

# Table of Contents

1. Scope .....	5
1.1. What this Abstract Specification is .....	5
1.2. What this Abstract Specification is not .....	5
2. Conformance .....	6
3. References .....	7
4. Terms and Definitions .....	8
4.1. <b>conceptual model</b> .....	8
4.2. <b>coordinate reference system</b> .....	8
4.3. <b>coordinate system</b> .....	8
4.4. <b>equilateral triangle</b> .....	8
4.5. <b>extension</b> .....	8
4.6. <b>hexagon</b> .....	8
4.7. <b>periodic tiling</b> .....	8
4.8. <b>profile</b> .....	9
4.9. <b>profile with extension(s)</b> .....	9
4.10. <b>space partitioning</b> .....	9
4.11. <b>tessellation</b> .....	9
4.12. <b>tile</b> .....	9
4.13. <b>tile scheme</b> .....	10
4.14. <b>tiling</b> .....	10
4.15. <b>tile set</b> .....	10
4.16. <b>tile set scheme</b> .....	10
5. Conventions .....	11
5.1. Abbreviations .....	11
5.2. Identifiers .....	11
6. Introduction .....	12
6.1. Key concept - The assumption of a 2D planar space .....	12
6.2. Two key use cases .....	13
6.2.1. Storage Use Case .....	13
6.2.2. Visualization Use Case .....	14
6.3. Characteristics of a Conceptual Model .....	15
6.4. Logical Model .....	15
6.4.1. Conceptual Model vs. Logical (Data) Model .....	16
7. Part 1: Conceptual Model for tiling any space .....	17
7.1. General Tile/Tiling Conceptual Model .....	17
8. Part 2: Logical Model for 2D Euclidean Space Tiles and Tiling: Normative .....	19
8.1. Restrictions to definitions of key concepts for the tiling of 2D plane .....	19
8.2. Base Requirement .....	20

8.2.1. Requirement 1 - Core .....	20
8.3. Requirement Class Tile Set Scheme .....	20
8.3.1. Requirement 2 - Tile Set Identifier (ID) .....	20
8.3.2. Requirement 3 - Tile Set Coordinate Reference System (CRS) .....	21
8.3.3. Requirement 4 - Tile Set Units of Measure (UoM) .....	21
8.3.4. Requirement 5 - Tile Set Extent .....	21
8.3.5. Requirement 6 - Tile Set Origin .....	22
8.4. Tessellation class .....	22
8.4.1. Tessellation Type square .....	23
8.4.2. Tessellation Type triangle .....	23
8.4.3. Tessellation Type Hexagon .....	24
8.5. Requirements Class: Tile Scheme .....	24
8.5.1. Requirement 9 - Tile Address (ID) .....	25
8.5.2. Requirement 10 - Tile Origin .....	25
8.5.3. Requirement 11 - Tile Reference .....	25
8.5.4. Requirement 12 - Tile Extent .....	26
8.6. Recommended Tile Set Metadata Class .....	26
8.7. Notes on Tile Indexes, Tile Pyramids, and regular tessellations .....	27
8.7.1. Tile indexes .....	27
8.7.2. Tile Pyramids .....	27
8.7.3. Regular Tessellations .....	27
Annex B A: Conformance Class Abstract Test Suite (Normative) .....	28
A.1. Conformance Class A .....	28
A.1.1. Requirement 1 - Core statement .....	28
A.2. Conformance Class Tile Set Schema .....	28
A.2.1. Requirement 2 .....	28
A.2.2. Requirement 3 .....	28
A.2.3. Requirement 4 .....	28
A.2.4. Requirement 5 .....	29
A.2.5. Requirement 6 .....	29
A.2.6. Requirement 7 .....	29
A.3. Conformance Class Tile .....	29
A.3.1. Requirement 9 .....	29
A.3.2. Requirement 10 .....	29
A.3.3. Requirement 12 .....	30
Additional Informative Material: Simple Example of an XML encoding .....	31
Annex B: Revision History .....	32
Annex C: Bibliography .....	33



**Open Geospatial Consortium**

Submission Date: 2019-MAY-01

Approval Date: <2019-mm-dd>

Publication Date: <2019-mm-dd>

External identifier of this OGC® document: <http://www.opengis.net/doc/AS/2D-tiles/1.0>

Internal reference number of this OGC® document: 19-014r1

Version: 1.0

Category: OGC® Candidate Abstract Specification

Editor: Carl Reed, PhD

## **Core Tiling Conceptual and Logical Models for 2D Euclidean Space**

### **Copyright notice**

Copyright © 2019 Open Geospatial Consortium

To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>

### **Warning**

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

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type: OGC® Abstract Specification

Document subtype: if applicable

Document stage: Draft

Document language: English

## License Agreement

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

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

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD.

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

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

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize

you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications. This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

## **i. Abstract**

This OGC Abstract Specification defines:

- A conceptual model for tiling space in any dimension and;
- A logical model for 2D tiled structures and by extension tiling. The logical model is based on the conceptual model.

The conceptual model specified in this Abstract Specification could be a sub-class in a more comprehensive Spatial Partitioning Conceptual Model.

## **ii. Keywords**

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

ogcdoc, OGC document, tiles, tiling, 2D, tessellation, tile set

## **iii. Preface**

Numerous OGC standards specify some type of tiling scheme. These standards include Web Map Tiling Service (WMTS), CDB, and GeoPackage. There have also been a number of OGC Innovation Initiatives focused on tiling and tiling schemes. However, a general tiling conceptual model and related logical model for 2-D Euclidian space have not been defined. This Abstract Specification defines a tiling conceptual model and a logical model for tiling 2-D planar space that is based on a tiling conceptual model.

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

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

## **iv. Submitting organizations**

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

- Carl Reed & Associates
- UK Met Office

## **v. Submitters**

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

<b>Name</b>	<b>Affiliation</b>
Carl Reed	Carl Reed & Associates
Chris Little	UK Met Office

# Chapter 1. Scope

## 1.1. What this Abstract Specification is

This OGC Abstract Specification consists of two main Parts: A General Tiling Conceptual Model and Logical Model for the Tessellation (Tiling) of 2D Euclidean Space.

Tiling of 2D Euclidean space is the most commonly known approach to partitioning space in traditional geospatial technology. However, there are also common elements and/or semantics for any approach to partitioning space in any dimension. The logical model in this document defines a set common required elements and then follows with more specific requirements for the two dimensional case.

More specifically, Part 1 of the Abstract Specification describes a general tiling conceptual model. The conceptual model is applicable to any dimension. The conceptual model makes no assumptions regarding content, use cases, implementation scenarios, or how the space is to be tessellated (tiled). The conceptual model is abstract and cannot be implemented as is.

Therefore Part 2 of this Abstract Specification defines a detailed logical model for the tessellation of 2D Euclidean Space. One or more logical models are required to provide the requirements and structure necessary for implementation. Therefore, in addition to the conceptual model, this Abstract Specification also specifies a core logical model for the 2D planar (Euclidean) use case. Other Parts that specify additional logical models, such as for 3D Euclidean space, may be added in the future

## 1.2. What this Abstract Specification is not

This Abstract Specification **does not**:

- Specify the content that could be organized in a tiled structure;
- Address concepts such as styling, levels of detail, attributes, and levels;
- Make any suggestions regarding how content will be processed and stored in the tiled structure;
- Provide guidance on formats, encodings, or any other implementation details.

**NOTE:** In this Abstract Specification, "tile" is NOT a packaged blob of data to download in a chunky streaming optimization scheme! This is a general misconception based on implementation specific requirements. Other OGC standards such as [Time Matrix Set \(TMS\) Standard](#) provide requirements and details on how to organize and access tiled blobs of geographic data.

The above concepts and implementation guidance is defined in profiles, extensions, and profiles with extensions based on this Abstract Specification. Examples of this type of guidance is the OGC Tile Matrix Set Standard, the OGC I3S Community Standard, and the OGC CDB standard.



# Chapter 2. Conformance

This standard defines Two conformance classes.

Requirements for N standardization target types are considered: \* Tile Requirement Class \* TileSet Requirement Class

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

In order to conform to this OGC® interface standard, a software implementation shall choose to implement: \* Any one of the conformance levels specified in Annex A (normative).

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

# Chapter 3. References

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

ISO / TC 211: ISO 19115-1:2014 Geographic information — Metadata — Part 1: Fundamentals (2014)

OGC: OGC 07-057r7, OGC Web Map Tile Service Implementation Standard (2010)

OGC: OGC 15-113r5, OGC Volume 1: OGC CDB Core Standard: Model and Physical Data Store Structure (2018)

OGC: OGC 15-104r5, Topic 21: Discrete Global Grid Systems Abstract Specification (2017)

OGC: OGC 12-128r15, OGC Geopackage Encoding Standard - with Corrigendum (2018)

OGC: OGC 17-083r2, Tile Matrix Set Standard (2019)

OGC: OGC 06-121r9, OGC Web Services Common (OWS Common) Encoding Standard (2010)

# Chapter 4. Terms and Definitions

This document uses the terms defined in Sub-clause 5.3 of [OGC 06-121r8], which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

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

## 4.1. conceptual model

description of common concepts and their relationships, particularly in order to facilitate exchange of information between parties within a specific domain [CEN ENV 1613:1994]. A conceptual model is explicitly chosen to be independent of design or implementation concerns.

## 4.2. coordinate reference system

coordinate system that is related to the real world by a datum [ISO 19111]

## 4.3. coordinate system

set of mathematical rules for specifying how coordinates are to be assigned to points [ISO 19111]

## 4.4. equilateral triangle

triangle in which all three sides are equal in length. In the familiar Euclidean geometry, an equilateral triangle is also equiangular: That is, all three internal angles are also equal in size and are each 60°.

## 4.5. extension

set of one or more conformance clauses that defines new options for an existing standard. Extensions are generally designed to add additional capabilities that are not part of the core standard. An example is OGC GeoPackage Extension for Tiled Gridded Coverage Data. [OGC TC PnP]

## 4.6. hexagon

six-sided polygon or 6-gon. The total of the internal angles of any simple (non-self-intersecting) hexagon is 720°. A regular hexagon is defined as a hexagon that is both equilateral and equiangular. It is bicentric, meaning that it is both cyclic (has a circumscribed circle) and tangential (has an inscribed circle). All internal angles are 120 degrees.

## 4.7. periodic tiling

tiling that repeats itself at regular intervals. If a region of the tiling can be outlined with a

parallelogram and then tile the rest of the plane by translating that parallelogram (rotations and reflections are not allowed), then the resulting tiling is periodic. Examples of periodic tilings include regular tessellations, tilings that use only congruent regular polygons, such as regular hexagons.

## 4.8. profile

proper subset of an existing standard including restrictions on or deletions of conformance clauses related to the subsetting. An example of a profile is the GML Simple Feature Profile. [ISO 19106:2004, Type 1 Profile]

## 4.9. profile with extension(s)

set of one or more conformance clauses from a base standard that includes at least one new conformance clause (extension). [OGC TC PnP] An example is OpenGIS® Web Map Services - Profile for EO Products.

## 4.10. space partitioning

process of dividing a geometric space (usually a Euclidean space) into two or more disjoint subsets (see also partition of a set). In other words, space partitioning divides a space into non-overlapping regions. Any point in the space can then be identified to lie in exactly one of the regions.

## 4.11. tessellation

partitioning of a space into a set of conterminous subspaces having the same dimension as the space being partitioned [ISO 19123]

### NOTE

*the tiling of two dimensional space using one or more geometric shapes, called tiles, with no overlaps and no gaps. In mathematics, tessellations can be generalized to higher dimensions and a variety of geometries. Regular tessellations are made up entirely of congruent regular polygons all meeting vertex to vertex, and the arrangement of the polygons around every vertex is the same. There are only three regular tessellations. These consist of a network of equilateral triangles, squares or hexagons [2: Irregular tessellations consist of figures that are not composed of regular polygons but still interlock without gaps or overlaps. The pattern of the common brick laying bond is also not a regular tessellation, as some vertices are along the sides of the polygons. Tessellations with more than one kind of polygon are also not regular].*

## 4.12. tile

geometric shape with known properties that is the result of the tiling (tessellation) of a plane. A tile consists of a single connected "piece" without "holes" or "lines" (topological disc).

**NOTE**

In the above definition, the term **hole** means that any given tile in a tileset cannot have sub-tiles or exclusion areas. The use of the term 'hole' in the above definition should not be confused with exclusion areas in a polygon geometry (aka donuts or islands).

## 4.13. tile scheme

scheme that defines the unique properties of each individual tile in a tile set. These properties include a unique identifier for each tile, the tile origin, and the extent of the tile.

## 4.14. tiling

in mathematics, a tiling (of the plane) is a collection of subsets of the plane, i.e. tiles that cover the plane without gaps or overlaps.

## 4.15. tile set

set of tiles with common properties that meets the definition of a tessellation. In short, a collection of subsets of the plane, i.e. tiles, which cover the space without gaps or overlaps.

## 4.16. tile set scheme

scheme that defines how space is partitioned into individual tiles. The scheme defines the spatial reference system, the geometric properties of a tile, which space a uniquely identified tile occupies, and reversely which unique identifier corresponds to a space satisfying the geometric properties to be a tile.

**NOTE**

*While in the general sense, a tiling scheme is not restricted to a coordinate reference system or a tile matrix set and allows for other spatial reference systems such as DGGS and other organizations including irregular ones, the logical model in this abstract specification focuses on the 2D case.*

# Chapter 5. Conventions

This Abstract Specification defines conceptual and logical models that specify mandatory and recommended requirements for tiles, tilesets, and tiling of a 2D Euclidean plane. This Abstract Specification does not mandate any particular coding or implementation pattern (e.g., XML, JSON)

## 5.1. Abbreviations

- 2D 2-dimensional
- TMS Tile Matrix Set
- WMTS Web Map Tiling Service

## 5.2. Identifiers

The normative provisions in this Abstract Specification are denoted by the URI

<http://www.opengis.net/spec/2d-tile-model/1.0/>

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

# Chapter 6. Introduction

For many geospatial technology applications and use cases, there is the need to partition the space of interest. Examples are numerous. Within the OGC standards baseline, [CDB](#) defines how to partition the globe into a set of 2D tiles for 24 levels of detail. The [Web Matrix Tile Set](#) (WMTS) standard defines the rules and requirements for a tile matrix set as a way to index space based on a set of regular grids defining a domain (tile matrix) for a limited list of scales in a restricted set of Coordinate Reference Systems (CRS). The new - and more general - Tile Matrix Set standard specifies the concept of a tile matrix set and tile matrix set limits and its implementation in 2D space. Finally, the [OGC I3S Community Standard](#) defines how to partition 3D space using a hierarchical, node-based spatial index structure in which each node's payload may contain features with associated geometry, textures and attributes.

In geometry, space partitioning is the process of dividing a space (usually a Euclidean space) into two or more disjoint subsets. In other words, space partitioning divides a space into non-overlapping regions. Any point in the space can then be identified to lie in exactly one of the regions.

Within that context, the requirements for partitioning space based on dimension can be considered. This OGC Abstract Specification presents a conceptual and logical model for partitioning (tiling) 2D Euclidean space. Tiling of 2D Euclidean space is the most commonly known approach to partitioning space. However, there are common concepts for any approach to partitioning space in any dimension. This document defines the common concepts that apply to any dimension and then follows with mandatory and recommended requirements for the tiling of 2d Euclidean space use case.

While developing the conceptual model and the associated logical model, the tiling sections/extensions to the OGC CDB, [GeoPackage](#), and WMTS standards along with the commercial MapBox Vector Tiles (MVT) specification were reviewed. The [OGC DGGS Abstract Specification](#) and various Engineering Reports from OGC Test Bed 13, Test Bed 14 and the Vector Tiles Pilot were also considered. Finally, the new [Tile Matrix Set](#) Standard was reviewed. Much of the content and requirements specified in this abstract specification are derived by abstracting what is contained in the existing and candidate standards and specifications.

The goal is to define a simple conceptual model that can support any and all requirements for tiled data stores and applications including extensions for visualization, portrayal, analytics, filtering, levels of detail, and so forth. Various application use cases and/or workflows, such as styling or dealing with topology, are not part of the core logical model. Instead, such applications may be thought of as profiles and/or extensions of the model that restrict core requirements or define additional requirements. For example, the majority of the CDB standard that deals with tiles defines additional requirements, such as level of detail, to optimize the data store for that domain of interest.

## 6.1. Key concept - The assumption of a 2D planar space

The tiling logical model defined in this document assumes that the tiling target is a two dimensional plane in Euclidean space. However, the content to be stored in a tiled structure is typically earth (or planetary body) referenced. Therefore, the transformation from a spheroidal coordinate system

into a planar system is required. The following paragraph, extracted from the Tile Matrix Set Standard, describes the solution for coordinate reference systems other than geographic (Latitude, Longitude).

*As stated in OGC 08-015r6 Abstract Specification Topic 2: Spatial referencing by coordinates, a coordinate system is a set of mathematical rules for specifying how coordinates are to be assigned to points in space. A Coordinate Reference System (CRS) is a coordinate system that is related to the real world by a reference datum. An example of mathematical rules is the application of a sphere or an ellipsoid centered in the datum and the use of a projection to transform the sphere or the ellipsoid into a planar representation of the world.*

## **6.2. Two key use cases**

There are two distinct but related use cases for implementing and using a tiled structure: storage and visualization. In both cases, the driving force is the desire to enhance performance. Performance could be in terms of access latency, caching, and/or streaming of relevant content to a client.

### **6.2.1. Storage Use Case**

In this use case, geospatial content is subdivided into small units called tiles. The intent is to significantly enhance such operations as search, update, and presentation of the source content in its "native" form. In this use case, the original geospatial data (raster, grid, vector, point clouds, etc.) is maintained. There is no special processing to create forms of the data specifically for high speed caching and rendering on the client. Such processing is typically required for the Visualization use case and is described in the next clause.

A very typical storage use case would be a city defining the tile structure for its spatial data store. They could begin by defining the geographic extent of the entire area of interest and could include a fringe area for future growth. Typically in the United States, cities and counties work in State Plane coordinates, such as Colorado North zone (NAD 83) for the City of Ft. Collins. The parameters are available in the EPSG database (EPSG 2876). The units of measure are feet. They pick a tile set origin that is the same for all tile sets in the data store. Typically, GIS practitioners think of origins being in the lower left corner so that option is selected for both the tile set and for each tile in a tile set. They then choose a regular square tessellation. They wish to create three tile sets with unique tileset IDs of Parcels, Streams, and Roads. Each tile set will have a different tile size. Parcels will have a tile width and height of 1000 feet. Streams will have a tile width and height of 5000 feet. Roads will have a tile width and height of 2000 feet.



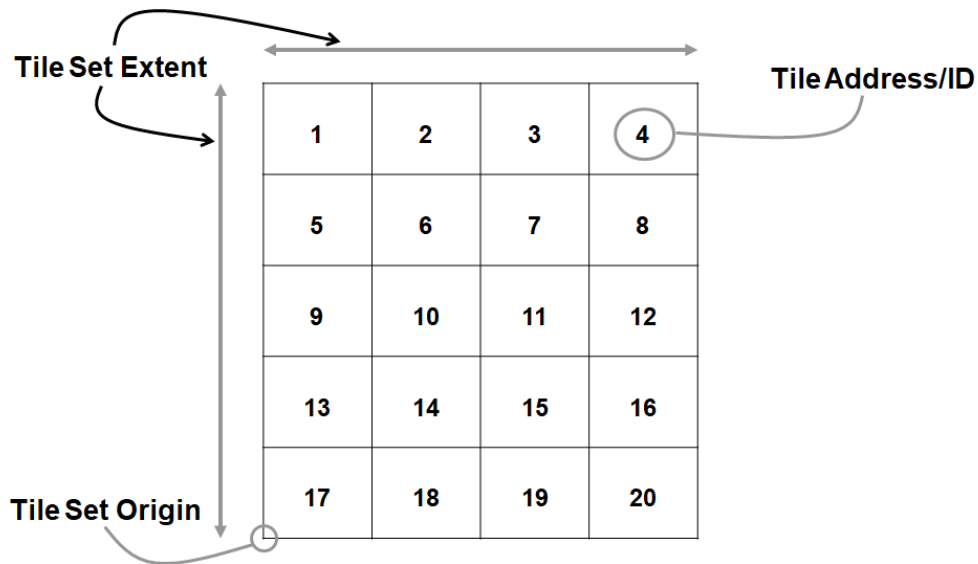


Figure 1. Example tile set for above parcel map use case

### 6.2.2. Visualization Use Case

The Visualization use case is focused on providing high speed geospatial content rendering and visualization capabilities on a client. In this use case, the source geospatial data is heavily processed, restructured, and reformatted to be optimized for visualization on one or more clients. For example, source vector data that is topologically structured and semantically rich may be processed into a tile matrix of PNG images and all geometry, topology, and semantics are lost.

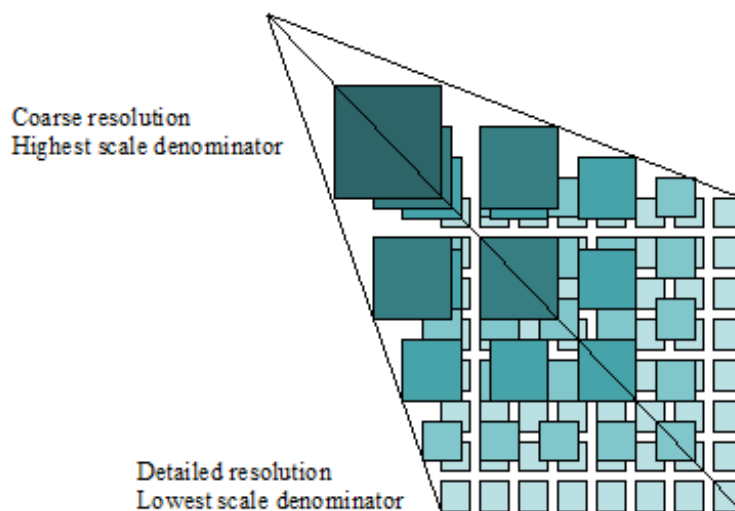


Figure 2. Example tile matrix for visualization

For example, the OGC Tile Matrix Set standard specifies rules for defining a tile matrix. From that standard:

*For the case of a two dimensional space, given the top left point of the tile matrix in CRS coordinates ( $tileMatrixMinX$ ,  $tileMatrixMaxY$ ), the width and height of the tile matrix in tile units ( $matrixWidth$ ,  $matrixHeight$ ), the width and height of a tile ( $tileWidth$ ,  $tileHeight$ ) in original grid cells (often referred to as pixels), the coefficient to convert the coordinate reference system (CRS) units into meters ( $metersPerUnit$ ) and the scale ( $1:scaleDenominator$ ), the bottom right corner of the bounding box of a tile matrix ( $tileMatrixMaxX$ ,  $tileMatrixMinY$ ) can be calculated as follows: . . .*

The TMS model is entirely consistent with the conceptual and logical models defined in this abstract specification – although the property names are different. The OGC TMS can be thought of as a profile with extensions of this abstract specification.

## 6.3. Characteristics of a Conceptual Model

The terms and definitions clause in this Abstract Specification provides a short definition for "conceptual model". This clause provides additional information on the OGC use of "conceptual model".

A conceptual model organizes the vocabulary needed to communicate consistently and thoroughly about the know-how of a problem domain. The aim of a conceptual model is to express the meaning of terms and concepts used by domain experts to discuss the problem, and to find the correct relationships between different concepts.

A conceptual model:

- is a representation of a system, made of the composition of concepts which are used to help people know, understand, or simulate a subject the model represents. A documented conceptual model represents 'concepts' (entities), the relationships between them, and a vocabulary.
- is explicitly defined to be independent of design or implementation concerns.
- organizes the vocabulary needed to communicate consistently and thoroughly about the know-how of a problem domain.
- starts with a glossary of terms and definitions. There is a very high premium on high-quality, design-independent definitions, free of data or implementation biases. The model also emphasizes rich vocabulary.
- is always about identifying the correct choice of terms to use in communications, including statements of rules and requirements, especially where high precision and subtle distinctions need to be made. The core concepts of a geospatial problem domain are typically quite stable over time.

## 6.4. Logical Model

A [logical data model](#) or logical schema is a data model of a specific problem domain expressed independently of a particular database management product or storage technology (physical data model) but in terms of data structures such as relational tables and columns, object-oriented classes, or XML tags. This is as opposed to a conceptual data model, which describes the semantics of an organization without reference to technology.

Logical data models represent the abstract structure of a domain of information. They are often diagrammatic in nature and are most typically used in business processes that seek to capture things of importance to an organization and how they relate to one another. Once validated and approved, the logical data model can become the basis of a physical data model and form the design of a database.

Logical data models should be based on the structures identified in a preceding conceptual data model, since this describes the semantics of the information context, which the logical model

should also reflect. Even so, since the logical data model anticipates implementation on a specific computing system, the content of the logical data model is adjusted to achieve certain efficiencies.

### **6.4.1. Conceptual Model vs. Logical (Data) Model**

A conceptual model differs from a logical model in important ways cite:[Gupta2012]. The goal of a conceptual model is to support the expression of natural-language statements, and supply their semantics — not unify, codify (and sometimes simplify) data. Therefore the vocabulary included in a conceptual model is far richer, as suits knowledge-intensive problem domains. In short, conceptual models are concept-centric; logical models are thing-entity-or-class-centric.

Logical models can usually be rather easily derived from conceptual models; the reverse is much harder (or impossible). Like logical models, conceptual models are often rendered graphically, but free of such distractions to stakeholders such as cardinalities.

# Chapter 7. Part 1: Conceptual Model for tiling any space

Part 1 of this Abstract Specification describes a general tiling conceptual model. The conceptual model is applicable to any dimension. The conceptual model makes no assumptions regarding content, use cases, implementation scenarios, or how the space is to be tessellated (tiled). The conceptual model is abstract and cannot be implemented as is.

Therefore Part 2 of this Abstract Specification defines a detailed logical model for the tessellation of 2D Euclidean Space.

## 7.1. General Tile/Tiling Conceptual Model

The following figure captures the fundamental conceptual model for a space partitioning based on tiles. This conceptual model can be applied to any spatial data independent of any space/time dimension.

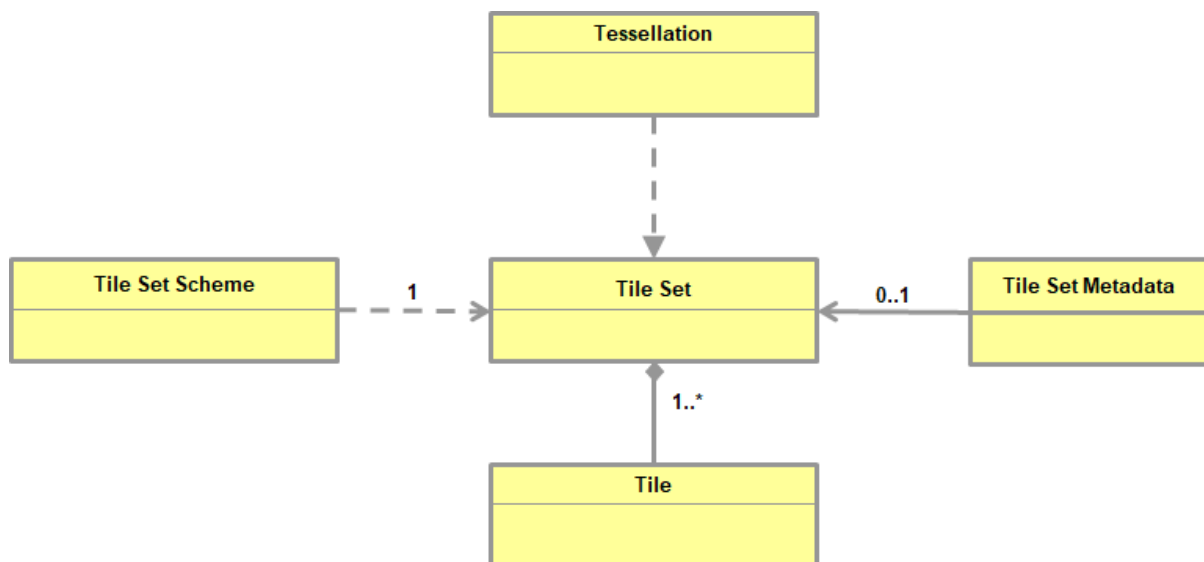


Figure 3. Conceptual for space partitioning based on tiles.

This figure captures the concepts for defining and implementing a tiled structure in any space. Each of the concepts is now briefly defined.

**tile:** The fundamental unit for partitioning space. A **tile** is a geometric shape with known properties that is the result of the tiling (tessellation) of the space defined by the tile schema. A tile consists of a single connected "piece" without "holes" or "lines" (topological disc). In two dimensional space, a tile could be regular (e.g. square) or irregular (e.g. Thiessen polygon). A voxel is an example of a regular partitioning of 3 dimensional space. Specific properties of a **tile**, such as width and height, are specified in a logical model

**tile set:** A set of tiles with common properties that meets the definition of the tile schema and tiled based on the tessellation rules. The common properties of a **tile set** are specified in the tile schema.

**tile set scheme:** A scheme that defines a set of common properties that define the **tile set**. The

schema could define the spatial reference system, the geometric properties of a tile, which space a uniquely identified tile occupies, and conversely which unique identifier corresponds to a space satisfying the geometric properties to be a tile. The properties required for a tiling approach that can be implemented are specified in the logical model.

**tessellation:** In addition to the definition provided in the Terms and Definition, the **tessellation** concept includes the rules for tessellating space into a **tileset**. The specific allowed/required **tessellation** approaches are specified in the logical model.

**tile set metadata:** In addition to the common properties that define a tile set, additional metadata may be provided. Such metadata could be an abstract, the owner, the author, or other common metadata. The logical model specifies which metadata elements are recommended.

# Chapter 8. Part 2: Logical Model for 2D Euclidean Space Tiles and Tiling: Normative

Part 2 of this Abstract Specification defines the mandatory and recommended requirements of the core model for the tessellation (tiling) of 2D Euclidean space. The specified mandatory elements *SHALL* be implemented regardless and independent of the implementation platform, programming languages, encodings, or any other implementation specific requirements.

The following figure, based on the Conceptual Model diagram, shows the properties by class (concept) and the relationships between the classes.

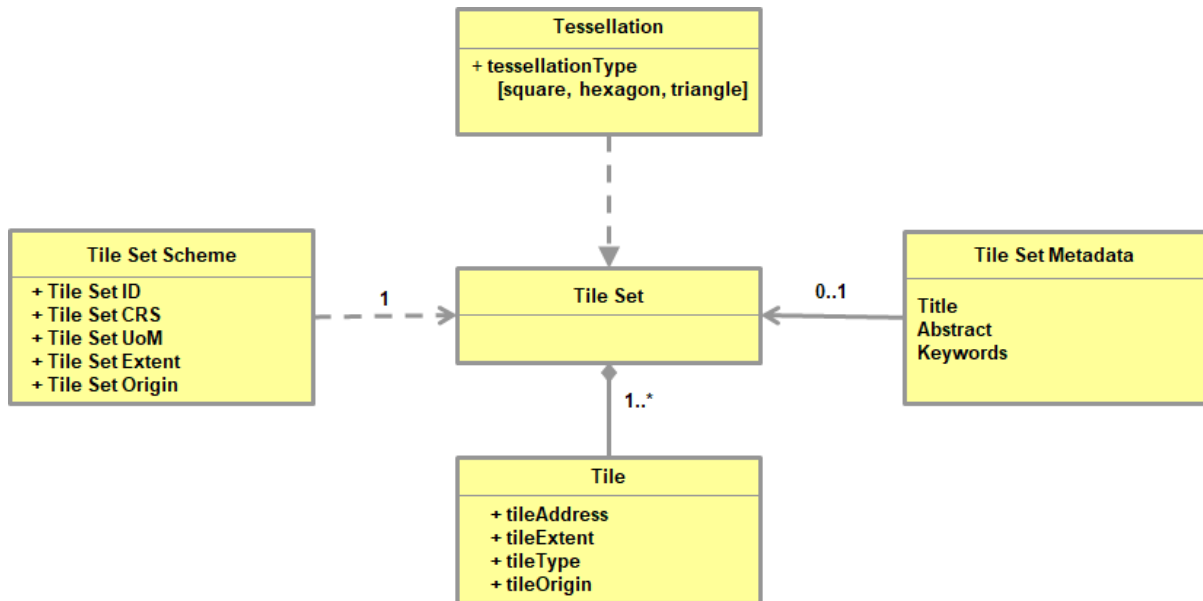


Figure 4. Logical Model for partitioning based on tiles in the 2D Euclidean case.

## 8.1. Restrictions to definitions of key concepts for the tiling of 2D plane

The general definitions for the key concepts are restricted as follows for the 2D use case. The key restrictions are highlighted.

**tile:** A geometric shape with known properties that is the result of the tiling (tessellation) of a 2D plane.

**tile set:** A set of tiles with common properties that meets the definition of the tile scheme and tiled based on the tessellation rules. **Tiles in a tile set are non-overlapping.**

**tessellation:** The tiles and associated tile set are generated by a **regular** (see Note below) tessellation of a 2D plane.

**tiling:** When tiling in **2D planar space**, the tiles are periodic. More specifically, a tiling consists of an arrangement of shapes covering the plane without overlap or gaps. Usually the tiling (tessellation) is restricted to a finite number of shapes and fixed sizes (although we can also consider tiles of geometrically decreasing and increasing sizes). The simplest and in some sense the

most natural tilings are periodic. In mathematical language, a pattern that repeats in a regular way is called [periodic](#). Periodic tiles consist of a meta-tiling of duplicated regions under translation. Three common periodic tilings used in GIS and clarified in the logical model for 2D tiling of Euclidean space are ones consisting of triangles, squares, or hexagons.

## 8.2. Base Requirement

### 8.2.1. Requirement 1 - Core

<b>Requirement 1</b>	/req/core
	A tiled data implementation SHALL include all mandatory requirements as specified in this Abstract Specification.

## 8.3. Requirement Class Tile Set Scheme

The following is the requirements class of the Tile Set Scheme

Element name: tileSetScheme

Requirements Class	
<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/tile-set">http://www.opengis.net/spec/2d-tile-model/1.0/req/tile-set</a>	
Target type	Token
Dependency	<a href="http://www.example.org/req/core">http://www.example.org/req/core</a>
<b>Requirement 2</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/id">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/id</a> Tile Set Identifier
<b>Requirement 3</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/crs">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/crs</a> Tile Set Coordinate Reference System
<b>Requirement 4</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/uom">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/uom</a> Tile Set Units of Measure
<b>Requirement 5</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/extent">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/extent</a> Tile Set Extent
<b>Requirement 6</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/origin">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/origin</a> Tile Set Origin
<b>Requirement 7</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/tessellation">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tileset/tessellation</a> Tile Set Tessellation Rules

This requirements class defines the requirements for defining a tile set.

### 8.3.1. Requirement 2 - Tile Set Identifier (ID)

<b>Requirement 2</b>	/req/core/tileset/id
	Each tile set in a tiled data store shall have a unique identifier.

Each tile set in a tiled data store shall have a unique identifier. This identifier could be a number or an alphanumeric string. For example, the ID could be CodeType, as an adaptation of MD\_Identifier class ISO 19115. An example of an alphanumeric identifier might be "Parcels" or "Street Network".

Element name: tileSetIdentifier

### 8.3.2. Requirement 3 - Tile Set Coordinate Reference System (CRS)

<b>Requirement 3</b>	/req/core/tileset/crs
	A tile scheme for a tile set shall have a coordinate reference system (CRS) that is consistent for all tiles in a given tile set.

A tile set scheme shall specify a coordinate reference system (CRS) that is the same for all tiles in a given tile set. The CRS for a given tile set could be based on an engineering datum or an earth centric datum. The CRS could be restricted to meet very specific requirements, say for visualization of pre-rendered tiles as defined in the WMTS or TMS standards. Further, at the core conceptual level, there is no requirement that all CRSs be the same for all tile sets in a tiled data store.

Having the same CRS for all tile sets in a data store is a requirement that would be specified in a profile. For example, CDB specifies WGS 84 (EPSG4326) for all tiles.

Element name: tileSetCRS

### 8.3.3. Requirement 4 - Tile Set Units of Measure (UoM)

<b>Requirement 4</b>	/req/core/tileset/uom
	A tile scheme for a tile set shall have a units of measure that is consistent for all tiles in a given tile set.

A tile set scheme shall specify a unit of measure (UoM) for a tile set, such as meters or feet, that is consistent for any given tile set. The UoM may be specified in the CRS definition or may be different from the CRS definition. At the core conceptual level, there is no requirement that all UoMs be the same for all tile sets in a tiled data store. Having a consistent UoM across all tile sets is a requirement that would be specified in a profile. (Example: "tileSetUoM=ft")

Element name: tileSetUoM

### 8.3.4. Requirement 5 - Tile Set Extent



<b>Requirement 5</b>	/req/core/tileset/extent  Each tile set SHALL have an extent in the TileSet coordinate reference system (CRS). The extent SHALL be expressed following the rules as specified in Clause 10.2.1 Basic bounding box parameters (OWS Common).
----------------------	---

In any tiling scheme, there are two extents to be considered. The extent of an individual tile and the extent of the tile set. This requirement is for defining the extent of the tile set.

Note: If there is one tile in a tile set, the extents are identical. In this tiling model there is no requirement, such as in DGGs, for a tile or tile set to cover the entire globe. A tile or tile set could have an extent that covers a building site, a city, a county, a country, and so forth. Any restrictions on the extent of a tile or tile set would be specified in one or more requirements in a profile or a profile with extensions of this abstract specification.

Element name: tileSetExtent

### 8.3.5. Requirement 6 - Tile Set Origin

<b>Requirement 6</b>	/req/core/tileset/origin  The tile set origin shall specify the spatial origin reference point for the entire tile set. Values shall be one of lower_left, upper_left, lower_right, upper_right, or center. The tile set origin also includes the coordinate of the tile set origin. The coordinate is expressed in the CRS of the tile set.
----------------------	--

The tile set origin defines where the spatial origin reference point is for the entire tile set. Typically, for a GIS implemented in the northern hemisphere, the tile set origin would be specified in the coordinates of CRS using the lower left hand corner. This Abstract Specification does not specify where the origin of a tile set is located. There is also not the requirement that all tile set origins in a tiled data store be the same (although having this rule sure helps downstream!). These rules would be specified in a profile or be specific to an implementation. For example, in WMTS the tile set origin is the upper left corner. In any case, there is a tile set origin.

Element name: tileSetOrigin

Example: tileSetOrigin=lower\_left

## 8.4. Tessellation class

Related to the Tile Set Scheme is the actual tessellation process used to generate the tile set. The following is the tessellation requirements class.

Unresolved directive in clause\_8\_2d\_logical\_model.adoc - include::requirements/REQ7\_core-

tessellation.adoc[]

A regular tessellation 2D planar surface can be performed using one of three geometric shapes: squares (rectangles), hexagons, and equilateral triangles. Squares (rectangles) represent the majority of current and past approaches for tessellating a 2D plane in geospatial systems. However, hexagons or triangles have been used in a variety of systems.

Typically there are also rules applied to specify the size and or extent for each tessellation type. For example, in a DGGS implementation, equal area hexagons are often used. In WMTS and TMS, each tile in a tile set are the same width and height.

Element name: tessellationType

### 8.4.1. Tessellation Type **square**

If the **tessellationType** is **square**, then the valid following properties *SHALL* be specified:

property name	description
tileWidth	Width of an individual tile. The default unit of measure is as specified by tileSetUoM,
tileHeight	Height of an individual tile. The default unit of measure is as specified by tileSetUoM.
equalArea	If all tiles are to be equal area this is the area in the unit of measure specified in tileSetUoM. If <b>equalArea</b> is specified then <b>width</b> and <b>height</b> are not required. If the desired unit of measure for the area is different than <b>tileSetUoM</b> , then the <b>unitofArea</b> property below needs to be specified.
<b>unitofArea</b>	Units of Measure for equal area tiles (e.g. square miles, hectares, etc)

### 8.4.2. Tessellation Type **triangle**

If the specified **tessellationType** is **triangle**, then the length of the side of the equilateral triangle *SHALL* be specified.

property name	description
length	Length of the triangle side in the units of measure specified in TileSetUOM. All three sides of the triangle have the same length.

property name	description
equalArea	If all tiles are to be equal area this is the area in the unit of measure specified in tileSetUoM. If equalArea is specified then length is not required. Length is calculated based on the properties of an equilateral triangle. If the desired unit of measure for the area is different from tileSetUoM, then the unitofArea property below needs to be specified.
unitofArea	Unit of Measure for equal area tiles (e.g. square miles, hectares, etc).

### 8.4.3. Tessellation Type Hexagon

In geometry, the hexagonal tiling or hexagonal tessellation is a regular tiling of the Euclidean plane, in which three hexagons meet at each vertex. For the purposes of this logical model, a tessellationType **hexagon** SHALL be a regular hexagon. A regular hexagon has vertices equally spaced around a circle and with all sides the same length. Further, the interior angle at each vertex is 120 degrees.

property name	description
length	Length of the hexagonal side in the units of measure specified in TileSetUOM. All six sides of the hexagon have the same length.
equalArea	If all hexagonal tiles are to be equal area, this property is the area in the unit of measure specified in TileSetUOM. If equalArea is specified then length is not required. If the desired unit of measure for the area is different from tileSetUoM, then the unit property below needs to be specified.
unitofArea	Unit of Measure for equal area tiles (e.g. square miles, hectares, etc).

## 8.5. Requirements Class: Tile Scheme

This clause specifies the requirements that define the properties of each individual tile in a tile set. Combined with the Tile Set Scheme common properties, there is enough information for any application - server or client - to create, access, process, analyze, and visualize any geospatial content provided in the tiled structure.

The Tile Scheme requirements class is defined as:

Requirements Class
<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/tile">http://www.opengis.net/spec/2d-tile-model/1.0/req/tile</a>

Target type	Token
Dependency	Tile Set Scheme
<b>Requirement 9</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tile/address">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tile/address</a> Tile Address
<b>Requirement 10</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tile/origin">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tile/origin</a> Tile Origin
<b>Requirement 11</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tile/reference">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tile/reference</a> Tile Reference
<b>Requirement 12</b>	<a href="http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tile/extent">http://www.opengis.net/spec/2d-tile-model/1.0/req/core/tile/extent</a> Tile Extent

### 8.5.1. Requirement 9 - Tile Address (ID)

<b>Requirement 9</b>	/req/core/tile/address A tile set <i>SHALL</i> use a spatial referencing method to assign a unique spatial reference (or index) to each tile across the entire tile set.
----------------------	---

A tile set shall use a spatial referencing method to assign a unique spatial reference (or index) to each tile across the entire tile set. These indices may be any alphanumeric string including controlled vocabularies such as the USGS quad index, IP address, tile row/column, or other indexing schemes. The default is sequential numbering beginning at the tile located at the tileSet origin. A value of *default* signifies row/column addressing.

Element name: tileAddress

### 8.5.2. Requirement 10 - Tile Origin

<b>Requirement 10</b>	/req/core/tile/origin The tile origin <i>SHALL</i> be defined. The same tile origin shall apply to all tiles in a given tile set. Tile origins may be one of "lower left", "upper left", "lower_right", "upper_right", and "centroid".
-----------------------	---

The tile origin defines where the spatial origin reference point is for each individual tile in a tile set. In a GIS being used in the northern hemisphere, this typically would be the lower left hand corner. All tile origin locations (such as lower left) would be the same for all tiles in a tile set. Neither the conceptual model nor the logical model specifies where the origin of a tile is located. These rules would be specified in a profile or be specific to an implementation. In any case, there is always a tile set origin.

Element name: tileOrigin

### 8.5.3. Requirement 11 - Tile Reference

*Requirement 11	/req/core/tile/reference Each tile <i>SHALL</i> be referenced at its centroid.
-----------------	---

Each tile must be referenced at its centroid. The centroid is the only location that will provide a systematic and consistent spatial reference point for all tiles in a tile set regardless of their shape. NOTE: The reference information does not need to be physically stored in the tile set store. The reference can be calculated from the tile extents (see requirement 12).

Element name: tileReference

### 8.5.4. Requirement 12 - Tile Extent

*Requirement 12	<p>/req/core/tile/reference</p> <p>Each tile <i>SHALL</i> have either an extent specified (bounding box) or <i>SHALL</i> have the ability for the tile extent to be calculated. Each tile extent <i>SHALL</i> be specified in the TileSet coordinate reference system (CRS).</p> <p>The encoding of the extent <i>SHALL</i> follow the rules as specified in Clause 10.2.1 Basic bounding box parameters (OWS Common). A value of 'default' means that the extent needs to be calculated.</p>
-----------------	---

Please note that the tile extent does not need to be stored as part of the tile metadata. However, at a minimum the tile extent must be able to be calculated. Tile extents are typically required for building a spatial index such as an R-Tree. See following Notes.

Element name: tileExtent

## 8.6. Recommended Tile Set Metadata Class

Providing more complete metadata for a tile set ensures enhanced provenance and quality of service for any content provided in a tiled structure.

The following suggested metadata elements are extracted from the OGC Tile Matrix Set candidate standard. Additional metadata elements may be added to this table. If additional metadata elements are required, the OGC CDB Standard provides an enumeration of the most common metadata standards used in the geospatial industry (Volume 1 Clause 5.1.6). The CDB Standard also provides a more detailed list of key metadata elements.

Element	Description	Reference	
title	Title of the tile set, normally used for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r9]	Zero or more (optional) Include when available and useful. Include one for each language represented.

Element	Description	Reference	
abstract	Brief narrative description of the tile set, normally available for display to a human	LanguageString data structure, see Figure 15 in OWS Common [OGC 06-121r9]	Zero or more (optional) Include when available and useful Include one for each language represented
keywords	Unordered list of one or more commonly used or formalized word(s) or phrase(s) used to describe this tile set.	MD_Keywords class in ISO 19115	Zero or more (optional) One for each keyword authority used

## 8.7. Notes on Tile Indexes, Tile Pyramids, and regular tessellations

### 8.7.1. Tile indexes

A tile index provides an efficient (fast) mechanism for finding any tile or set of tiles in a tiled data store when using a spatial filter, including a point and query operation. Typical spatial indexing are R-Trees and Quad-Trees. The logical model specifies no requirements for how a spatial index is constructed or the technology used. However, the tile extents, tile addresses, and tile references can be used to populate a spatial index, such as an R-Tree.

### 8.7.2. Tile Pyramids

A tile pyramid structure would be specified as a set of rules (requirements) applied using the core logical model. A tile pyramid is usually based on using a square tile tessellation for some area of interest and a common tile set extent for all tile sets that define the pyramid. The base of the pyramid could be one tile whose extent is identical to that of the tile set. The next tile set (level in the pyramid) could divide the tile set extent into 4 tiles. The next tile set could be based on dividing the tile set extent into 16 tiles and so forth. The OGC Tile Matrix Set candidate standard covers this topic extensively.

### 8.7.3. Regular Tessellations

Note: A tiling is said to be regular if the [symmetry group](#) of the tiling acts transitively on the flags of the tiling, where a flag is a triple consisting of a mutually incident vertex, edge and tile of the tiling. This means that, for every pair of flags, there is a symmetry operation mapping the first flag to the second. This is equivalent to the tiling being an edge-to-edge tiling by congruent regular polygons. There must be six equilateral triangles, four squares or three regular hexagons at a vertex, yielding the three regular tessellations. Grünbaum, Branko; Shephard, Geoffrey C. (1977). "Tilings by regular polygons". *Math. Mag.* 50 (5): 227–247. doi:10.2307/2689529

# Annex B A: Conformance Class Abstract Test Suite (Normative)

## A.1. Conformance Class A

Core

### A.1.1. Requirement 1 - Core statement

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ1_core.adoc
<b>Requirement:</b>	/req/req-class-a/core
<b>Test purpose:</b>	To verify that a tiling specification conforms to the tiling logical core model for the 2D Euclidean plane use case.
<b>Test method:</b>	Inspect documentation.

## A.2. Conformance Class Tile Set Schema

Based on the Requirement Class Tile Set Schema

requirements/rc-tileset-schema.adoc

### A.2.1. Requirement 2

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ2_core-tileset-id.adoc
<b>Requirement:</b>	/req/core/tileset/id
<b>Test purpose:</b>	To verify that a tile set has a unique identifier.
<b>Test method:</b>	Either automated test or visual inspection.

### A.2.2. Requirement 3

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ3_core-tileset-crs.adoc
<b>Requirement:</b>	/req/core/tileset/crs
<b>Test purpose:</b>	To verify that a tile set has a unique CRS identifier.
<b>Test method:</b>	Either automated test or visual inspection.

### A.2.3. Requirement 4

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ4_core-tileset-uom.adoc
<b>Requirement:</b>	/req/core/tileset/uom
<b>Test purpose:</b>	To verify that a tile set has a unique UoM identifier.
<b>Test method:</b>	Either automated test or visual inspection.

### A.2.4. Requirement 5

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ5_core-tileset-extent.adoc
<b>Requirement:</b>	/req/core/tileset/extent
<b>Test purpose:</b>	To verify that a tile set has an extent specified in the tile set CRS and following the OWS COMmon requirements for expressing a bbox.
<b>Test method:</b>	Either automated test or visual inspection.

### A.2.5. Requirement 6

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ6_core-tileset-origin.adoc
<b>Requirement:</b>	/req/core/tileset/origin
<b>Test purpose:</b>	To verify that origin reference point for the entire tile set is specified.
<b>Test method:</b>	Either automated test or visual inspection.

### A.2.6. Requirement 7

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ7_core-tessellation.adoc
<b>Requirement:</b>	/req/core/tileset/tessellation
<b>Test purpose:</b>	To verify that a tessellation type of 'square', 'hexagon', or 'triangle' is specified.
<b>Test method:</b>	Either automated test or visual inspection.

## A.3. Conformance Class Tile

Requirement Class Tile Schema

requirements/rc-tile-schema.adoc

### A.3.1. Requirement 9

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ9_core-tile-address.adoc
<b>Requirement:</b>	/req/core/tile/address
<b>Test purpose:</b>	To verify that a spatial referencing method to assign a unique spatial reference (or index) to each tile across the entire tile set as been specified.
<b>Test method:</b>	Either automated test or visual inspection.

### A.3.2. Requirement 10

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ10_core-tile-origin.adoc
<b>Requirement:</b>	/req/core/tile/origin
<b>Test purpose:</b>	To verify that a spatial origin reference point is for each individual tile in a tile set is specified.



<b>Test method:</b>	Either automated test or visual inspection.
---------------------	---

### A.3.3. Requirement 12

<b>Test id:</b>	/conf/conf-class-a/requirements/REQ12_core-tile-extent.adoc
<b>Requirement:</b>	/req/core/tile/extent
<b>Test purpose:</b>	To verify that the extent of the tile is provided or can be calculated.
<b>Test method:</b>	Either automated test or visual inspection.

# Additional Informative Material: Simple Example of an XML encoding

Below is a modification of an XML example from the WMTS standard but revised to meet the above example requirements. This could just have easily been written in JSON, Javascript, and so on.

```
<ows:Identifier>Fort Collins Tiled Data Store</ows:Identifier>

<tileSet>
  <tileSetIdentifier>Parcels</tileSetIdentifier>
  <tileSetCRS>EPSG2876</tileSetCRS>
  <tileSetUOM>FT</tileSetUOM>
  <tileSetOrigin>lower_left</tileSetOrigin>
  <!-- Extent defined. lower left point of tile set bounding box -->
  <ows:BoundingBox>
    <LowerCorner>3080000,1400000</LowerCorner>
    <UpperCorner>3600000,1500000</UpperCorner>
  </ows:BoundingBox>

  < tessellationType>Square</tessellationType>
  <!-- width and height of each tile in specified units -->
  <tileWidth>1000</tileWidth>
  <tileHeight>1000</tileHeight>
  <tileOrigin>lower_left</tileOrigin>
  <tileAddress>default</tileAddress>
  <tileExtent>default</tileExtent>
</tileSet>
```

## Annex B: Revision History

<b>Date</b>	<b>Release</b>	<b>Editor</b>	<b>Primary clauses modified</b>	<b>Description</b>
2019-April-28	1.0	C. Reed Editor	all	initial version
2019-June-3	1.0	C. Reed		Various

# Annex C: Bibliography

[1] Web: Gupta, Sanjay. Difference between Conceptual, Logical and Physical Data Models.  
<http://uksanjay.blogspot.com/2012/06/difference-between-conceptual-logical.html>

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