

Open Geospatial Consortium

Submission Date: <yyyy-mm-dd>

Approval Date: <yyyy-mm-dd>

Publication Date: <yyyy-mm-dd>

External identifier of this OGC® document: <http://www.opengis.net/doc/IS/cdb-geopackage/1.2>

Internal reference number of this OGC® document: 19-054

Version: 1.2

Category: OGC® Draft Implementation Standard

Editor: Carl Reed, others

Optional CDB GeoPackage Extension

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® Standard

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 optional OGC CDB extension defines the requirements and provides CDB specific guidance on using GeoPackage containers in a CDB data store. There is a companion CDB Best Practice document that provide rules and guidance for transforming CDB structured Shapefiles into CDB structure GeoPackages that are compliant with the requirements and conformance classes as defined in this document.

ii. Keywords

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

ogcdoc, OGC document, CDB, GeoPackage

iii. Preface

Background for this optional CDB Extension

The original requirement for this optional CDB extension was documented in [OGC Change Request 545](#). This OGC change request was submitted based on work performed in [OGC Testbed 13](#). The testbed activity and related change request captured a broad community requirement for being able to use GeoPackage containers in a CDB data store. At the same time, an additional requirement was identified to test and identify best practices for moving CDB vector files stored as Shapefiles into one or more GeoPackages.

In 2019, the CDB SWG executed the [CDB Vector Data in GeoPackage Interoperability Experiment \(IE\)](#). The participants in this IE tested transforming CDB Shapefile vector data into one or more GeoPackage(s) and storing the result in a CDB data store. GeoPackage Version 1.2 and CDB Version 1.1 and related Best Practices were the standards baseline used for this experiment. The IE built on the work described in the [OGC CDB, Leveraging the GeoPackage Discussion Paper](#). A primary objective of the IE was to agree and document possible change requests and/or best practices for storing vector data in a CDB data using encodings and/or containers other than Shapefiles.

OGC Patent Declaration

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):

Organization name(s)

v. Submitters

- Carl Reed & Associates
- FlightSafety Visual Systems
- CAE Inc.

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

Name	Affiliation
Carl Reed (<i>editor</i>)	Carl Reed & Associates
Ryan Franz	FlightSafety Visual Systems
David Graham	CAE Inc.

===

== Scope

This optional OGC CDB Extension defines the behavior and requirements for encoding vector data in a GeoPackage container for use in a CDB data store. The requirements and related guidance are grounded in the existing CDB Core requirements and the GeoPackage core requirements for vector data. As such, any GeoPackage that is to be referenced/used in a CDB data store must be 1.) compliant with the CDB core requirements for vector data and 2.) compliant with the GeoPackage core requirements for encoding vector data. Please note that some of the core GeoPackage requirements are profiled in order to be consistent with the CDB core requirements as specified in [Volume 1: OGC CDB Core Standard: Model and Physical Data Store Structure](#). Further, there are associated normative additions to the CDB Core to accommodate not just the use of GeoPackages but also other vector storage encodings/containers.

Before reading this standard, please remember that the idea is to restrict the encoding of a dataset to a single vector format per CDB Version. Since a “CDB” is made of one or more “Versions” (as specified by Configuration.xml), and that each CDB Version can have a different encoding for a given dataset, the result is that a “CDB” may pretty well exist with multiple encodings for the same dataset. This means that if you wish to use GeoPackage containers, you need to create a version that will just contain GeoPackages of the vector data and not include any Shapefiles.

== Conformance

This standard defines [\[TBD\]](#) requirements / conformance classes.

The standardization targets of all conformance classes are "GeoPackages in CDB Data Store".

===

GeoPackage Geometry Type	Shapefile Vector Type
Point	Point
Linestring	Polyline
Polygon	Polygon
MultiPoint	MultiPoint
Point with "z" column set	PointZ
Linestring with "z" column set	PolylineZ
Polygon with "z" column set	PolygonZ
Linestring with "m" column set	PolylineM
Polygon with "M" column set	PolygonM
MultiPoint with "m" column set	MultiPointM

===

=== Identified Use Cases While previous work focused on the process and requirements for transforming CDB structured vector Shapefiles into GeoPackages, the identified use cases are apropos when defining how to store and access any other vector/geometry content regardless of encoding or format. However, vector data encodings and containers developed outside a CDB data transformation process would also require definition of additional requirements. For example, a GeoJSON encoding of feature data would require a transformation to "break" the GeoJSON file into the appropriate tile and LoD structure in order to be consistent with the CDB data store and structure requirements. The same would be true of vector data provided as none CDB structured Shapefiles, CityGML encodings, Oracle Spatial structured data, Esri GeoDataBase files and so on.

Both the OGC OGC CDB - Leveraging GeoPackage Discussion Paper and the OGC CDB Vector Data in GeoPackage Interoperability Experiment Engineering Report focused on a constrained set of well defined use cases related to the current CDB structure and how to best use GeoPackage for resolving each of the use cases. As such, dealing with non-CDB structured vector content was not addressed.

The four initial use cases identified in the OGC Discussion Paper are:

. Replace each Shapefile with a GeoPackage: The easiest way to integrate a GeoPackage container into a CDB data store is to replace each Shapefile in a CDB data store with a GeoPackage. . Make each CDB tile a layer in a single GeoPackage: Constructing each vector tile within CDB as a table within a GeoPackage for a given CDB dataset is a straightforward approach to utilize GeoPackage capabilities and significantly reduce file counts in a CDB (note that in GeoPackage a table is known as a layer). . Store each CDB LOD as a layer in GeoPackage: Design approach #3 incorporates the lessons learn from

===

<p>Requirement 1 Vector Format by Version</p>	<p>http://www.opengis.net/spec/cdb/1.0/geopackage/cdb-core</p> <p>Any version in a CDB data store <i>SHALL</i> contain one and only one vector data format.</p>
<p>===</p> <p>There is a related CDB recommendation that a feature should not have two representations.</p> <p><Insert recommendation include here. Page 230></p> <p>=== Requirement on geometry types and feature instances The following requirement clarifies the geometry types allowed for instances of feature data referenced to a tile. Essentially, all instances of a specific feature type (code?) SHALL be of the same geometry type.</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>
<p>Requirement 2 Vector GeoPackage Geometry Type</p>	<p>http://www.opengis.net/spec/cdb/1.2/geopackage/vector-geom-rule</p> <p>All instances of a given feature code <i>SHALL</i> be of the same geometry type. While the GeoPackage model supports encoding of 8 different types that can be stored in the same GeoPackage, the CDB standard requires a maximum of one geometry type for point features, a maximum of one geometry type for lineal features and a maximum of one geometry type for polygon features for each tile (for a maximum of 3 feature geometry types per tile).</p>

<p>===</p> <p>=== GeoPackage Requirements This section defines the GeoPackage requirements that need to be considered when implementing a CDB compliant GeoPackage for use in a CDB data store. This section also provides any clarifications and/or profiles of those requirements.</p> <p>==== GeoPackage Requirements - core The following requirement captures all of the core GeoPackage requirements that need to be implemented in order to be a fully compliant GeoPackage for use in a CDB data store. Please note that GeoPackage Requirements 10 and 11 on Coordinate Reference Systems has been profiled to be consistent with the mandatory requirements in the CDB standard.</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>
<p>Requirement 3 GeoPackage Core</p>	<p>http://www.opengis.net/spec/cdb/1.0/geopackage/geopackage-core</p> <p>Any CDB structured GeoPackage <i>SHALL</i> be compliant with GeoPackage Requirements 1 through 16 inclusive. Please see Requirement 4 of this standard for additional clarification on GeoPackage Requirements 10 and 11 - Spatial Reference Systems (aka coordinate reference systems in CDB).</p>

<p>===</p> <p>==== GeoPackage Requirements constrained by CDB - CRS Profile The following requirement clarifies (profiles) GeoPackage requirements 10 and 11 for specifying a Coordinate Reference System (CRS). The GeoPackage standard uses the term Spatial Reference System instead of Coordinate Reference System. This is because GeoPackage standard was designed to be flexible and to be able to accommodate any reference system. As such, GeoPackage tables that deal with CRS us the "spatial_ref" literal as part of the table name.</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>
<p>Requirement 4 CDB - GeoPackage Core CRS</p>	<p>http://www.opengis.net/spec/cdb/1.0/geopackage/cdb-geopackage-core-crs</p> <p>As per CDB Requirement 8, any CDB structured GeoPackage <i>SHALL</i> specify geographic locations using WGS-84 (World Geodetic System 1984), equivalent to EPSG (European Petroleum Survey Group) code 4326 (2 dimensions) or EPSG code 4979 (3 dimensions). If a geographic location also has an altitude, the altitude <i>SHALL</i> be expressed relative to the WGS-84 reference ellipsoid.</p>
<p>===</p> <p>Please also note the definition of Volume 1 CDB Core Requirement 7 Units of Measure and Requirement 111 Vextex CRS for guidance on proper and compliant use of CRS in a CDB compliant GeoPackage.</p> <p>==== GeoPackage Requirements Options - Features The following requirement clarifies the GeoPackage requirements that shall be followed when implementing vector fectures in a GeoPackage. Please note that GeoPackage geometry and feature definitions are consistent with the OGC Simple Features model and ISO 19107.</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>

<p>Requirement 5 GeoPackage Vector Features</p>	<p>http://www.opengis.net/spec/cdb/1.0/geopackage/geopackage-features</p> <p>Any CDB structured GeoPackage that encodes features <i>SHALL</i> be compliant with GeoPackage Requirements 18 through 33 inclusive and GeoPackage Requirements 146 and 150. Please see Requirement 20 of the GeoPackage standard for additional clarification on vector feature geometry types. These requirements are included in Clause 2.1 Features of the GeoPackage Standard.</p>
<p>===</p> <p>==== Requirement CDB Core - Need to discuss this one. Could be changed to a backend process and not a client. The following requirement defines how vector data in a GeoPackage is to be structured based on requirements specified in the CDB Core document.</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>
<p>Requirement 6 GeoPackage polygon readers</p>	<p>http://www.opengis.net/spec/cdb/1.0/geopackage/polygon-rules-reader</p> <p>NOTE: This requirement needs to be discussed. Perhaps should be a data prep rule. Although the above are guidelines, GeoPackage readers <i>SHALL</i> handle the following cases with proper error handling and reporting for Polygon geometries:</p> <ul style="list-style-type: none"> * Has no self-intersections or co-linear segments * Has no identical consecutive points (no zero-length segments) * Does not degenerate into zero-area parts * Does not have clock-wise inner rings (“Dirty Polygon”)

<p>===</p> <p>=== CDB Requirements for a CDB compliant GeoPackage Vector encoding This section documents the CDB requirements that need to be considered when implementing a CDB compliant GeoPackage for use in a CDB data store. This section also provides any clarifications and/or profiles of those requirements. The majority of the requirements referenced in this section are detailed in Volume 1: OGC CDB Core Standard: Model and Physical Data Store Structure (hereafter known as CDB Core). In the GeoPackage in CDB standard, requirements are specified that themselves reference specific requirements in the Core</p> <p>==== CDB Vector Data Core Requirements The following requirement profiles the CDB general data representation core requirements class: Requirements Class - General Data Representation (Requirements 6-10). Please note that the profile excludes the raster imagery compression requirement. The related CDB conformance class documented in Annex A of the CDB Core standard is A.1.1 General CDB Data Store and Implementation.</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>
<p>Requirement 7 CDB general data requirements for GeoPackage: Profile</p>	<p>http://www.opengis.net/spec/cdb/1.0/geopackage/cdb-geopackage-data</p> <p>Any CDB structured GeoPackage that encodes vector features <i>SHALL</i> be compliant with CDB Requirements 7 through 10. These requirements are documented in CDB Requirements Class General Data Representation Requirements. Please note that Requirement 6 image compression is excluded.</p>

<p>===</p> <p>==== CDB Tiling and LoD Rules for Structuring GeoPackages The following requirement references the CDB requirements for the rules for tiling the vector data in a GeoPackage and determination of the level of detail (LoD) in which the GeoPackage should be stored. Complete details for the CDB tiling and LoD storage model are found in CDB Volume 1 and clause "CDB Concepts".</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>
<p>Requirement 8 CDB Tiles/Geocells and LoD relationships for GeoPackage</p>	<p>http://www.opengis.net/spec/cdb/1.2/geopackage/cdb-geopackage-tiles-and-lods</p> <p>Any CDB structured GeoPackage that encodes vector features <i>SHALL</i> be compliant with CDB Tiles/Geocells and LoD relationships 11 through 16 and 41 inclusive These requirements are documented in CDB Requirements Class Tiles/Geocells and LoD relationships (11-16 and 41).</p>
<p>===</p> <p>==== GeoPackage in CDB - Tiled Vector Data Sets Clause 3.6 - Tiled Data Sets specifies the requirements and provides detailed supporting information on the tiling structure and LoD structure for tiled vector data. Note: All vector data sets are tiled in a CDB structured data store. The level-of-detail organization for GeoPackage vector datasets mimics the concept of map scaling commonly found in cartography. Please especially note Requirement 64 in the Core. Also pay careful attention to the tables and equations for calculating the latitude and longitude elements of the file names for the GeoPackages.</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>

Requirement 9 vector tiled data	http://www.opengis.net/spec/cdb/1.2/geopackage/cdb-core-vector-tiled-data Any CDB structured GeoPackage that encodes vector features SHALL be compliant with CDB Core Tiled Data Requirements 64 through 67 inclusive These requirements are documented in the CDB Core Requirements Class Tiled Datasets (64-67).
----------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

===

==== CDB GeoPackage Vector Datasets

===== CDB Vector data general requirements

This clause references requirements in section 5.7 Vector Datasets in the Volume 1 CDB Core Model Standard. The OGC suggests that clause 5.7 of the Core CDB standard be read in its entirety prior to developing capabilities to transform any non-CDB data source into a CDB structured GeoPackage that meets the mandatory requirements and would pass the CDB compliance tests.

From Volume 1: All of the information that is needed to instance features is organized in accordance to the CDB tile structure. All the tiled Vector dataset files are located in the same directory. The dataset's second component selector (CS2) is used to differentiate between files with the same extension or with the same Vector features.

Further, the developer also needs to understand the feature coding system used in a CDB data store. From Volume 1: *The Vector dataset concept and the feature concepts overlap somewhat; some of the Vector datasets are generalizations or specializations of feature codes. Section 1.5 CDB Data Dictionary provides a recommended mapping of the feature attributes across the CDB compliant datasets. Note that the same feature should not have two representations.* More specifically, visit section 3.3.8.1 Feature Classification for more details. Basically, for the file path and naming requirements to be properly implemented, the correct feature codes need to be used. See http://schemas.opengis.net/cdb/1.1/Feature_Data_Dictionary.xml and http://schemas.opengis.net/cdb/1.1/Feature_Data_Dictionary.xsd for a complete list of feature codes. The .xml file is actually located in http://schemas.opengis.net/cdb/cdb-1_1_0.zip .

The first set of tiled vector data requirements relates to coordinates for lights and the structure of vertex coordinates.

===

<p>Requirement 10 Tiled Vector Datasets</p>	<p>http://www.opengis.net/spec/cdb/1.2/geopackage/cdb-core-tiled-vector-datasets</p> <p>Any CDB structured GeoPackage that encodes vector features SHALL be compliant with CDB Core Tiled Vector Datasets Requirements 107 through 111 inclusive. These requirements are documented in the CDB Core Requirements Class Tiled Vector Datasets (107-111).</p>
<p>===</p> <p>===== CDB Attribution Attributes are used to describe one or more real or virtual characteristics of a feature. Features can be assigned a variable number of attributes. The following requirements from the core document and related informative discussion in Volume 1 Core describe the CDB compliant usage of attributes in a CDB datastore.</p> <p>[width="90%",cols="2,6"]</p>	<p>===</p>
<p>Requirement 11 Tiled Vector Datasets Attribution</p>	<p>http://www.opengis.net/spec/cdb/1.2/geopackage/cdb-core-tiled-vector-datasets-attribution</p> <p>Any CDB structured GeoPackage that encodes vector features SHALL be compliant with CDB Core Tiled Vector Datasets Requirements 112 through 116 inclusive. These requirements are documented in the CDB Core Requirements Class 5.7.1.2 CDB Attribution (112-116).</p>
<p>===</p> <p><<< //// add or remove annexes after "A" as necessary //// [appendix] :appendix-caption: Annex == Conformance Class Abstract Test Suite (Normative)</p> <p>[NOTE] Ensure that there is a conformance class for each requirements class and a test for each requirement (identified by requirement name and number)</p> <p>=== Conformance Class A</p> <p>===== Requirement 1</p> <p>[cols=">20h,<80d",width="100%"]</p>	<p>===</p>

Test id:	/conf/conf-class-a/req-name-1
Requirement:	/req/req-class-a/req-name-1
Test purpose:	Verify that...
Test method:	Inspect...
=== ==== Requirement 2 <<< [appendix] :appendix-caption: Annex == GeoPackage and Shapefile Geometry types cross walk. [cols=",,,",options="header",]	===
Value	GeoPackage geometry type
Shapefile Vector type	Fields
0	
Null shape	None
1	Point
Point	X, Y
3	Linestring
Polyline	MBR, Number of parts, Number of points, Parts, Points
5	Polygon
Polygon	MBR, Number of parts, Number of points, Parts, Points
8	MultiPoint
MultiPoint	MBR, Number of points, Points
11	Point
PointZ	X, Y, Z <i>Optional</i> : M
13	Linestring
PolylineZ	<i>Mandatory</i> : MBR, Number of parts, Number of points, Parts, Points, Z range, Z array <i>Optional</i> : M range, M array
15	Polygon

PolygonZ	<i>Mandatory:</i> MBR, Number of parts, Number of points, Parts, Points, Z range, Z array <i>Optional:</i> M range, M array
18	Multipoint
MultiPointZ	<i>Mandatory:</i> MBR, Number of points, Points, Z range, Z array <i>Optional:</i> M range, M array
21	Point
PointM	X, Y, M
23	Linestring
PolylineM	<i>Mandatory:</i> MBR, Number of parts, Number of points, Parts, Points <i>Optional:</i> M range, M array
25	Polygon
PolygonM	<i>Mandatory:</i> MBR, Number of parts, Number of points, Parts, Points <i>Optional:</i> M range, M array
28	MultiPoint
MultiPointM	<i>Mandatory:</i> MBR, Number of points, Points <i>Optional Fields:</i> M range, M array
31	Surface
MultiPatch	<i>Mandatory:</i> MBR, Number of parts, Number of points, Parts, Part types, Points, Z range, Z array <i>Optional:</i> M range, M array
=== <<< //// Revision History should be the last annex before the Bibliography Bibliography should be the last annex //// [appendix] :appendix-caption: Annex == Revision History [width="90%",options="header"]	===
Date	Release
Editor	Primary clauses modified
Description	2019-11-06
1.2	C. Reed
all	initial version