

Testbed-12 GeoPackage US Topo Engineering Report

Table of Contents

1. Introduction	6
1.1. Scope	6
1.2. Document contributor contact points	6
1.3. Future Work	6
1.4. Foreword	6
2. References	7
3. Terms and definitions	8
3.1. GeoPackager	8
3.2. raster tile	8
3.3. symbology	8
3.4. style	8
3.5. filter	8
3.6. OGC Web Processing Service (WPS).....	8
3.7. Stylesheet	9
4. Conventions	10
4.1. Abbreviated terms	10
4.2. UML notation	10
5. Overview	11
5.1. Requirements	11
5.2. Use Cases	11
5.3. Approaches	11
5.3.1. Feature input source translation	12
5.3.2. Raster input source translation	12
5.3.3. Styling approach.....	12
5.3.4. Style and symbology encoding	12
5.3.5. Raster tile encoding	12
5.3.6. Integration Use Case experiences.....	13
5.3.7. Integration obstacles	13
5.3.8. Encoding Obstacles	13
5.3.9. Process Obstacles	14
5.3.10. Interoperability Experiment Outcomes	14
5.4. Proposed style encoding extension	14
5.5. Proposed hierarchical layer/raster extension.....	14
6. Status Quo & New Requirements Statement	15
6.1. Status Quo	15
6.2. Requirements Statement	15
7. Solutions	16
7.1. Targeted Solutions	16

7.1.1. Converter WPS	16
7.1.2. Resulting GeoPackage	18
7.1.3. Style Encoding	20
7.2. Alternatives considered	21
7.3. Recommendations	21
7.4. Styling Examples	22
Appendix A: Data Mappings from GDB to SQLite	25
A.1. Naming and Organizing Layers Created from the GeoDatabase Source	25
Appendix B: Symbology Encoding	29
B.1. History	30
B.2. Examples	30
B.2.1. Contour Styling	31
B.2.2. Examples of symbology generated for Layers found in VECTOR_3330_Benicia_7_5_Min.gdb	32
B.2.3. BureauofLandManagement	32
B.2.4. CellGrid_7_5Minute	32
B.2.5. Elev_Contour	33
B.2.6. FishandWildlifeService	33
B.2.7. GagingStation	33
B.2.8. GU_NativeAmericanArea	34
B.2.9. IntermittentNHDArea	34
B.2.10. IntermittentNHDWaterbody	35
B.2.11. LANDCOVER_WOODLAND	35
B.2.12. NHDArea	35
B.2.13. NHDFlowline	36
B.2.14. NHDLine	37
B.2.15. NHDPoint	38
B.2.16. NHDWaterbody	39
B.2.17. Struct_Point	39
B.2.18. TNMDerivedNames	40
B.2.19. Trans_RailFeature	40
B.2.20. Trans_RoadSegment	40
B.2.21. trans_roadsegment_ntdnofs	42
B.2.22. Trans_TrailSegment	43
B.2.23. Wetlands	43
Appendix C: UML Models for Symbology Extension	44
Appendix D: Converter WPS Description	52
D.1. Potential CR	52
D.2. High Level Map of Input/Output	52
D.3. GetCapabilities	53
D.4. Datasource operations	54

D.4.1. CreateDatasource (GDB).....	54
D.4.2. CreateDatasource (WMTS).....	54
D.4.3. UpdateDatasource	55
D.4.4. DeleteDatasource.....	56
D.5. GeoPackage operations	57
D.5.1. CreateGeoPackage	57
D.5.2. Transcript	57
Appendix E: Revision History	74

Publication Date: 2017-05-12

Approval Date: 2017-01-26

Posted Date: 2016-11-02

Reference number of this document: OGC 16-037

Reference URL for this document: <http://www.opengis.net/doc/PER/t12-B004>

Category: Public Engineering Report

Editor: Robert Cass

Title: Testbed-12 GeoPackage US Topo Engineering Report

Testbed-12 GeoPackage US Topo ER (16-037)

COPYRIGHT

Copyright © 2017 Open Geospatial Consortium. To obtain additional rights of use, visit <http://www.opengeospatial.org/>

WARNING

This document is an OGC Public Engineering Report created as a deliverable of an initiative from the OGC Innovation Program (formerly OGC Interoperability Program). It is not an OGC standard and 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, any OGC Engineering Report should not be referenced as required or mandatory technology in procurements. However, the discussions in this document could very well lead to the definition of an OGC Standard.

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.

Abstract

This OGC Engineering Report documents the outcome of the US Topo experiment. The focus of the US Topo experiment was to generate GeoPackages by combining USGS Topo Map Vector Data Products [1: <http://viewer.nationalmap.gov/basic/?basemap=b1&category=vectorcmb&q=&zom=4&bbox=-129.46289063,10.22843727,-64.42382813,58.07787627&preview=&avail=Topo%20Map%20Vector%20Data&refpoly=>]; and the Topo TNM Style Template [2: <https://viewer.nationalmap.gov/tools/topotemplate/>]. The output GeoPackages will contain both features and instructions for styling these features as well as orthoimagery, shaded relief raster tilesets, national wetlands raster tilesets and elevation data derived from USGS provide 1/9 arc second elevation imagery. The process used to generate the GeoPackage is explained. Problems and obstacles encountered decoding the source product and styles and converting these artifacts to a GeoPackage are explained with recommendations for improvements. Additionally, the experience applying the generated GeoPackage in two use cases proposed for this testbed will be evaluated. The introduction of symbolization for vector features will be articulated as a proposed extension for GeoPackage. Any issues related to encoding the TNM style template using the extension are documented.

Business Value

GeoPackage is an open standard with an established extension mechanism. Using an extension to express a method of styling features will test the extension mechanism and advance the work of the GeoPackage SWG regarding the inclusion of at least one form of styling information in a GeoPackage.

Technology Value

This ER will provide a useful perspective for the GeoPackage SWG membership as they develop methodologies for encoding styling information and feature collections. This ER may also be relevant to efforts of working groups that advance the definition of symbology and approaches to styling that were also encountered in Testbed-12.

How does this ER relate to the work of the Working Group

This ER is relevant to the GeoPackage SWG because it addresses the need for symbolizing features in a GeoPackage. Methods for storing styles and evaluating styles are presented in this ER that will inform the pursuit of style encoding standards for GeoPackage.

Keywords

ogcdocs, testbed-12, topographic, GeoPackage, styles, styling, vectors, symbology, portrayal

Proposed OGC Working Group for Review and Approval

GeoPackage SWG

Chapter 1. Introduction

1.1. Scope

This OGC Engineering Report describes the processes and constructs used and challenges encountered while encoding USGS topographic feature data, shaded relief and orthoimagery tiled data in GeoPackage format. This Engineering Report will also detail the encoding schema for the data and styles used. Additionally, this Engineering Report will describe prototype GeoPackage extensions for storing styles and hierarchical feature classes. This OGC® document is applicable to the work of the GeoPackage SWG as a means of advancing the development of a method of encoding styles for vector features to meet Use Case 5 from OGC 15-122r1. Additionally, this document is applicable to readers interested in encoding datasets and styles like the ones addressed in the document.

1.2. Document contributor contact points

All questions regarding this document should be directed to the editor or the contributors:

Table 1. Contacts

Name	Organization
Robert Cass	Compusult Ltd.

1.3. Future Work

It is expected that this document may result in changes in other documents. Additionally, this document provides recommendations for improved approaches to encoding the required data sources in a GeoPackage that bring the produced geopackage closer in alignment to the pdf maps that the USGS currently produces.

1.4. Foreword

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.

Chapter 2. References

The following documents are referenced in 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.

- OGC® WPS 2.0 Interface Standard, 2015-10-05, <http://docs.opengeospatial.org/is/14-065/14-065.html>
- OGC® GeoPackage Encoding Standard, 2016, <http://www.geopackage.org/spec/>
- OGC® Web Map Service, 2006-03-15, http://portal.opengeospatial.org/files/?artifact_id=14416
- OGC® Web Map Tile Service Implementation Standard, 2010-04-06, http://portal.opengeospatial.org/files/?artifact_id=35326
- GDAL Geospatial Data Abstraction Library, 2016, <http://www.gdal.org>
- SpatialLite, 2016, <https://www.gaia-gis.it/fossil/libspatialite/home>

Chapter 3. Terms and definitions

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

3.1. GeoPackager

computer software that uses input data sources to create a GeoPackage containing an encoding of those input sources

3.2. raster tile

rectangular piece of a larger tile mosaic or matrix intended to be viewed between certain map scales that is encoded in a raster format

3.3. symbology

electronic encoding of all the information required to correctly visually portray a geographic feature

3.4. style

electronic encoding of all the symbologies used to visually portray geographic features that also maps these symbologies to features

3.5. filter

boolean clause used to evaluate the data attributes of a feature to determine what symbology will be used to portray this feature

3.6. OGC Web Processing Service (WPS)

The term OGC Web Processing Service (WPS) describes a service instance of an OGC WPS server. A WPS is a container for one or many processes. The capabilities of a WPS contain the list of services which are hosted by the WPS. Each process is further described by the Process Description.

3.7. Stylesheet

A set of encoded instructions that determine how to render geographic features on a computer screen. The stylesheet is used to organize features into layers and to define rules that use the attributes of these features to determine how to physically represent these features using drawing instructions such as color, line width, fill patterns etc. A well known example of this sort of stylesheet is OGC Styled Layer Descriptor.

Chapter 4. Conventions

None.

4.1. Abbreviated terms

- API Application Program Interface
- TNM The National Map
- USGS United States Geological Survey
- WMS OGC Web Map Service
- WMTS OGC Web Map Tile Service
- WPS OGC Web Processing Service
- SE Symbology Encoding
- SLD OGC Styled Layer Descriptor
- CR Change Request
- GDB an ESRI © file geodatabase [3: <http://desktop.arcgis.com/en/arcmap/10.3/manage-data/administer-file-gdbs/file-geodatabases.htm>]
- GDAL Geospatial Data Abstraction Library [4: <http://www.gdal.org/>]
- MXD ESRI ArcMap map document [5: <http://support.esri.com/other-resources/gis-dictionary/term/map%20document>]
- SVG Scalable Vector Graphics [6: <https://www.w3.org/Graphics/SVG/>]

4.2. UML notation

Most diagrams that appear in this standard are presented using the Unified Modeling Language (UML) static structure diagram, as described in Subclause 5.2 of [OGC 06-121r9].

Chapter 5. Overview

5.1. Requirements

The requirements for this testbed-12 activity were:

- Document all problems and obstacles with using the USGS Topo Combined Vector Product to create the GeoPackage.
- Test a workflow and document all problems and obstacles with encoding the GeoPackage with standard USGS US Topo Map symbology as defined in the Topo TNM Style Template.
- Document the process used to convert the USGS Topo Combined Vector Product data to GeoPackage format (so that USGS can reproduce the GeoPackage format) must be recorded in the ER.
- Document the process used to encode GeoPackage data with US Topo Map symbology as defined in the Topo TNM Style Template (so that USGS can reproduce the encoded symbolized GeoPackage format) must be documented.
- Document the process for end users using the Use Cases described. Include any obstacles or limitations to the end users.

5.2. Use Cases

In a briefing presented to the GeoPackage Standards working group on Nov. 10th, 2015 by Randy Gladish of Image Matters LLC [7: https://portal.opengeospatial.org/files/?artifact_id=65723], use cases for styling and symbolization were articulated. Two of these use cases; UC4 and UC5 are addressed in the USGS Topo experiment.

- Geometry styling (UC4) - Geometry styles will be applied based on the application of styling directives such as those found in the OGC Style Layer Descriptor (SLD) standard. The output of the interpretation will be a set of symbols/drawing instructions cross-linked to the features they are meant to symbolize. The interpretation rules from the styling will not be carried over.
- Feature Type Styling (UC5) - similar to Geometry Styling, this approach relies on the interpretation of a set of styling directives that process feature attributes to determine the symbolization for a feature. The relevant rules for the style will either be stored in the GeoPackage to be interpreted at render time, or be pre-rendered into geometry styling. The benefit of maintaining these rules in the GeoPackage is that the symbolization of a feature can be re-interpreted if its attributes change. The USGS Topo experiment is designed to generate discussion regarding the implementation of UC5. The implementation created for the testbed will also expose an alternate method of styling within the OGC and it may be useful to provide the findings to spur more work developing and extending current styling methodologies or developing new ones.

5.3. Approaches

The processing of any GDBs will be done by a conversion engine. One publicly exposed interface is

a web accessible GUI that users can use to define GDB and OGC Web Map Tiling Service (WMTS) datasources. When a user prompts for the creation of a GeoPackage, the conversion engine will combine these datasources in combination with parameters such as Area of Interest (AOI), scale zoom levels and stylesheet to generate the required GeoPackage.

An additional interface will be a WPS providing clients the ability to programmatically follow the same flow.

5.3.1. Feature input source translation

Translation is performed using GDAL ogr2ogr which is a tool that translates various vector feature formats into other vector feature formats. The end result is a Spatialite database that can be worked with directly to query and encode into a GeoPackage. Feature classes in the FileGDB are directly translated to feature tables in the GeoPackage. Features are stored in a geographic projection.

5.3.2. Raster input source translation

Raster data is harvested directly using a WMTS/WMS client to populate a raster tile pyramid with tiles from online USGS tile and WMS sources. These include a shaded relief server and an imagery server. Tiles are provided in Web Mercator projection (EPSG:3857) [8: https://en.wikipedia.org/wiki/Web_Mercator]. This may prove to be less effective than providing the data in the correct UTM projection that the cell data should be viewed in.

Additional elevation data will be encoded in the GeoPackage produced using 1/9 arc second elevation data products directly available for download.

5.3.3. Styling approach

The approach to styling uses a stylesheet that includes:

- Filter rules for selecting features for symbology application.
- Appropriate matching of styles and scale denominators
- The definition of layers, and layer drawing order

5.3.4. Style and symbology encoding

- Style encoding according to UC4 is performed by mapping a set of symbologies directly to a feature type. Multiple symbologies can be applied to a single feature depending on scale and the creation of compound styles.
- The symbology encoding instructions such as line width or color are encoded in JSON in a single column in the table.

5.3.5. Raster tile encoding

- For the purposes of the experiment, raster tiles are encoded in a standard GeoPackage tile pyramid based on the Web Mercator tile matrix set.

5.3.6. Integration Use Case experiences

- A State Geologist intends to produce a 24,000-scale map using base map data and symbology in a US Topo Map design and layout. The Geologist will use ArcGIS to create the map and will add her own geologic data layers to the base data. The base data will be downloaded in the OGC GeoPackage and the US Topo Map specified symbology will be encoded with the GeoPackage.
- A County Emergency Manager would like to obtain symbolized base map data in a US Topo Map design to support an evacuation exercise. The Emergency Manager does not have any sophisticated GIS software. He has an iPad that he takes into the field with him and would like to be able to use a map-like display and query the attributes of the data. In addition, he would like to be able to display a hillshade and local imagery along with the map display.

5.3.7. Integration obstacles

- UTM projection v. Spherical Mercator. The data in the USGS FileGDBs is intended to be viewed in the correct UTM projection for the cell area. However, source data for hill Shading and orthoimagery is provided by WMTS services whose tiles are in Web Mercator.
- New methods for styling, storing generalized geometries and display hierarchies for views of a layer are all prototypical and are not defined in the current GeoPackage standard. To conform to the TNM template, these extensions to the GeoPackage standard are required, but will only be supported by client software capable of rendering this display.
- The requirement to create and maintain a proprietary stylesheet to support the proper creation of the GeoPackage can be a stumbling block as it requires extra training. The automatic generation of such a file is difficult as the proper tools are not available to generate this file. The ESRI MXD file is not readable. Therefore the stylesheet must be created manually.

5.3.8. Encoding Obstacles

- Determining symbology for display at different scales required a best guess. This is due to the fact that different scale ranges requiring different symbologies for the same feature is not part of the TNM template.
- Font sets are collections of encoded glyphs that are used by computer programs to display letters or icons. Styling approaches using Esri tools often use Unicode symbols from Esri supplied fontsets to create icons on the map. These symbols needed to be converted to images because the style system used in the experiment did not support this approach.
- SVG symbols should be used; instead, png symbols were created using screen shots of current symbology in ArcGIS
- There was no existing way to represent layer "hierarchies" as expressed in the TNM template.
- Some sophisticated filter expressions that exist in Esri tools are not supported directly e.g. "T" & Mid([PLSSID]. 5, 3)
- Font translation is tricky depending on the supported platforms.
- Contour lines have a polygon "z" shape. There is no way of preserving this in the stored geometries based on the GeoPackage design.
- Esri layer groupings in the MXD file allow for multiple "views" of the same feature class to

appear as "layers" in a client. In ArcMap, symbology can be assigned at this level. This was not supported in GeoPackage. Solutions to get around this are the creation of sub feature classes at the GeoPackage level; essentially a materialized view of the data or the merging of multiple styles under a single layer. Ultimately, in order to maintain the 1:1 mapping of feature classes to GeoPackage features, separate feature layers were created for each "view".

5.3.9. Process Obstacles

There were no significant process obstacles.

5.3.10. Interoperability Experiment Outcomes

While generated GeoPackage and the WPS service instances used for creating them were available early in the experiment, no client or service providers volunteered to participate in interoperability work despite invitations during telecons. Generated GeoPackages were visualized successfully in Compusult clients and new approaches for styling that were developed during the testbed were adopted by Compusult. Our best guess is that the nature of the symbology representation presented a challenge to GeoPackage clients that could not be met during the testbed. In the future, we recommend that the thread architects advocate for interoperability partnerships early in the testbed, so that there is at least one partnership per component.

5.4. Proposed style encoding extension

There are 2 aspects to the proposed style extension that are documented. One is the style encoding, the other is the relation of styles to the database. A style encoding consists of the symbology to be used at a particular scale, and in the case of dynamic styling, the filter expression used to evaluate a feature's symbology based on its attributes. The encoding will be documented in [Symbology Encoding](#).

5.5. Proposed hierarchical layer/raster extension

The Testbed-12 thread participants decided that the layer hierarchy not be expressed in the production of GeoPackages for this experiment.

Chapter 6. Status Quo & New Requirements Statement

6.1. Status Quo

- Currently, there is no method for encoding the styles used for feature portrayal in a GeoPackage.
- Currently, the USGS has topographic data and its portrayal specification is encoded in a particular format, arcmap/geodatabase.
- Currently, the USGS provides topo data as well as contextual imagery and shaded relief in a combined PDF product.

6.2. Requirements Statement

- USGS requires that their topographic data be encoded in a GeoPackage. Additionally, the style for that data must be encoded in the GeoPackage according to the cartographic design used for US Topo maps
- USGS requires that supporting orthoimagery and shaded relief be encoded as raster tiles in the GeoPackage
- To support the USGS requirements, a GeoPackage extension must be created to support styling the stored vector features.
- Some OGC members have requested that elevation data be packaged in the GeoPackage as well.

Chapter 7. Solutions

7.1. Targeted Solutions

In order to achieve the immediate goals of the for Testbed 12, the following solutions were adopted:

1. Encoding of topographic data and supported data in GeoPackage using the GeoPackage Converter WPS
2. Encoding of styles from the USGS topographic style template to be used to render topographic data on a client using a style specification system that employs attribute based symbology assignment.
3. Encoding of shaded relief and orthoimagery provided by WMS/WMTS using the Compusult GeoPackage Converter WPS.
4. Use of a mobile client to demonstrate offline rendering of vector features and supporting raster data according to use cases
5. Generation of a new GeoPackage extension to support vector feature styles for rendering derived from symbology encoding that supports re-styling after feature changes.
6. Elevation data for supporting the 7.5' x 7.5' quad to be encoded in a tile pyramid in the GeoPackage using a GeoPackage Elevation Extension that was prototyped during an interoperability experiment [9: <https://github.com/opengeospatial/geopackage-elevation>].
7. All data to be placed in Web Mercator to support available tile sets for shaded relief.

7.1.1. Converter WPS

In order to make the conversion of stock USGS data to a GeoPackage encoded product, a WPS was designed and delivered to perform two basic tasks:

1. Register data sources which provide the base data that will be encoded in the GeoPackage. The following data sources were used :
 - a. A Geodatabase provided by USGS as an example dataset for the Testbed 12 area of interest is: https://prd-tnm.s3.amazonaws.com/StagedProducts/Vector/GDB/VECTOR_27656_Mare_Island_7_5_Min.zip
 - b. Shaded Relief WMTS tiles are located at: <http://basemap.nationalmap.gov/arcgis/rest/services/USGSShadedReliefOnly/MapServer/WMTS/1.0.0/WMTSCapabilities.xml>
 - c. Wetlands tiles derived from: <http://107.20.228.18/arcgis/services/Wetlands/MapServer/WMSServer?request=GetCapabilities&service=WMS>
 - d. Imagery tiles derived from: http://isse.cr.usgs.gov/arcgis/services/Orthoimagery/USGS_EROS_Ortho/ImageServer/WMSServer?request=GetCapabilities&service=WMS
 - e. Intersecting elevation products provided by the USGS such as: [16](https://prd-</div><div data-bbox=)

tm.s3.amazonaws.com/StagedProducts/Elevation/19/IMG/ned19_n37x50_w122x00_ca_sanfrancisco_topobathy_2010.zip

2. Combine registered data sources over an area defined by the bounds of the named GDB.

The WPS operations are described in [Converter WPS Description](#)

A simple client was created to use the WPS and demonstrate the call/response flow. It is located at <http://ogc-tb12fo.compusult.net/wes/WPSClient.jsp>.

The screenshot below shows the client with some requests and responses inline as the client executes the creation of a GeoPackage.

Request Information

GDB Source : Stylesheet :

WMTS Source :

WMS Source 1: WMS Source 2:

Generating Geopackage...

WPS Request

WPS EXECUTE [CreateDatasource] :

```
<wps:Execute xmlns:wps="http://www.opengis.net/wps/2.0" xmlns:ows="http://www.opengis.net/ows/2.0"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS" version="2.0.0" response="document">
<ows:Identifier>CreateDatasource</ows:Identifier> <wps:Input id="DATASOURCE"> <wps:Input id="DATA"> <wps:Reference
xlink:href="https://prd-tnm.s3.amazonaws.com/StagedProducts/Vector/GDB/VECTOR_69123_Ackerson_Mountain_
</wps:Reference> </wps:Input> <wps:Input id="STYLE"> <wps:Reference xlink:href="http://ogc-
tb12fo.compusult.net/wes/WFPS/sld/tnm_stylesheet.zip"></wps:Reference> </wps:Input> <wps:Input id="TYPE">
<wps>Data>GEODATABASE</wps>Data> </wps:Input> <wps:Input id="NAME"> <wps>Data>Ackerson Mountain,
20160810</wps>Data> </wps:Input> <wps:Input id="DESCRIPTION"> <wps>Data>Created</wps>Data> </wps:Input
id="SRS"> <wps>Data>EPSG:3857</wps>Data> </wps:Input> </wps:Input> <wps:Output id="DATASOURCE_IDS"
</wps:Execute>
```

WPS GETSTATUS :

```
<?xml version="1.0" encoding="UTF-8"?> <wps:GetStatus xmlns:wps="http://www.opengis.net/wps/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../
service="WPS" version="2.0.0"> <wps:JobID>28c10f3b-eb45-472a-ae32-efd09bafffc9</wps:JobID> </wps:GetStatus
```

WPS GETSTATUS :

```
<?xml version="1.0" encoding="UTF-8"?> <wps:GetStatus xmlns:wps="http://www.opengis.net/wps/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../
service="WPS" version="2.0.0"> <wps:JobID>28c10f3b-eb45-472a-ae32-efd09bafffc9</wps:JobID> </wps:GetStatus
```

WPS GETSTATUS :

```
<?xml version="1.0" encoding="UTF-8"?> <wps:GetStatus xmlns:wps="http://www.opengis.net/wps/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../
service="WPS" version="2.0.0"> <wps:JobID>28c10f3b-eb45-472a-ae32-efd09bafffc9</wps:JobID> </wps:GetStatus
```

WPS GETSTATUS :

```
<?xml version="1.0" encoding="UTF-8"?> <wps:GetStatus xmlns:wps="http://www.opengis.net/wps/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../
service="WPS" version="2.0.0"> <wps:JobID>28c10f3b-eb45-472a-ae32-efd09bafffc9</wps:JobID> </wps:GetStatus
```

Figure 1. WPS Client Example

7.1.2. Resulting GeoPackage

The result of the conversion process yields a GeoPackage with the following table of contents :

Table 2. Table GeoPackage Contents

table_name	data_type	identifier	description
ned19_n38x25_w122x25_ca_sanfrancisco_topobathy_2010	2d-gridded-coverage	ned19_n38x25_w122x25_ca_sanfrancisco_topobathy_2010.tiff	
GPKG8caeccd6_5595_48a4_b09e_6666a799817f	tiles	USGSShadedReliefOnly	
GPKG76332686_db49_43ef_aa08_b61864ee8686	tiles	Imagery	
GPKG6758ae81_f0c7_406d_9965_60c1a2304351	tiles	Wetlands	
LANDCOVER_WOODLAND	features	Woodland	Woodland
NHDArea	features	Perennial NHDArea	Perennial NHDArea
Intermittent NHDWaterbody	features	Intermittent NHDWaterbody	Intermittent NHDWaterbody
Intermittent NHDArea	features	Intermittent NHDArea	Intermittent NHDArea
Wetlands	features	NWI Wetlands w/ US Topo Symbology	NWI Wetlands w/ US Topo Symbology
NHDWaterbody	features	Perennial NHDWaterbody	Perennial NHDWaterbody
Elev_Contour	features	Contours	Contours
NHDPoint	features	NHD Points	NHD Points
GagingStation	features	Gaging Station	Gaging Station
NHDFlowline	features	NHDFlowline	NHDFlowline
NHDLLine	features	NHDLLine	NHDLLine
GU_PLSSTownship	features	Township/Range	Township/Range
GU_PLSS SpecialSurvey	features	GU.GU_PLSS SpecialSurvey	GU.GU_PLSS SpecialSurvey
GU_PLSS FirstDivision	features	Section	Section
Trans_RoadSegment	features	Census Roads	Census Roads
trans_roadsegment_ntdnofs	features	USFS Roads	USFS Roads

table_name	data_type	identifier	description
Trans_TrailSegment	features	Trail	Trail
Trans_RailFeature	features	Railroad	Railroad
Trans_AirportRunway	features	AirportRunway	Airport Runway
GU_StateOrTerritory	features	State or Territory	State or Territory
GU_County OrEquivalent	features	County or Equivalent	County or Equivalent
GU_Reserve	features	National Cemetery	National Cemetery
NationalParkService	features	National Park Service	National Park Service
Department ofDefense	features	Department of Defense	Department of Defense
ForestService	features	Forest Service	Forest Service
FishandWildlife Service	features	Fish and Wildlife Service	Fish and Wildlife Service
GU_Native AmericanArea	features	NativeAmerican Lands	Native American Lands
Bureauof LandManagement	features	Bureau of Land Management	Bureau of Land Management
TNMDerivedNames	features	Structures from GNIS	Structures from GNIS
Struct_Point	features	Structures	Structures
Trans_AirportPoint	features	AirportPoint Labels Only	AirportPoint Labels Only
reQziqrD_cOst_hMDi_Dxpp_1XIPFPgUmlnr	features	TNM Derived Names	TNM Derived Names
CellGrid_7_5Minute	features	7.5_min Cells	7.5_min Cells

7.1.3. Style Encoding

Style encoding was accomplished through the use of a stylesheet that defined the appropriate symbology for feature layers based on filter criteria and scale. If there is no filter criteria pertaining to a style, the style was generalized for the whole feature type. In the process of styling the features, ancillary tables for the encoded feature types were created to relate individual features to:

1. A determined symbology based on scale
2. If required, a rule that determined the symbology based on scale and a rule attribute

In the process of constructing the symbology for a feature type, we determined that there are

limitations in that the layer hierarchy defined in the Topographic Map Template could not be followed as there was no way of directly encoding the groups implied in the template. There were two ways the feature data could be encoded as closely as possible to the Feature Classes that appear in the GeoDataBase.

Styles covering all required symbologies and feature types defined by definition queries were merged into a set of styles applied to the single feature class. A good example of this is the group of sub layers derived from the GU_Reserve feature class. On their own in the template MXD, there were the following "sub features" each with an implied style based on a criteria.

In this case, the method employed did not segregate the data into separate feature tables in the GeoPackage. Instead, all features remained in the same base table and were simply styled according to the requirements dictated in the original MXD based template.

The other approach was to generate individual feature tables derived from the query definitions in the TNM template. This was the approach chosen as the user would have similar control over each layer as they would when viewing the data in ArcMap.

7.2. Alternatives considered

Regarding encoding of style information in the GeoPackage itself, the OGC SLD/SE standard was considered. However, this did not meet the requirements this Testbed activity. We were not aware of any other efforts and decided therefore to use the JSON based in-house approach developed by Compusult. This approach is described in [UML For Symbology Encodings](#).

7.3. Recommendations

Having reviewed the GeoPackage output and comparing it to the available PDF maps that are provided by the USGS, and based on the experiences of this testbed, the following recommendations for encoding USGS Topo Data and supporting raster data as a GeoPackage are:

1. Encode all required data in proper UTM projection for the quad in question. For the testbed, the data was encoded in EPSG:3857, which is not ideal for the purpose. There are two main reasons this approach was adopted:
 - a. There is no formal tile matrix set proposed for individual UTM zones.
 - b. Supporting data sources were only available in EPSG:3857.
2. Ensure that supporting data sources are available in the desired projections as much as possible to ensure data quality. If the USGS provides the sources, they have control of the provenance and quality of the data.
3. Rigorous Quality Assurance needs to be done to ensure that the imagery and elevation products are accurately re-projected to the targeted projection
4. Data for non-critical datasets should be provided by WMS in order to provide UTM projected information. Primary examples would be wetlands and shaded relief data.
5. USGS provides metadata documents for the elevation and GDB products. A format for these metadata documents should be provided so that the metadata can be put in the correct place in

the GeoPackage. Metadata for basemap data produced by web services should be provided for this purpose as well.

6. WMTSes for source data will require larger scales to provide functionality parallel to pdf products. If this is not possible, WMS services should be made available for harvesting the data at these scales.
7. It would be beneficial for the USGS to provide a WCS interface that can be used to extract elevation data clipped to the bounds and correct coordinate system for the desired 7.5'x7.5' area. Currently a producer needs to clip and convert the elevation data provided in the zip file from the USGS. Providing this service will allow the USGS to control the quality of the elevation data provided in the required coordinate system.
8. In order to ease the development of styles based on all possible feature types and permutations, it would be a good idea to have a set of test data available that conformed to all possible style cases to verify that all the style encodings rendered correctly. Without this reference dataset, to prove all style rules, a producer would need to find sample data from across all available datasets.
9. It was noticed that the layer bounds for the Shaded Relief layer in the WMTS did not correspond to the bounds of its tile matrix. This should be fixed.

7.4. Styling Examples

The following screenshots show examples of styled topographic feature data displayed in the Compusult GOMobile Client.

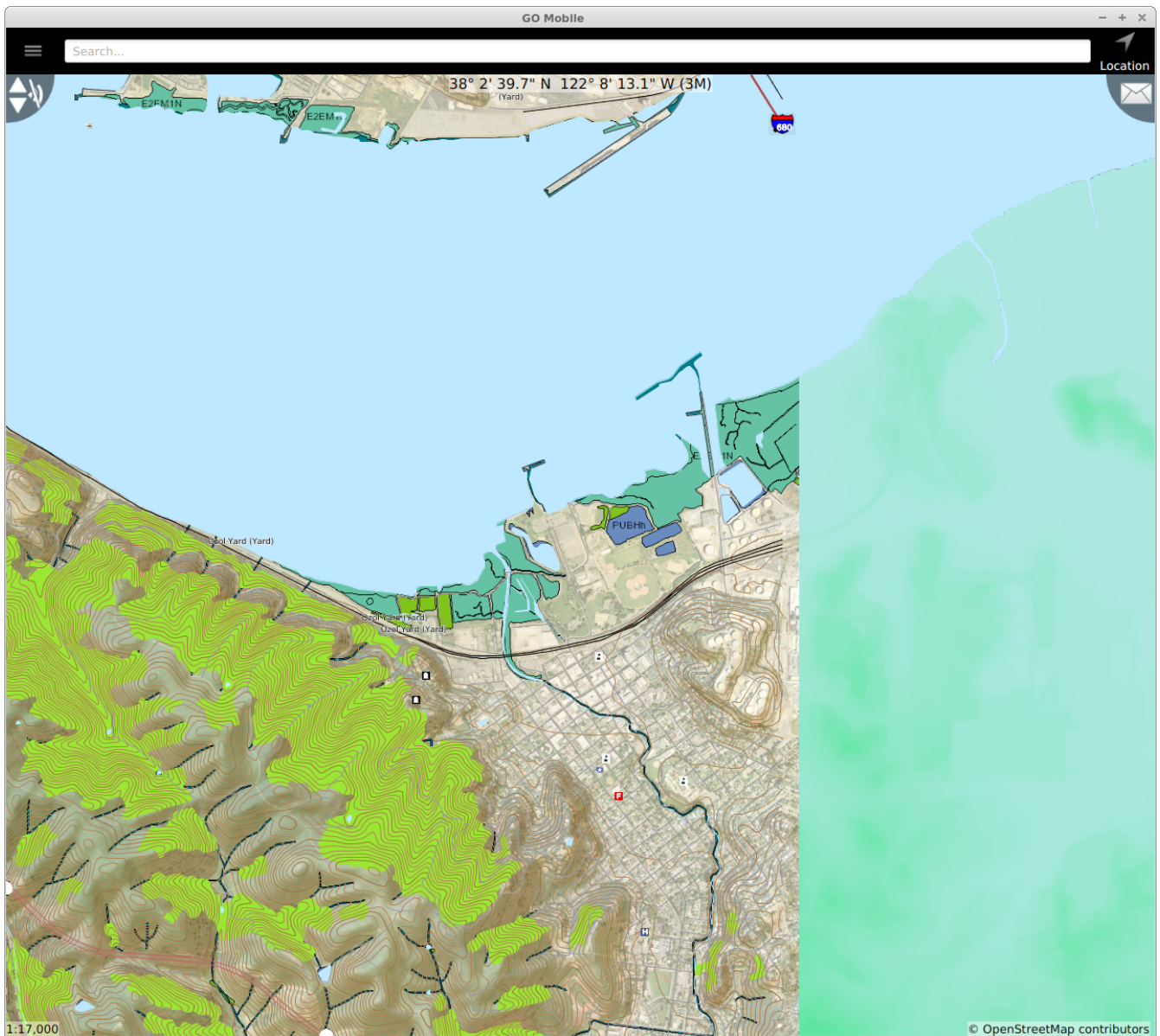


Figure 2. Styled data for Benicia

This image shows the following USGS feature types superimposed on imagery harvested from the imagery server mentioned above:

- Schools
- Fire Stations
- Place Names
- Contours with labels
- Post Offices
- Rail Line
- Forested Areas

This image also shows raster imagery from:

- Shaded Relief
- Imagery

- Wetlands
- and 1/9th arc second Elevation Data

This data was derived from the USGS GDB product available at https://prd-tnm.s3.amazonaws.com/StagedProducts/Vector/GDB/VECTOR_3330_Benicia_7_5_Min.zip and generated using the WPS Converter. The GeoPackage is located at https://portal.opengeospatial.org/files/?artifact_id=71497.



Figure 3. Styled data for Mare Island

This image shows the following USGS feature types superimposed on imagery harvested from the imagery server mentioned above as well as wetlands raster data and shaded relief.

Appendix A: Data Mappings from GDB to SQLite

This appendix explains and illustrates the mapping of some GDB tables and fields to SQLite tables and fields.

It also details the process used to name and organize layers and contents in the GeoPackage for the various sources of data.

A.1. Naming and Organizing Layers Created from the GeoDatabase Source

The following USGS topographical layers are defined in the contents listed in TopoTNMStyleTemplate_v10_0. Each layer was reproduced in parallel in the resulting GeoPackage as a result of the conversion.

Table 3. Mapping from TNM layers to GeoPackage Tables

Name from TNM Template	Generated GeoPackage Table
7.5_min Cells	CellGrid_7_5Minute
Airport Runway	Trans_AirportRunway
AirportPoint Labels Only	Trans_AirportPoint
Bureau of Land Management	BureauofLandManagement
Census Roads	Trans_RoadSegment
Contours	Elev_Contour
County or Equivalent	GU_CountyOrEquivalent
Department of Defense	DepartmentofDefense
Fish and Wildlife Service	FishandWildlifeService
Forest Service	ForestService
Gaging Station	GagingStation
GU.GU_PLSSSpecialSurvey	GU_PLSSSpecialSurvey
Intermittent NHDArea	IntermittentNHDArea
Intermittent NHDWaterbody	IntermittentNHDWaterbody
National Cemetery	GU_Reserve
National Park Service	NationalParkService
Native American Lands	GU_NativeAmericanArea
NHD Points	NHDPoint

Name from TNM Template	Generated GeoPackage Table
NHDFlowline	NHDFlowline
NHDLLine	NHDLLine
NWI Wetlands w/ US Topo Symbology	Wetlands
Perennial NHDArea	NHDArea
Perennial NHDWaterbody	NHDWaterbody
Railroad	Trans_RailFeature
Section	GU_PLSSFirstDivision
State or Territory	GU_StateOrTerritory
Structures	Struct_Point
Structures from GNIS	TNMDerivedNames
TNM Derived Names	reQzigrD_cOst_hMDi_Dxpp_1XlPFPgUMlnr
Township/Range	GU_PLSSTownship
Trail	Trans_TrailSegment
USFS Roads	trans_roadsegment_ntdnofs
Woodland	LANDCOVER_WOODLAND

The layer organization of the TNM Topo data in the TNM Template MXD file has a hierarchy. Within this hierarchy, some layers are derived directly from feature classes, others are derived using layer definition queries against a feature class. For example, Boundaries is a group that contains the following sub-layers:

Table 4. Boundaries Group Organization

Presentation Layer	Source Feature Class	Definition Query
State or Territory	GU_StateOrTerritory	
County or Equivalent	GU_CountyOrEquivalent	
National Cemetery	GU_Reserve	FType = 670
National Park Service	GU_Reserve	FType = 674
Forest Service	GU_Reserve	FTYPE = 671 OR FTYPE = 672
Fish and Wildlife Service	GU_Reserve	FTYPE = 676
Native American Lands	GU_NativeAmericanArea	NAME = 'Annette Island Reserve'
Bureau of Land Management	GU_Reserve	FTYPE = 675

As seen in [Boundaries Group Organization](#), there are 8 presentation layers created using 4 different

feature classes.

When mapped to the GeoPackage, it was decided to create feature tables for each presentation layer as it was defined in the table of contents. However, layer groups such as Boundaries were not represented in the GeoPackage.

For example "Fish and Wildlife Service" was mapped to a feature table in the GeoPackage called "FishandWildlifeService". In the GeoPacakage table of contents, the entry appears in [Fish and Wildlife gpkg_contents](#):

Table 5. Fish and Wildlife gpkg_contents

Column	Value
table_name	FishandWildlifeService
data_type	features
identifier	Fish and Wildlife Service
description	Fish and Wildlife Service
last_change	2016-10-27T18:27:45.583Z
min_x	-20037508.342789
min_y	-20037508.3427892
max_x	20037508.3427892
max_y	20037508.3427892
srs_id	3857

The generated feature table is below. In addition, the data types for each column are provided.

Table 6. FishAndWildlifeService table

column name	type
Source_Originator	TEXT
Distribution_Policy	TEXT
featureid	TEXT
Source_FeatureID	TEXT
Source_DatasetID	TEXT
GNIS_ID	TEXT
Shape_Length	REAL
OwnerOrManagingAgency	INTEGER
FCode	INTEGER
Shape_Area	REAL

FType	INTEGER
Source_DataDesc	TEXT
AreaSqKM	REAL
id	INTEGER
Name	TEXT
CSLT_SOURCE_TABLE	TEXT
Data_Security	INTEGER
LoadDate	TEXT
AdminType	INTEGER
Permanent_Identifier	TEXT
OBJECTID	INTEGER
the_geom	MULTIPOLYGON

Some new columns have been added beyond the columns in the original feature class:

- id - this is a primary key used for feature identification in the GeoPackage context
- CSLT_SOURCE_TABLE - this is the originating table/feature class from the GDB
- the_geom - this is the GeoPackage compatible representation of the original geometry from the GeoDatabase.

The entry in the processing stylesheet to generate this table is below:

Fish and Wildlife Definition

```
<Layer name="Fish and Wildlife Service" srs="+init=EPSG:4326" identifiable="1">
  <StyleName>FishandWildlifeService</StyleName>
  <Sources>
    <Source>
      <Table>GU_Reserve</Table>
      <SQL>SELECT *, ST_AsText(ST_GeomFromWKB(WKT_GEOMETRY)) geom FROM GU_Reserve
WHERE FTYPE = 676</SQL>
    </Source>
  </Sources>
</Layer>
```

[Fish and Wildlife Definition](#) illustrates that the feature type was populated using a query constructed from the basic query definition included in the TNM template.

A similar approach was followed for Hydrography Areas, two layers are derived from the NHDWaterbody feature class and two layers are derived from NHDArea feature class.

Appendix B: Symbology Encoding

This appendix covers the encoding of symbologies specific to the USGS Topographic case.

For any feature stored in a GeoPackage feature table, there may be 0 to many symbology encodings. It is up to a client application to determine how to render a feature with no symbology.

In the system used to render the USGS topographical data case, it is intended for clients to use the symbology linked to the current scale of the map. For any scale, a feature may have an assigned symbology. The linking of features to styles is discussed in [UML Models For Symbology Encoding](#).

For the purposes of the testbed, symbology was encoded using a JSON encoding. This annex discusses the JSON encoding and provides the encodings used for the USGS Topo data.

The JSON encoding is a straightforward encoding of symbology attributes in a KVP format such as strokeColor and strokeWeight, and their interpretation is determined by the geometry type of the feature being rendered. For example, a stroke attribute of a feature is interpreted as a border attribute for a polygon and a line attribute for a line feature.

The method of storing the symbology was chosen to be a single column in the styles table to support multiple encoding formats such as SE xml.

Supporting multiple encoding types has interoperability issues, and ultimately, as the GeoPackage community pursues symbology encoding and storage, it is hoped that the encodings supported are narrowed to those required and agreed upon as a group much the same as the image formats agreed upon in the tiling specification were narrowed to jpeg and png. It must be acknowledged that the requirements of various communities will force the need for more than one symbology encoding.

During production of a GeoPackage, filter rules based on feature type and attributes would be used to determine an appropriate style. In addition to filter rules, styles are grouped by scale range to determine which style would be applied by a rendering client at any particular scale. The symbology defined for the style is encoded in JSON and stored in the associated feature "_styles" table. The scale ranges are stored in the associated style_rules table. These styles are recorded as "fixed" styles used for rendering the feature.

In addition, to support changing feature attributes and corresponding styles, a separate set of tables to support these "dynamic" rules are constructed. These are linked to the main feature table using a mapping defined in the gpkgext_extended_contents using a mapping from the feature table name a style package name. The names of the tables associated with this style package start with the style package name followed by the suffixes "_dynamic_styles", "_dynamic_styles_expressions", "_dynamic_style_images". When a feature's attributes change, the client can look up the dynamic style expressions for that feature type and use each expression to determine if there is a match between the expression and the feature attributes. If there is a match, the style encoded in _dynamic_styles would be used to change the symbology encoded for the feature in the "fixed" style table so that the feature is rendered using the new symbology the next time the map is drawn.

JSON encoding was chosen for a number of reasons :

- JSON is more straightforward to parse than XML as would be the case with SE. When performance is a premium in mobile applications, this is an important consideration.
- A KVP approach is more flexible than a structured document in that new vendor specific extensions or versionable updates can be added, be backwards compatible and not be hampered by requiring schema adherence. A good example of this approach is the use of style attributes in HTML CSS. There are browser variants, deprecated attributes and experimental attributes in addition to core attributes defined in a specification.

Styling approaches also use images to represent features or fills. In order to maintain this approach and support portability and offline use, images used in styles are stored in a table for lookup. The reference to this image is accomplished in the symbology encoding by simply using the table id of the image in the style attribute :

imageencoding

```
{"imageOpacity":1,"image":9}
```

In the example [imageencoding](#) the image reference is simply a number. The number refers to the id of the image in the corresponding `_style_images` table (see [Hydrography Areas Style Images](#) as an example).

B.1. History

The JSON symbology encoding has its roots in the mapnik symbology stylesheet. Most attributes in the JSON encoding used in the testbed have mapnik equivalents and have the same function as those used in mapnik. There are also direct parallels with OGC Symbology Encoding as well as SVG names are commonly used.

Originally, the storage of the styles was done in a sparse table in which each possible attribute was stored in a column. It was determined that this would not work for two main reasons. It was not easily extensible, and did not support multiple encodings.

B.2. Examples

Here is an example of a JSON encoding for a Hydrography Line Feature:

```
{
  "strokeCap":"butt",
  "strokeJoin":"round",
  "strokeColor":"#bee8ff",
  "strokeOpacity":1,
  "strokeWeight":1
}
```

Here is an example of a JSON encoding for a Hydrography area label:

```
{
  "textColour": "#4eb4c7",
  "horizontalAlignment": "middle",
  "verticalAlignment": "middle",
  "labelExpression": "[GNIS_NAME]",
  "textPlacement": "point",
  "textOpacity": 1,
  "wrapWidth": 85,
  "haloRadius": 1,
  "haloColour": "white",
  "fontFamily": "DejaVu Sans Condensed Oblique",
  "fontSize": 9, "deltaX": 0, "deltaY": 0
}
```

Here is an example of a JSON encoding for a Hydrography area polygon:

```
{
  "fillColor": "#bee8ff", "fillOpacity": 1
}
```

Here is an example of a JSON encoding for a StructuresFromGNIS Point Feature using a png image as an example:

```
{
  "imageOpacity": 1, "image": 1
}
```

Looking up the image in the corresponding table, we can see that the value of 1 in the "image" style attribute corresponds to the Cemetery image:

Table 7. StructuresFromGNIS_style_images

id	path	data
1	/opt/Compusult/GPKGUpload/GiJ LqrWQ_tTlS_Cjgq_ntMM_YC3bZ utwCfum/images/Cemetery.png	BLOB
2	/opt/Compusult/GPKGUpload/GiJ LqrWQ_tTlS_Cjgq_ntMM_YC3bZ utwCfum/images/Post Office.png	BLOB

B.2.1. Contour Styling

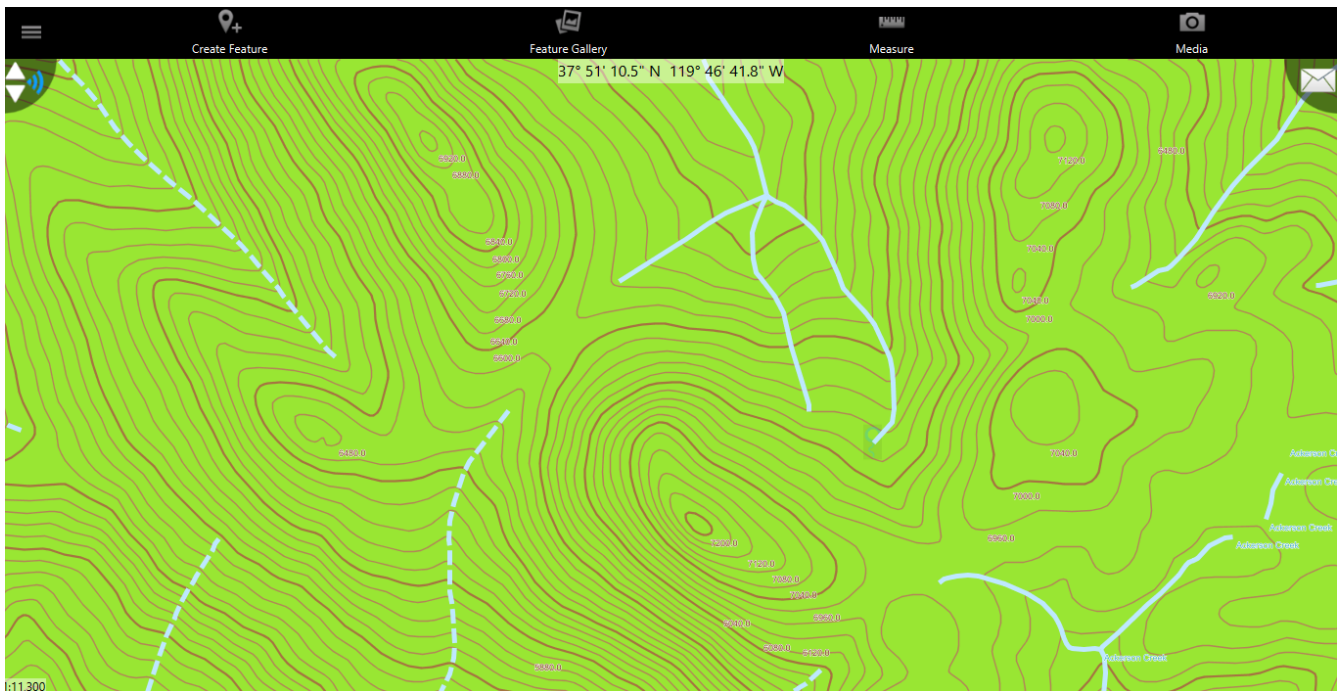


Figure 4. Contour Style data over Ackerson

This image shows dense contours in Yosemite. The contours appear on top of the green woodland area, drainage lines are also visible. One can see the index contours (100m intervals) appear as thicker lines. The ticks used in various depression encodings are represented in the "linePattern" attribute. The values for this attribute will point to ids in the corresponding dynamic styles images table.

B.2.2. Examples of symbology generated for Layers found in VECTOR_3330_Benicia_7_5_Min.gdb

The following examples list the query definitions and symbology applied for each query definition for feature classes found in the Benicia GDB. This is not an exhaustive list of examples for all possible types of features, but it should give a good representative example of the symbology developed for the experiment.

B.2.3. BureauofLandManagement

FTYPE = 675

```
{"fillColor":"#cdaa66","fillOpacity":1}
```

B.2.4. CellGrid_7_5Minute

null

```
{"strokeCap":"butt","strokeJoin":"round",
  "strokeColor":"#ff0000","strokeOpacity":1,"strokeWeight":2.5}
```

B.2.5. Elev_Contour

FCode = 10101

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#b38659","strokeOpacity":1,"strokeWeight":1}
```

FCode = 10102

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#a67442","strokeOpacity":1,"strokeWeight":2}
```

FCode = 10103

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#cfa48e","strokeOpacity":1,"strokeWeight":1}
```

FCode = 10104

```
{"strokeCap":"butt","strokeJoin":"round",  
  "linePattern":1,"strokeColor":"#b38659","strokeOpacity":1,"strokeWeight":1}
```

FCode = 10105

```
{"strokeCap":"butt","strokeJoin":"round",  
  "linePattern":2,"strokeColor":"#a67442","strokeOpacity":1,"strokeWeight":2}
```

FCode = 10106

```
{"strokeCap":"butt","strokeJoin":"round",  
  "linePattern":3,"strokeColor":"#cfa48e","strokeOpacity":1,"strokeWeight":1}
```

B.2.6. FishandWildlifeService

FTYPE = 676

```
{"fillColor":"#9ed7b4","fillOpacity":1}
```

B.2.7. GagingStation

(Permanent_Identifier is not null and length(trim(Permanent_Identifier) > 0) and Permanent_Identifier <> 'NULL') and Permanent_Identifier <> 'UNK')

```
{"imageOpacity":1,"image":1}
```

B.2.8. GU_NativeAmericanArea

NAME = 'Annette Island Reserve'

```
{"fillColor":"#ff9100","fillOpacity":1}
```

B.2.9. IntermittentNHDArea

FCode = 43100

```
{"fillPattern":1,"fillOpacity":1}
```

(FCode = 34300 or (FCode = 34306 or FCode = 34305))

```
{"fillColor":"#9c9c9c","fillOpacity":1}
```

FCode = 53700

```
{"fillPattern":2,"fillOpacity":1}
```

FCode = 36400

```
{"fillPattern":3,"fillOpacity":1}
```

FCode = 46003

```
{"fillPattern":4,"fillOpacity":1}
```

FCode = 48400

```
{"fillPattern":5,"fillOpacity":1}
```

(FCode = 40300 or (FCode = 40309 or (FCode = 40308 or FCode = 40307)))

```
{"fillPattern":6,"fillOpacity":1}
```

B.2.10. IntermittentNHDWaterbody

FCode = 37800

```
{"fillPattern":1,"fillOpacity":1}
```

FCode = 43605

```
{"fillPattern":2,"fillOpacity":1}
```

(FCode = 39001 or (FCode = 39005 or (FCode = 39006 or FCode = 43614)))

```
{"fillPattern":3,"fillOpacity":1}
```

FCode = 36100

```
{"fillPattern":4,"fillOpacity":1}
```

FCode = 43611

```
{"fillPattern":5,"fillOpacity":1}
```

B.2.11. LANDCOVER_WOODLAND

null

```
{"fillColor":"#99e633","fillOpacity":1}
```

B.2.12. NHDArea

(FCode = 46000 or FCode = 46006)

```
{"fillColor":"#bee8ff","fillOpacity":1}
```

(FCode = 33602 or (FCode = 33600 or FCode = 33601))

```
{"fillColor":"#bee8ff","fillOpacity":1}
```

(FCode = 31200 or (FCode = 36200 or FCode = 44500))

```
{"fillColor":"#bee8ff","fillOpacity":1}
```

B.2.13. NHDFlowline

(FCode = 42808 or (FCode = 42806 or (FCode = 42816 or (FCode = 42805 or (FCode = 42804 or (FCode = 42815 or (FCode = 42814 or (FCode = 42802 or (FCode = 42813 or (FCode = 42801 or (FCode = 42812 or (FCode = 42800 or (FCode = 42810 or FCode = 42809)))))))))))))

```
{"strokeCap":"butt","strokeJoin":"round",  
"strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```

(FCode = 46000 or FCode = 46006)

```
{"strokeCap":"butt","strokeJoin":"round",  
"strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```

(FCode = 33601 or FCode = 33600)

```
{"strokeCap":"butt","strokeJoin":"round",  
"strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```

FCode = 56600

```
{"strokeCap":"butt","strokeJoin":"round",  
"strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```

FCode = 42000

```
{"strokeCap":"butt","strokeJoin":"round",  
"strokeColor":"#343434","strokeOpacity":1,"strokeWeight":1}
```

(FCode = 42803 or (FCode = 42807 or FCode = 42811))

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
"strokeJoin":"round","strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```

FCode = 46003

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
"strokeJoin":"round","strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```


B.2.14. NHDLine

(FCode = 34306 or (FCode = 36900 or (FCode = 39800 or FCode = 48300)))

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#343434","strokeOpacity":1,"strokeWeight":1}
```

(FCode = 34300 or FCode = 34305)

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#9f551f","strokeOpacity":1,"strokeWeight":1}
```

FCode = 36200

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```

FCode = 56800

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#9f551f","strokeOpacity":1,"strokeWeight":1}
```

FCode = 41100

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#343434","strokeOpacity":1,"strokeWeight":1}
```

(FCode = 43100 or FCode = 48700)

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```

FCode = 43400

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#343434","strokeOpacity":1,"strokeWeight":1}
```

FCode = 45000

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#bee8ff","strokeOpacity":1,"strokeWeight":1}
```

FCode = 47800

```
{"dashPattern": "15.0,5.0", "strokeCap": "butt",
  "strokeJoin": "round", "strokeColor": "#bee8ff", "strokeOpacity": 1, "strokeWeight": 1}
```

FCode = 47800

```
{"dashPattern": "15.0,5.0", "strokeCap": "butt",
  "strokeJoin": "round", "strokeColor": "#bee8ff", "strokeOpacity": 1, "strokeWeight": 1}
```

B.2.15. NHDPoint

(FCode = 44100 or (FCode = 44101 or FCode = 44102))

```
{"imageOpacity": 1, "image": 1}
```

(FCode = 34300 or (FCode = 34305 or FCode = 34306))

```
{"imageOpacity": 1, "image": 2}
```

FCode = 36900

```
{"imageOpacity": 1, "image": 3}
```

FCode = 39800

```
{"imageOpacity": 1, "image": 4}
```

FCode = 43100

```
{"imageOpacity": 1, "image": 5}
```

(FCode = 43600 or (FCode = 43601 or (FCode = 43618 or (FCode = 43619 or (FCode = 43609 or (FCode = 43603 or (FCode = 43606 or (FCode = 43625 or (FCode = 43626 or (FCode = 43607 or (FCode = 43623 or (FCode = 43610 or (FCode = 43611 or (FCode = 43612 or (FCode = 43605 or (FCode = 43604 or (FCode = 43624 or (FCode = 43617 or (FCode = 43614 or (FCode = 43615 or (FCode = 43613 or FCode = 43621))))))))))))))))))))))))))))))

```
{"imageOpacity": 1, "image": 6}
```

FCode = 43608

```
{"imageOpacity":1,"image":7}
```

FCode = 45800

```
{"imageOpacity":1,"image":8}
```

FCode = 48700

```
{"imageOpacity":1,"image":9}
```

FCode = 48800

```
{"imageOpacity":1,"image":10}
```

B.2.16. NHDWaterbody

FCode = 49300

```
{"fillColor":"#bee8ff","fillOpacity":1}
```

(FCode = 43619 or (FCode = 43603 or (FCode = 43626 or (FCode = 43607 or (FCode = 43608 or (FCode = 43609 or (FCode = 43610 or (FCode = 43612 or (FCode = 43623 or (FCode = 43624 or FCode = 43613))))))))))

```
{"fillColor":"#bee8ff","fillOpacity":1}
```

(FCode = 43618 or (FCode = 39000 or (FCode = 39004 or (FCode = 39009 or (FCode = 39011 or (FCode = 39010 or (FCode = 39012 or (FCode = 43600 or (FCode = 43601 or (FCode = 43604 or (FCode = 43606 or (FCode = 43625 or (FCode = 43625 or (FCode = 43615 or (FCode = 43617 or FCode = 43621))))))))))))))

```
{"fillColor":"#bee8ff","fillOpacity":1}
```

B.2.17. Struct_Point

FCode = 83006

```
{"imageOpacity":1,"image":1}
```

(FCode = 73002 or (FCode = 73003 or (FCode = 73004 or (FCode = 73005 or FCode = 73006))))

```
{"imageOpacity":1,"image":2}
```

FCode = 74026

```
{"imageOpacity":1,"image":3}
```

FCode = 80012

```
{"imageOpacity":1,"image":4}
```

FCode = 74034

```
{"imageOpacity":1,"image":5}
```

FCode = 74036

```
{"imageOpacity":1,"image":6}
```

B.2.18. TNMDerivedNames

GAZ_FEATURECLASS = 'Cemetery'

```
{"imageOpacity":1,"image":1}
```

GAZ_FEATURECLASS = 'Post Office'

```
{"imageOpacity":1,"image":2}
```

B.2.19. Trans_RailFeature

FCode = 20200

```
{"strokeCap":"butt","strokeJoin":"round",  
  "linePattern":1,"strokeColor":"#000000","strokeOpacity":1,"strokeWeight":1}
```

B.2.20. Trans_RoadSegment

TNMFRC = 1

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#af4b4c","strokeOpacity":1,"strokeWeight":2.5}
```

TNMFRC = 2

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#c96767","strokeOpacity":1,"strokeWeight":1}
```

TNMFRC = 3

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#6e6f6e","strokeOpacity":1,"strokeWeight":1}
```

TNMFRC = 4

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#9b9b9b","strokeOpacity":1,"strokeWeight":1}
```

TNMFRC = 5

```
{"strokeCap":"butt","strokeJoin":"round",  
  "strokeColor":"#c96767","strokeOpacity":1,"strokeWeight":1}
```

TNMFRC = 6

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#9b9b9b","strokeOpacity":1,"strokeWeight":1}
```

TNMFRC = 7

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#9b9b9b","strokeOpacity":1,"strokeWeight":1}
```

TNMFRC = 8

```
{"dashPattern":"15.0,4.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#9c9c9c","strokeOpacity":1,"strokeWeight":1}
```

TNMFRC = 8

```
{ "dashPattern": "15.0,4.0", "strokeCap": "butt",  
  "strokeJoin": "round", "strokeColor": "#9c9c9c", "strokeOpacity": 1, "strokeWeight": 1 }
```

B.2.21. trans_roadsegment_ntdnofs

TNMFRC = 1

```
{ "strokeCap": "butt", "strokeJoin": "round",  
  "strokeColor": "#af4b4c", "strokeOpacity": 1, "strokeWeight": 2.5 }
```

TNMFRC = 2

```
{ "strokeCap": "butt", "strokeJoin": "round",  
  "strokeColor": "#c96767", "strokeOpacity": 1, "strokeWeight": 1 }
```

TNMFRC = 3

```
{ "strokeCap": "butt", "strokeJoin": "round",  
  "strokeColor": "#6e6f6e", "strokeOpacity": 1, "strokeWeight": 1 }
```

TNMFRC = 4

```
{ "strokeCap": "butt", "strokeJoin": "round",  
  "strokeColor": "#9b9b9b", "strokeOpacity": 1, "strokeWeight": 1 }
```

TNMFRC = 5

```
{ "strokeCap": "butt", "strokeJoin": "round",  
  "strokeColor": "#c96767", "strokeOpacity": 1, "strokeWeight": 1 }
```

TNMFRC = 6

```
{ "dashPattern": "15.0,5.0", "strokeCap": "butt",  
  "strokeJoin": "round", "strokeColor": "#9b9b9b", "strokeOpacity": 1, "strokeWeight": 1 }
```

TNMFRC = 7

```
{ "dashPattern": "15.0,5.0", "strokeCap": "butt",  
  "strokeJoin": "round", "strokeColor": "#9b9b9b",  
  "strokeOpacity": 1, "strokeWeight": 1 }
```

TNMFRC = 8

```
{"dashPattern":"15.0,4.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#9c9c9c",  
  "strokeOpacity":1,"strokeWeight":1}
```

TNMFRC = 8

```
{"dashPattern":"15.0,4.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#9c9c9c","strokeOpacity":1,"strokeWeight":1}
```

B.2.22. Trans_TrailSegment

FCode = 20600

```
{"dashPattern":"15.0,5.0","strokeCap":"butt",  
  "strokeJoin":"round","strokeColor":"#9b9b9b","strokeOpacity":1,"strokeWeight":1}
```

B.2.23. Wetlands

WETLAND_TYPE = 'Freshwater Emergent Wetland'

```
{"fillPattern":1,"fillOpacity":1}
```

WETLAND_TYPE = 'Freshwater Forested/Shrub Wetland'

```
{"fillPattern":2,"fillOpacity":1}
```

Appendix C: UML Models for Symbology Extension

This appendix provides a UML model of the extension tables used to encode styling in GeoPackage for this experiment. The intent of this appendix is to illustrate these tables and their usage via an example for the USGS HydrographyAreas feature class.

[Static Style Table Example](#) is a DB profile UML diagram illustrating the tables involved in the symbolization of a specific feature table HydrographyAreas. These tables are required to use the static styling approach where a specific symbology is mapped to a specific feature using a style _links table. Following [Static Style Table Example](#) are detailed descriptions of the following extended tables :

HydrographyAreas_style_images

A table containing binary encoded images referenced in symbologies

HydrographyAreas_style_rules

A table defining style rule metadata such as applicable scale ranges and rule types

HydrographyAreas_styles

A table that contains encoded styles defining the style, its encoding method and the priority/order of the style

HydrographyAreas_style_links

A table mapping style rules to individual features

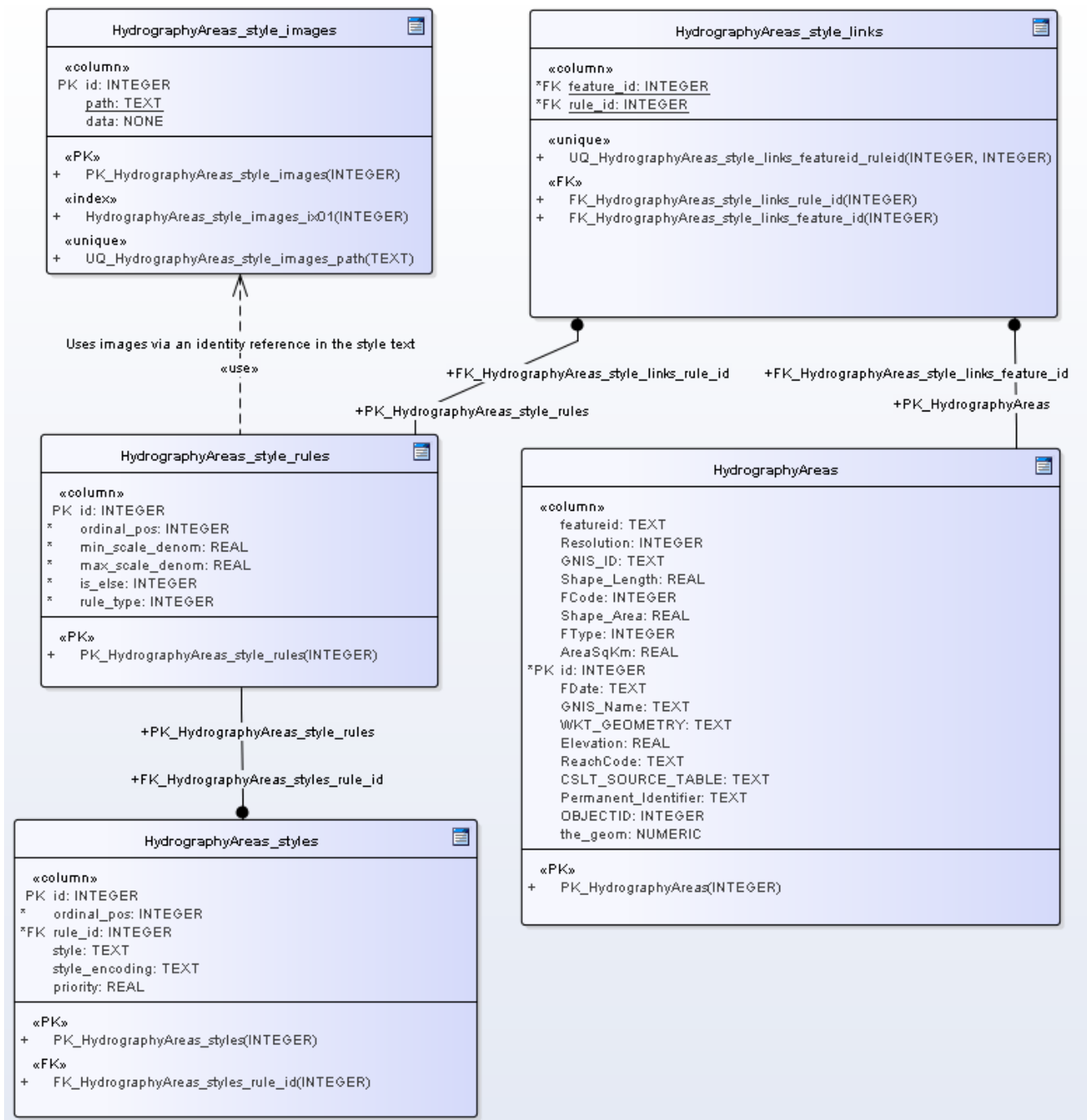


Figure 5. Static Style Table Example

Table 8. HydrographyAreas_style_rules

Column Name	Data Type	Description	Key
id	INTEGER	The primary key of the rule limited to values in the range [0, 9223372036854775807]	PK

Column Name	Data Type	Description	Key
ordinal_pos	INTEGER	indicates the order of application for the rule if there are more than one style rules applicable for a feature. Falls into the range [0, 9223372036854775807].	
min_scale_denom	REAL	A real value used for evaluating when the rule should be applied to a feature. When the scale denominator of the visible map is > min_scale_denom, the rule is active	
max_scale_denom	REAL	A real value used for evaluating when the rule should be applied to a feature. When the scale denominator of the visible map is < max_scale_denom, the rule is active	
is_else	INTEGER	a positive number that indicates true (1) or false (0). A true value indicates that the rule should short circuit any further evaluation of rules for a feature. A false indicates that further rules can be evaluated and applied.	rule_type

Table 9. HydrographyAreas_style_images

Column Name	Data Type	Description	Key
id	INTEGER	The primary key of the image limited to values in the range [0, 9223372036854775807]	PK
path	TEXT	The original path of the symbol such as "/usr/share/images/tower.png"	
data	BLOB	binary data	

Table 10. HydrographyAreas_styles

Column Name	Data Type	Description	Key
id	INTEGER	The primary key of the style limited to values in the range [0, 9223372036854775807]	PK
ordinal_pos	INTEGER	indicates the order of application for the style if there are more than one style rules applicable for a feature. Falls into the range [0, 9223372036854775807].	
rule_id	INTEGER	A positive number that is a foreign key to the corresponding rules table	FK
style	TEXT	The encoded style or symbology instructions detailing how the feature should appear	
style_encoding	TEXT	a mime type that indicates how the style is encoded e.g. "application/vnd.ogc.se_xml"	
priority	REAL	A priority used to weight the symbolization of the feature used to remove cluttering	

Table 11. HydrographyAreas_style_links

Column Name	Data Type	Description	Key
feature_id	INTEGER	A foreign key to a feature in the related feature table	FK
rule_id	INTEGER	A foreign key to a rule related rule table	FK

In addition to supporting direct representational styling an additional extension is used in GeoPackage to support dynamic styling of features. Dynamic styling is applicable when the attributes of a feature that determine its appearance have changed. For example, a user changes a road attribute such as lanes from "2" to "4". Styling rules exist that require road features with a lane value of "4" to be rendered with a median. Unless this rule is dynamically interpreted, the road will

not be updated visually unless there is a mechanism to find an applicable set of style rules for the changed feature and determine which one applies to the new condition. If such a mechanism exists, the applicable rule will be used to determine the new symbology. In the road case, the expression `lanes=4` would be encountered, the corresponding symbology would be set for the feature and clients would now render the feature with a median.

To support this dynamic mechanism, a further extension to the style extension is encoded to encompass the necessary links between symbology and rule expressions.

[Dynamic Style Table Example](#) is a DB profile UML diagram illustrating the tables used to support the dynamic symbolization of a specific feature table `HydrographyAreas`. The core idea is that a set of expressions are evaluated for any feature in the feature table if the match expression is satisfied, any style from the dynamic styles table that is linked to that expression is applied to the feature. Application of these styles could occur at run time, or they can be computed and stored in the data model for static styles. Following [Dynamic Style Table Example](#) are detailed descriptions of the following extended tables. These extended tables are linked via a mapping in `gpkg_extended_contents` that maps the feature table to the corresponding style tables below. :

[HydrographyAreas_dynamic_style_images](#)

A table containing binary encoded images referenced in symbologies used in dynamic styles.

[HydrographyAreas_dynamic_styles](#)

A table defining style rule metadata such as applicable scale ranges encoded styles defining the symbology, its encoding method and the painting priority of the style.

[HydrographyAreas_dynamic_style_expressions](#)

A table that contains match expressions in SQL syntax and order of evaluation for all match expressions.

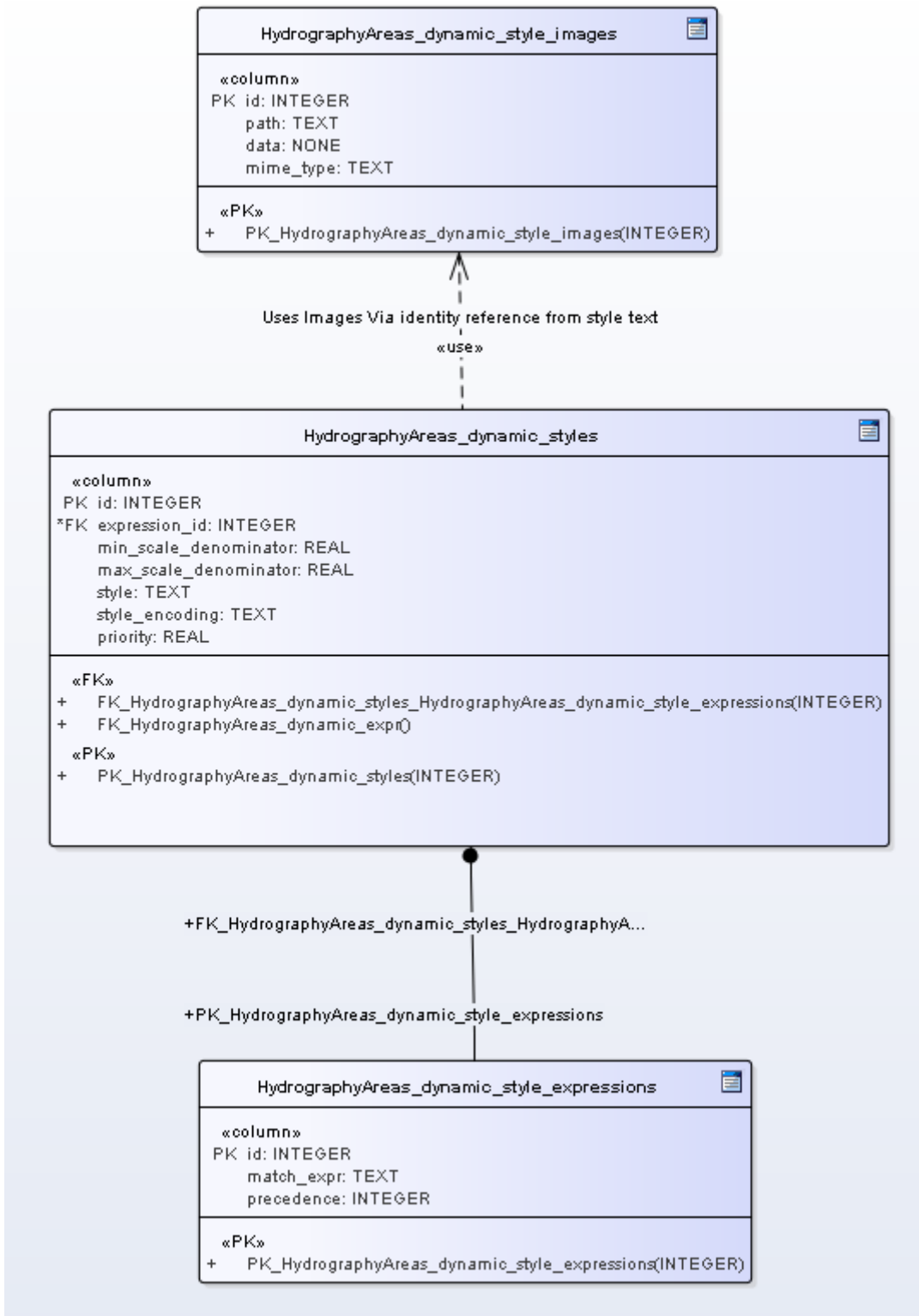


Figure 6. Dynamic Style Table Example

Table 12. HydrographyAreas_dynamic_style_images

Column Name	Data Type	Description	Key
id	INTEGER	The primary key of the image limited to values in the range [0, 9223372036854775807]	PK

Column Name	Data Type	Description	Key
path	TEXT	The original path of the symbol such as "/usr/share/images/tower.png"	
data	BLOB	binary data	
mime_type	TEXT	mime type of the stored image such as "image/png"	

Table 13. HydrographyAreas_dynamic_styles

Column Name	Data Type	Description	Key
id	INTEGER	The primary key of the rule limited to values in the range [0, 9223372036854775807]	PK
expression_id	INTEGER	A foreign key to the dynamic_style_expressions table for the feature table	FK
min_scale_denom	REAL	A real value used for evaluating when the style should be applied to a feature. When the scale denominator of the visible map is > min_scale_denom, the style is active	
max_scale_denom	REAL	A real value used for evaluating when the style should be applied to a feature. When the scale denominator of the visible map is < max_scale_denom, the style is active	
style	TEXT	the text encoding of the symbology information for example an OGC symbology encoding document	
style_encoding	TEXT	the mime type of the style encoding such as "application/vnd.ogc.se_xml"	

Column Name	Data Type	Description	Key
priority	REAL	indicates the order of application for the styles if there are more than one styles applicable for a feature.	

Table 14. HydrographyAreas_dynamic_style_expressions

Column Name	Data Type	Description	Key
id	INTEGER	A foreign key to a feature in the related feature table	PK
match_expr	TEXT	A SQL expression that can be applied to attributes of the feature, the intention being that it can be used directly in queries against the feature table	
precedence	INTEGER	This value determines the order of evaluation for all match expressions associated with a feature	

Appendix D: Converter WPS Description

This appendix describes the operations available to clients of the GeoPackage generator WPS. This WPS takes a different approach than other WPSes that take a context document with offerings as input. JSON documents describe the nature of each datasource. Also JSON documents describe how to create a GeoPackage from multiple data sources. This WPS can also take multiple datasource arguments to generate a single GeoPackage.

The WPS is located at the following URL : <http://ogc-tb12fo.compusult.net/wes/GeopackageWPS>

D.1. Potential CR

WPS does not support arbitrary numbers of outputs of the same type. This introduces extra processing to the client because we want to call "Make these n data sources", the results are not matched to the requested data sources, so a custom client must be created to process a custom response that would parse the results. The nature of the results is not automatically discoverable from the capabilities, however the nature of the inputs is discoverable.

This issue was filed as an official CR and is visible at : http://ogc.standardstracker.org/show_request.cgi?id=403

D.2. High Level Map of Input/Output

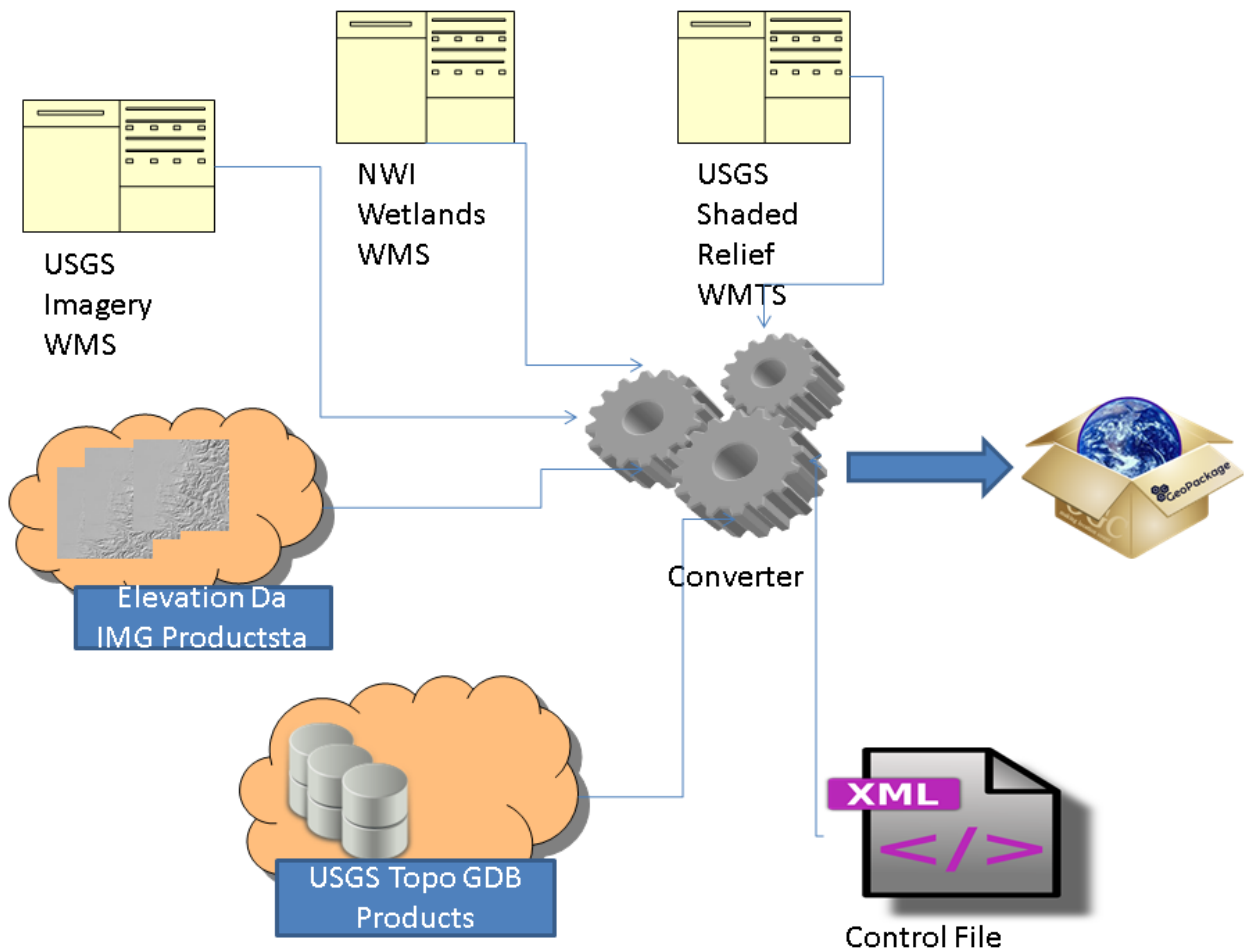


Figure 7. Diagram of inputs

In this diagram, we can see the inputs from external sources for a single GeoPackage consist of:

1. Tiles created from the USGS Imagery WMS
2. Tiles created from the NWI Wetlands WMS
3. Tiles directly from the USGS Shaded Relief WMTS
4. 1/9th arc second IMG Elevation products from the USGS downloaded directly from the Amazon cloud store
5. USGS Topo GDB products downloaded directly from the Amazon cloud store
6. The stylesheet that indicates how to use the feature classes from the GDB to create feature layers and symbolize them

D.3. GetCapabilities

```
<wps:GetCapabilities
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd"
  service="WPS">
</wps:GetCapabilities>
```

D.4. Datasource operations

D.4.1. CreateDatasource (GDB)

```
<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd"
  service="WPS" version="2.0.0" response="document" mode="async">
  <ows:Identifier>CreateDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE">
    <wps:Input id="DATA">
      <wps:Reference xlink:href="http://aparsons-
1.compusult.net:8888/wes/banner/images/VECTOR_22352_Jamaica_7_5_Min.zip"></wps:Referen
ce>
    </wps:Input>
    <wps:Input id="STYLE">
      <wps:Reference xlink:href="http://aparsons-
1.compusult.net:8888/wes/banner/images/tm_stylesheet.zip"></wps:Reference>
    </wps:Input>
    <wps:Input id="TYPE">
      <wps>Data>GEODATABASE</wps>Data>
    </wps:Input>
    <wps:Input id="NAME">
      <wps>Data>Jamaica New</wps>Data>
    </wps:Input>
    <wps:Input id="DESCRIPTION">
      <wps>Data>Created</wps>Data>
    </wps:Input>
    <wps:Input id="SRS">
      <wps>Data>EPSG:4326</wps>Data>
    </wps:Input>
  </wps:Input>
  <wps:Output id="DATASOURCE_IDS">
  </wps:Output>
</wps:Execute>
```

D.4.2. CreateDatasource (WMTS)

```

<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd"
  service="WPS" version="2.0.0" response="document" mode="async">
  <ows:Identifier>CreateDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE">
    <wps:Input id="DATA">
      <wps:Reference
xlink:href="http://basemap.nationalmap.gov/arcgis/rest/services/USGSShadedReliefOnly/M
apServer/WMTS/1.0.0/WMTSCapabilities.xml"></wps:Reference>
      </wps:Input>
      <wps:Input id="STYLE">
        <wps:Data>default028mm</wps:Data>
      </wps:Input>
      <wps:Input id="TYPE">
        <wps:Data>WMTS</wps:Data>
      </wps:Input>
      <wps:Input id="MIN_SCALE">
        <wps:Data>0</wps:Data>
      </wps:Input>
      <wps:Input id="MAX_SCALE">
        <wps:Data>10</wps:Data>
      </wps:Input>
      <wps:Input id="IDENTIFIER">
        <wps:Data>USGSShadedReliefOnly</wps:Data>
      </wps:Input>
      <wps:Input id="MIME_TYPE">
        <wps:Data>image/jpg</wps:Data>
      </wps:Input>
      <wps:Input id="NAME">
        <wps:Data>WMTS Complete</wps:Data>
      </wps:Input>
      <wps:Input id="DESCRIPTION">
        <wps:Data>Created WMTS</wps:Data>
      </wps:Input>
      <wps:Input id="SRS">
        <wps:Data>EPSG:3857</wps:Data>
      </wps:Input>
    </wps:Input>
    <wps:Output id="DATASOURCE_IDS">
  </wps:Output>
</wps:Execute>

```

D.4.3. UpdateDatasource

```

<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd"
  service="WPS" version="2.0.0" response="document" mode="sync">
  <ows:Identifier>UpdateDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE">
    <wps:Input id="DATA">
      <wps:Reference xlink:href="http://aparsons-
1.compusult.net:8888/wes/banner/images/VECTOR_22352_Jamaica_7_5_Min.zip"></wps:Referen
ce>
      </wps:Input>
      <wps:Input id="STYLE">
        <wps:Reference xlink:href="http://aparsons-
1.compusult.net:8888/wes/banner/images/tnm_stylesheet.zip"></wps:Reference>
      </wps:Input>
      <wps:Input id="TYPE">
        <wps>Data>GEODATABASE</wps>Data>
      </wps:Input>
      <wps:Input id="NAME">
        <wps>Data>Jamaica (Updated)</wps>Data>
      </wps:Input>
      <wps:Input id="DESCRIPTION">
        <wps>Data>Updated</wps>Data>
      </wps:Input>
      <wps:Input id="SRS">
        <wps>Data>EPSG:4326</wps>Data>
      </wps:Input>
      <wps:Input id="ID">
        <wps>Data>92</wps>Data>
      </wps:Input>
    </wps:Input>
    <wps:Output id="RESULT">
  </wps:Output>
</wps:Execute>

```

D.4.4. DeleteDatasource

```

<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd"
  service="WPS" version="2.0.0" response="raw" mode="sync">
  <ows:Identifier>DeleteDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE_ID">
    <wps:Data>9</wps:Data>
  </wps:Input>
  <wps:Output id="RESULT">
  </wps:Output>
</wps:Execute>

```

D.5. GeoPackage operations

D.5.1. CreateGeoPackage

```

<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd"
  service="WPS" version="2.0.0" response="document" mode="sync">
  <ows:Identifier>CreateGeoPackage</ows:Identifier>
  <wps:Input id="INPUT_GEOPACKAGE">
    <wps:Data>{
      "title": "TNM Combined",
      "descr": "TNM Combined from WPS",
      "geom": "POLYGON((-125 38.4,-125 40.9,-121.8 40.9,-121.8 38.4,-125 38.4))",
      "ds_ids": "97,110",
      "portfolioIds": "154",
    } </wps:Data>
  </wps:Input>
  <wps:Output id="GENERATION_STATUS">
  </wps:Output>
</wps:Execute>

```

D.5.2. Transcript

The following transcript shows the process of creating datasources and the ultimate GeoPackage.

```

<?xml version="1.0" encoding="UTF-8"?>
<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"

```

```

xmlns:ows="http://www.opengis.net/ows/2.0"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
version="2.0.0" response="document" mode="async">
  <ows:Identifier>CreateDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE">
    <wps:Input id="DATA">
      <wps:Reference xlink:href="https://prd-
tnm.s3.amazonaws.com/StagedProducts/Vector/GDB/VECTOR_3330_Benicia_7_5_Min.zip"></wps:
Reference>
    </wps:Input>
    <wps:Input id="STYLE">
      <wps:Reference xlink:href="http://ogc-
tb12fo.compusult.net/wes/WFPS/sld/tnm_stylesheet.zip"></wps:Reference>
    </wps:Input>
    <wps:Input id="TYPE">
      <wps>Data>GEODATABASE</wps>Data>
    </wps:Input>
    <wps:Input id="NAME">
      <wps>Data>Benicia, California 20160809</wps>Data>
    </wps:Input>
    <wps:Input id="DESCRIPTION">
      <wps>Data>Created</wps>Data>
    </wps:Input>
    <wps:Input id="SRS">
      <wps>Data>EPSG:3857</wps>Data>
    </wps:Input>
  </wps:Input>
  <wps:Output id="DATASOURCE_IDS"></wps:Output>
</wps:Execute>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>afa2f142-54c1-490d-bdf5-2781b0938754</JobID>
  <Status>Accepted</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:37:43.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>afa2f142-54c1-490d-bdf5-2781b0938754</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

```

```

<JobID>afa2f142-54c1-490d-bdf5-2781b0938754</JobID>
<Status>Running</Status>
<ExpirationDate xsi:nil="true"/>
<EstimatedCompletion xsi:nil="true"/>
<NextPoll>2016-10-27T15:37:43.000-02:30</NextPoll>
<PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>afa2f142-54c1-490d-bdf5-2781b0938754</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>afa2f142-54c1-490d-bdf5-2781b0938754</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:37:43.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>afa2f142-54c1-490d-bdf5-2781b0938754</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>afa2f142-54c1-490d-bdf5-2781b0938754</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:37:43.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>afa2f142-54c1-490d-bdf5-2781b0938754</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

```

```

<JobID>afa2f142-54c1-490d-bdf5-2781b0938754</JobID>
<Status>Running</Status>
<ExpirationDate xsi:nil="true"/>
<EstimatedCompletion xsi:nil="true"/>
<NextPoll>2016-10-27T15:37:43.000-02:30</NextPoll>
<PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>afa2f142-54c1-490d-bdf5-2781b0938754</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>afa2f142-54c1-490d-bdf5-2781b0938754</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:37:43.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>afa2f142-54c1-490d-bdf5-2781b0938754</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>afa2f142-54c1-490d-bdf5-2781b0938754</JobID>
  <Status>Succeeded</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion>2016-10-27T15:37:06.000-02:30</EstimatedCompletion>
  <NextPoll>2016-10-27T15:37:43.000-02:30</NextPoll>
  <PercentCompleted>100</PercentCompleted>
</StatusInfo>
<wps:GetResult
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>afa2f142-54c1-490d-bdf5-2781b0938754</wps:JobID>
</wps:GetResult>
<Result
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

```



```

<JobID>afa2f142-54c1-490d-bdf5-2781b0938754</JobID>
<ExpirationDate xsi:nil="true"/>
<Output>
  <Data mimeType="text/plain">238</Data>
</Output>
</Result>
<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0" response="document" mode="async">
  <ows:Identifier>CreateDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE">
    <wps:Input id="DATA">
      <wps:Reference
xlink:href="http://basemap.nationalmap.gov/arcgis/rest/services/USGSShadedReliefOnly/M
apServer/WMTS/1.0.0/WMTSCapabilities.xml"></wps:Reference>
      </wps:Input>
    <wps:Input id="STYLE">
      <wps:Data>GoogleMapsCompatible</wps:Data>
    </wps:Input>
    <wps:Input id="TYPE">
      <wps:Data>WMTS</wps:Data>
    </wps:Input>
    <wps:Input id="MIN_SCALE">
      <wps:Data>10</wps:Data>
    </wps:Input>
    <wps:Input id="MAX_SCALE">
      <wps:Data>15</wps:Data>
    </wps:Input>
    <wps:Input id="IDENTIFIER">
      <wps:Data>USGSShadedReliefOnly</wps:Data>
    </wps:Input>
    <wps:Input id="MIME_TYPE">
      <wps:Data>image/jpg</wps:Data>
    </wps:Input>
    <wps:Input id="NAME">
      <wps:Data>USGSShadedReliefOnly</wps:Data>
    </wps:Input>
    <wps:Input id="DESCRIPTION">
      <wps:Data>Created WMTS</wps:Data>
    </wps:Input>
    <wps:Input id="SRS">
      <wps:Data>EPSG:3857</wps:Data>
    </wps:Input>
  </wps:Input>
  <wps:Output id="DATASOURCE_IDS"></wps:Output>
</wps:Execute>
<StatusInfo

```

```

xmlns="http://www.opengis.net/wps/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<JobID>42b69753-1871-45f4-a014-69b21a2d47c9</JobID>
<Status>Accepted</Status>
<ExpirationDate xsi:nil="true"/>
<EstimatedCompletion xsi:nil="true"/>
<NextPoll>2016-10-27T15:38:08.000-02:30</NextPoll>
<PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>42b69753-1871-45f4-a014-69b21a2d47c9</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<JobID>42b69753-1871-45f4-a014-69b21a2d47c9</JobID>
<Status>Running</Status>
<ExpirationDate xsi:nil="true"/>
<EstimatedCompletion xsi:nil="true"/>
<NextPoll>2016-10-27T15:38:08.000-02:30</NextPoll>
<PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>42b69753-1871-45f4-a014-69b21a2d47c9</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<JobID>42b69753-1871-45f4-a014-69b21a2d47c9</JobID>
<Status>Running</Status>
<ExpirationDate xsi:nil="true"/>
<EstimatedCompletion xsi:nil="true"/>
<NextPoll>2016-10-27T15:38:08.000-02:30</NextPoll>
<PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>42b69753-1871-45f4-a014-69b21a2d47c9</wps:JobID>
</wps:GetStatus>
<StatusInfo

```

```

xmlns="http://www.opengis.net/wps/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<JobID>42b69753-1871-45f4-a014-69b21a2d47c9</JobID>
<Status>Succeeded</Status>
<ExpirationDate xsi:nil="true"/>
<EstimatedCompletion>2016-10-27T15:37:13.000-02:30</EstimatedCompletion>
<NextPoll>2016-10-27T15:38:08.000-02:30</NextPoll>
<PercentCompleted>100</PercentCompleted>
</StatusInfo>
<wps:GetResult
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>42b69753-1871-45f4-a014-69b21a2d47c9</wps:JobID>
</wps:GetResult>
<Result
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<JobID>42b69753-1871-45f4-a014-69b21a2d47c9</JobID>
<ExpirationDate xsi:nil="true"/>
<Output>
  <Data mimeType="text/plain">239</Data>
</Output>
</Result>
<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0" response="document" mode="async">
  <ows:Identifier>CreateDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE">
    <wps:Input id="DATA">
      <wps:Reference
xlink:href="http://107.20.228.18/arcgis/services/Wetlands/MapServer/WMServer?request=
GetCapabilities&service=WMS"></wps:Reference>
      </wps:Input>
    <wps:Input id="STYLE">
      <wps>Data>default</wps>Data>
    </wps:Input>
    <wps:Input id="TYPE">
      <wps>Data>WMS</wps>Data>
    </wps:Input>
    <wps:Input id="MIN_SCALE">
      <wps>Data>10</wps>Data>
    </wps:Input>
    <wps:Input id="MAX_SCALE">
      <wps>Data>15</wps>Data>
    </wps:Input>

```

```

    <wps:Input id="WIDTH">
      <wps:Data>256</wps:Data>
    </wps:Input>
    <wps:Input id="HEIGHT">
      <wps:Data>256</wps:Data>
    </wps:Input>
    <wps:Input id="IDENTIFIER">
      <wps:Data>1</wps:Data>
    </wps:Input>
    <wps:Input id="MIME_TYPE">
      <wps:Data>image/jpg</wps:Data>
    </wps:Input>
    <wps:Input id="NAME">
      <wps:Data>Wetlands</wps:Data>
    </wps:Input>
    <wps:Input id="DESCRIPTION">
      <wps:Data>Created WMS</wps:Data>
    </wps:Input>
    <wps:Input id="SRS">
      <wps:Data>EPSG:3857</wps:Data>
    </wps:Input>
  </wps:Input>
  <wps:Output id="DATASOURCE_IDS"></wps:Output>
</wps:Execute>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>f8d027c4-608b-41d1-9f07-ef1eae72fd7d</JobID>
  <Status>Accepted</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:18.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>f8d027c4-608b-41d1-9f07-ef1eae72fd7d</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>f8d027c4-608b-41d1-9f07-ef1eae72fd7d</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:18.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>

```

```

<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
version="2.0.0">
  <wps:JobID>f8d027c4-608b-41d1-9f07-ef1eae72fd7d</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>f8d027c4-608b-41d1-9f07-ef1eae72fd7d</JobID>
  <Status>Succeeded</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion>2016-10-27T15:37:18.000-02:30</EstimatedCompletion>
  <NextPoll>2016-10-27T15:38:18.000-02:30</NextPoll>
  <PercentCompleted>100</PercentCompleted>
</StatusInfo>
<wps:GetResult
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
version="2.0.0">
  <wps:JobID>f8d027c4-608b-41d1-9f07-ef1eae72fd7d</wps:JobID>
</wps:GetResult>
<Result
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>f8d027c4-608b-41d1-9f07-ef1eae72fd7d</JobID>
  <ExpirationDate xsi:nil="true"/>
  <Output>
    <Data mimeType="text/plain">240</Data>
  </Output>
</Result>
<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
version="2.0.0" response="document" mode="async">
  <ows:Identifier>CreateDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE">
    <wps:Input id="DATA">
      <wps:Reference
xlink:href="http://isse.cr.usgs.gov/arcgis/services/Orthoimagery/USGS_EROS_Ortho/Image
Server/WMSServer?request=GetCapabilities&service=WMS"></wps:Reference>
    </wps:Input>
    <wps:Input id="STYLE">
      <wps>Data>default</wps>Data>
    </wps:Input>
    <wps:Input id="TYPE">

```

```

        <wps:Data>WMS</wps:Data>
    </wps:Input>
    <wps:Input id="MIN_SCALE">
        <wps:Data>10</wps:Data>
    </wps:Input>
    <wps:Input id="MAX_SCALE">
        <wps:Data>15</wps:Data>
    </wps:Input>
    <wps:Input id="WIDTH">
        <wps:Data>256</wps:Data>
    </wps:Input>
    <wps:Input id="HEIGHT">
        <wps:Data>256</wps:Data>
    </wps:Input>
    <wps:Input id="IDENTIFIER">
        <wps:Data>0</wps:Data>
    </wps:Input>
    <wps:Input id="MIME_TYPE">
        <wps:Data>image/jpg</wps:Data>
    </wps:Input>
    <wps:Input id="NAME">
        <wps:Data>Imagery</wps:Data>
    </wps:Input>
    <wps:Input id="DESCRIPTION">
        <wps:Data>Created WMS</wps:Data>
    </wps:Input>
    <wps:Input id="SRS">
        <wps:Data>EPSG:3857</wps:Data>
    </wps:Input>
</wps:Input>
    <wps:Output id="DATASOURCE_IDS"></wps:Output>
</wps:Execute>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d4b2705e-bffe-4b2b-8b52-3ba6b6db1913</JobID>
  <Status>Accepted</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:23.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d4b2705e-bffe-4b2b-8b52-3ba6b6db1913</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"

```

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<JobID>d4b2705e-bffe-4b2b-8b52-3ba6b6db1913</JobID>
<Status>Running</Status>
<ExpirationDate xsi:nil="true"/>
<EstimatedCompletion xsi:nil="true"/>
<NextPoll>2016-10-27T15:38:23.000-02:30</NextPoll>
<PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d4b2705e-bffe-4b2b-8b52-3ba6b6db1913</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<JobID>d4b2705e-bffe-4b2b-8b52-3ba6b6db1913</JobID>
<Status>Succeeded</Status>
<ExpirationDate xsi:nil="true"/>
<EstimatedCompletion>2016-10-27T15:37:23.000-02:30</EstimatedCompletion>
<NextPoll>2016-10-27T15:38:23.000-02:30</NextPoll>
<PercentCompleted>100</PercentCompleted>
</StatusInfo>
<wps:GetResult
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d4b2705e-bffe-4b2b-8b52-3ba6b6db1913</wps:JobID>
</wps:GetResult>
<Result
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<JobID>d4b2705e-bffe-4b2b-8b52-3ba6b6db1913</JobID>
<ExpirationDate xsi:nil="true"/>
<Output>
  <Data mimeType="text/plain">241</Data>
</Output>
</Result>
<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0" response="document" mode="async">
  <ows:Identifier>CreateDatasource</ows:Identifier>
  <wps:Input id="DATASOURCE">
    <wps:Input id="DATA">

```

```

        <wps:Reference xlink:href="https://prd-
tnm.s3.amazonaws.com/StagedProducts/Elevation/19/IMG/ned19_n38x25_w122x25_ca_sanfranci
sco_topobathy_2010.zip#"></wps:Reference>
    </wps:Input>
    <wps:Input id="STYLE">
        <wps:Data>Float32</wps:Data>
    </wps:Input>
    <wps:Input id="TYPE">
        <wps:Data>ELEVATION</wps:Data>
    </wps:Input>
    <wps:Input id="MIME_TYPE">
        <wps:Data>GTiff</wps:Data>
    </wps:Input>
    <wps:Input id="NAME">
        <wps:Data>https://prd-
tnm.s3.amazonaws.com/StagedProducts/Elevation/19/IMG/ned19_n38x25_w122x25_ca_sanfranci
sco_topobathy_2010.zip</wps:Data>
    </wps:Input>
    <wps:Input id="DESCRIPTION">
        <wps:Data>Created Elevation</wps:Data>
    </wps:Input>
    <wps:Input id="SRS">
        <wps:Data>EPSG:3857</wps:Data>
    </wps:Input>
</wps:Input>
    <wps:Output id="DATASOURCE_IDS"></wps:Output>
</wps:Execute>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Accepted</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>

```



```

    <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
    <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>

```

```

    <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
    <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>

```

```

    <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
    <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Running</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <Status>Succeeded</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion>2016-10-27T15:38:11.000-02:30</EstimatedCompletion>
  <NextPoll>2016-10-27T15:38:31.000-02:30</NextPoll>
  <PercentCompleted>100</PercentCompleted>
</StatusInfo>
<wps:GetResult
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</wps:JobID>
</wps:GetResult>
<Result
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>d8d3c1ea-a591-4f80-b84e-533ab7166783</JobID>
  <ExpirationDate xsi:nil="true"/>
  <Output>
    <Data mimeType="text/plain">242</Data>

```

```

</Output>
</Result>
<wps:Execute
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:ows="http://www.opengis.net/ows/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0" response="document" mode="async">
  <ows:Identifier>CreateGeoPackage</ows:Identifier>
  <wps:Input id="INPUT_GEOPACKAGE">
    <wps>Data>{ "title": "TNM Combined", "descr": "TNM Combined from WPS", "geom":
    "POLYGON((-122.25 38.125,-122.125 38.125,-122.125 38,-122.25 38,-122.25 38.125))#",
    "ds_ids": "238,239,240,241,242", "portfolioIds": "154", } </wps>Data>
  </wps:Input>
  <wps:Output id="GENERATION_STATUS"></wps:Output>
</wps:Execute>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>069075dc-677e-4f26-9218-d0df640f3657</JobID>
  <Status>Accepted</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:39:12.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>069075dc-677e-4f26-9218-d0df640f3657</wps:JobID>
</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>069075dc-677e-4f26-9218-d0df640f3657</JobID>
  <Status>Accepted</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion xsi:nil="true"/>
  <NextPoll>2016-10-27T15:39:12.000-02:30</NextPoll>
  <PercentCompleted>0</PercentCompleted>
</StatusInfo>
<!-- MANY MORE STATUS POLLS .....-->
<wps:GetStatus
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>069075dc-677e-4f26-9218-d0df640f3657</wps:JobID>

```

```

</wps:GetStatus>
<StatusInfo
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>069075dc-677e-4f26-9218-d0df640f3657</JobID>
  <Status>Succeeded</Status>
  <ExpirationDate xsi:nil="true"/>
  <EstimatedCompletion>2016-10-27T15:58:33.000-02:30</EstimatedCompletion>
  <NextPoll>2016-10-27T15:39:12.000-02:30</NextPoll>
  <PercentCompleted>100</PercentCompleted>
</StatusInfo>
<wps:GetResult
  xmlns:wps="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd" service="WPS"
  version="2.0.0">
  <wps:JobID>069075dc-677e-4f26-9218-d0df640f3657</wps:JobID>
</wps:GetResult>
<Result
  xmlns="http://www.opengis.net/wps/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <JobID>069075dc-677e-4f26-9218-d0df640f3657</JobID>
  <ExpirationDate xsi:nil="true"/>
  <Output>
    <Data mimeType="text/plain">http://ogc-
tb12fo.compusult.net/wes/gpkg/gp/1HhN7nt4_NKsX_lG5q_nbvt_Qb18RBZE65o1</Data>
  </Output>
</Result>

```

Appendix E: Revision History

Table 15. Revision History

Date	Release	Editor	Primary clauses modified	Descriptions
Apr 15, 2016	.1	R. Cass	all	initial version
Jun 30, 2016	.2	R. Cass	all	initial draft
July 25, 2016	.3	R. Cass	all	2nd draft
Sept 30, 2016	.4	R. Cass	all	3rd draft
Oct 31, 2016	.5	R. Cass	all	Implemented recommendations from Scott Serich, Jeff yutzler, improvements in grammar, typos, updated screenshots, fleshed out WPS description, appendix A content
Nov 1, 2016	.6	R. Cass	all	Implemented recommendations from Carl Reed

Date	Release	Editor	Primary clauses modified	Descriptions
Nov 11, 2016	.7	R. Cass	all	<p>Implemented recommendations from Kristin Fishburn</p> <p>[appendix] == Bibliography - [1] Erwan Bocher, Olivier Ertz: Towards Cartographic Portrayal Interoperability – the Revision of OGC Symbology Encoding Standard. 1st ICA European Symposium on Cartography, Nov 2015, Vienne, Austria. 1 (1), pp.116-119, (2015). - [2] OFFICE OF GEOMATICS, NATIONAL GEOSPATIAL-INTELLIGENCE AGENCY (NGA) STANDARDIZATION DOCUMENT :</p> <p>NGA.SIG.0011_1.0_0_WEBMERC Implementation Practice Web Mercator Map Projection. (2014) - [3] Mapnik configuration XML</p> <p>https://github.com/mapnik/mapnik/wiki/SymbologySupport (2016)</p>