compositing with discrete exponential maps, *ACM Transactions on Graphics*, 25 (3), pp 605–613.
45. Cheng, C., Tong, X., Chen, B., Zhai, W., 2016, "A Subdivision Method to Unify the Existing Latitude and Longitude Grids", *ISPRS International Journal of Geo-Information*, 5(9), 161, DOI:10.3390/ijgi5090161
46. Li, S., Cheng, C., Chen, B., Meng, M., 2016, "Integration and management of massive remote-sensing data based on GeoSOT subdivision model", *J. Appl. Remote Sens.* 10(3), 034003, DOI:10.1117/1.JRS.10.034003
47. Allen, J. F., 1983, "Maintaining Knowledge about Temporal Intervals", *Communications of the ACM*, vol. 26 pp. 832–843
48. ISO/IEC: ISO/IEC 2382:2015, *Information technology – Vocabulary.* International Organization for Standardization, International Electrotechnical Commission, Geneva (2015). <https://www.iso.org/standard/63598.html>
49. ISO: ISO 8601-1:2019, *Date and time – Representations for information interchange – Part 1: Basic rules.* International Organization for Standardization, Geneva (2019). <https://www.iso.org/standard/70907.html>
50. ISO: ISO 19101-1:2014, *Geographic information – Reference model – Part 1: Fundamentals.* International Organization for Standardization, Geneva (2014). <https://www.iso.org/standard/59164.html>
51. ISO: ISO 19103:2015, *Geographic information – Conceptual schema language.* International Organization for Standardization, Geneva (2015). <https://www.iso.org/standard/56734.html>
52. ISO: ISO 19107:2019, *Geographic information – Spatial schema.* International Organization for Standardization, Geneva (2019). <https://www.iso.org/standard/66175.html>
53. ISO: ISO 19108:2002, *Geographic information – Temporal schema.* International Organization for Standardization, Geneva (2002). <https://www.iso.org/standard/26013.html>



54. ISO: ISO 19111:2019, *Geographic information – Referencing by coordinates*. International Organization for Standardization, Geneva (2019). <https://www.iso.org/standard/74039.html>
55. ISO: ISO 19112:2019, *Geographic information – Spatial referencing by geographic identifiers*. International Organization for Standardization, Geneva (2019). <https://www.iso.org/standard/70742.html>
56. ISO: ISO 19115-1:2014, *Geographic information – Metadata – Part 1: Fundamentals*. International Organization for Standardization, Geneva (2014). <https://www.iso.org/standard/53798.html>
57. ISO: ISO 19117:2012, *Geographic information – Portrayal*. International Organization for Standardization, Geneva (2012). <https://www.iso.org/standard/46226.html>
58. ISO: ISO 19123:2005, *Geographic information – Schema for coverage geometry and functions*. International Organization for Standardization, Geneva (2005). <https://www.iso.org/standard/40121.html>
59. ISO: ISO 19156:2011, *Geographic information – Observations and measurements*. International Organization for Standardization, Geneva (2011). <https://www.iso.org/standard/32574.html>
60. Arliss Whiteside: OGC 05-042r2, *OpenGIS Web services architecture description*. Open Geospatial Consortium (2005). [https://portal.ogc.org/files/?artifact\\_id=13140](https://portal.ogc.org/files/?artifact_id=13140)
61. Arliss Whiteside Jim Greenwood : OGC 06-121r9, *OGC Web Service Common Implementation Specification*. Open Geospatial Consortium (2010). [https://portal.ogc.org/files/?artifact\\_id=38867](https://portal.ogc.org/files/?artifact_id=38867)
62. Simon Cox, Chris Little: OGC 16-071r3, *Time Ontology in OWL*. Open Geospatial Consortium (2020). <https://www.w3.org/TR/2020/CR-owl-time-20200326/>
63. Peter Baumann: OGC 17-089r1, *OGC Web Coverage Service (WCS) 2.1 Interface Standard – Core*. Open Geospatial Consortium (2018). <http://docs.opengeospatial.org/is/17-089r1/17-089r1.html>