Guidance for Submitting Crowdsourced Bathymetry Data to the IHO Data Center for Digital Bathymetry

Background

This document is intended to provide guidance on how to implement data transfers between CSB data contributors (Trusted Nodes or individual contributors) and the IHO Data Centre for Digital Bathymetry (DCDB). Additional information and details, including optional and mandatory data and metadata fields, can be found in the IHO Publication B-12: Guidance on Crowdsourced Bathymetry.

The purpose of B-12 is to provide information and guidance to all stakeholders of CSB data. Prior to submitting data to the DCDB it is recommended that CSB data contributors review B-12, specifically:

Preparing to Submit Data

Those interested in contributing data or becoming a Trusted Node should begin by contacting the DCDB at bathydata@iho.int. During initial discussions with the DCDB the following points will be addressed:

- The DCDB will review an **example file** from the new CSB data contributor to ensure data types and formatting are in agreement with B-12 requirements.
- The DCDB and CSB data contributor will discuss the **uniqueID** to be used in the metadata and data transmission. This unique value is composed of a prefix identifying the contributor followed by a hyphen and a UUID. The prefix generally consists of up to six upper case characters and is determined in collaboration with the DCDB. It allows users to locate their submissions on map viewers (eg: DCDB Bathymetry Viewer). The UUID portion of the uniqueID is a standardized unique number and should be generated by the CSB data contributor. This allows for the vessel to be anonymous on map viewers and in the archive if desired. Generally there is one UUID per vessel, consistent for all submissions from that vessel, although other conventions can be discussed with the DCDB. Basic information regarding UUIDs can be found in this Wikipedia article. Several examples demonstrating the generation of a UUID are available on GitHub.
- Once a file has been reviewed and the uniqueID has been determined, the DCDB will provide an **authentication token**. This token is required in order to upload data.

After the above points have been addressed, a CSB data contributor can submit CSB data files and metadata to the DCDB via a public application programming interface (API). Note that submission will fail without a valid uniqueID and authentication token.

Introduction to cURL

The recommended method to upload data to the API is by utilizing the command-line tool, cURL, within a command prompt to make an HTTP POST Request. Data can be uploaded in GeoJSON or XYZ format. Details on the submission of each format type can be found below.

cURL is a command-line tool that comes preinstalled on a computer's OS. With the use of cURL, HTTPS requests can be made directly from an OS command prompt. To read more about cURL, view the official documentation here. Windows users can find additional installation information at https:/curl.se/windows.

Submitting XYZ File Data with cURL

Prior to submission, the contents of the XYZ file should be consistent with the formatting demonstrated below. Note that in the case of an XYZ file submission, the file itself does not contain any metadata. Metadata is in JSON format and located within the command prompt rather than within the XYZ file.

```
LON,LAT,DEPTH,TIME
38.871667, 1.294000, 9.5, 2014-09-14T21:00:01.000Z
38.871667, 1.294000, 9.4, 2014-09-14T21:00:02.000Z
38.871667, 1.294000, 9.4, 2014-09-14T21:00:03.000Z
```

The cURL request below should be run from the directory that contains the file to be submitted. Alternatively, the absolute path for the file can be used.

In this cURL request, the *uniqueID* should replace the given **<uniqueID>** parameters.

The *authentication token* provided by the DCDB should replace the **<valid token>** parameters.

The *file name* should replace **<example.xyz>** and should be located in the same directory. Alternatively, the absolute path for the file can be provided.

Note that *metadata* is in JSON format and located within the command prompt rather than the XYZ file.

The example demonstrates the minimum required metadata for submission to the DCDB. Additional information about the vessel, sensors, sensor installation, and processing allows data users to assess the quality of the data, apply corrections, etc. This greatly increases the potential applications of the data and should be added whenever possible. Additional information regarding optional metadata fields can be found in Section 3 of B-12.

```
$ curl 'https://www.ngdc.noaa.gov/ingest-external/upload/csb/test/xyz' -i -X POST \
   -H 'x-auth-token: <valid token>' \
   -H 'Content-Type: multipart/form-data' \
   -F 'file=@<example.xyz>;type=multipart/form-data' \
   -F 'metadataInput={
   "crs": {
      "horizontal": {
         "type": "EPSG",
         "value": 4326
      },
      "vertical": "Transducer"
   },
   "providerContactPoint": {
      "orgName": "Example Cruises Inc",
      "email": "support@example.com",
      "logger": "Rose Point ECS",
      "loggerVersion": "1.0"
   },
   "convention": "XYZ CSB 3.0",
   "dataLicense": "CC0 1.0",
   "platform": {
      "uniqueID": "<uniqueID>",
      "correctors": {
         "positionReferencePoint": "Transducer",
      }
   }
}'
```

Submitting GeoJSON File Data with cURL

Prior to submission, the contents of the GeoJSON file should be consistent with the formatting demonstrated below. In the case of GeoJSON submission, the metadata is included within the file itself rather than in the command prompt.

This example file demonstrates the minimum required metadata for submission to the DCDB. Additional information about the vessel, sensors, sensor installation, and any processing allows data users to assess the quality of the data, apply corrections, etc. This greatly increases the potential applications of the data and should be added whenever possible. Additional information regarding optional metadata fields can be found in Section 3 of B-12.

```
{
 "type": "FeatureCollection",
 "crs": {
          "horizontal": {
                    "type": "EPSG",
                    "value":4326
          },
          "vertical": "Transducer"
```

```
 },
 "properties": {
        "providerContactPoint": {
                "orgName": "Example Cruises Inc",
                 "email": "support@example.com",
                "logger": "Rose Point ECS",
                "loggerVersion": "1.0"
        },
        "convention": "GeoJSON CSB 3.0",
        "dataLicense": "CC0 1.0",
        "platform": {
                 "uniqueID":  "EXAMPLE-f8c469f8-df38-11e5-b86d-9a79f06e9478",
                "correctors": {
                        "positionReferencePoint": "GNSS",
                }
        }
  },
  "features": [
        {
                "type": "Feature",
                "geometry": {
                        "type": "Point",
                        "coordinates": [
                                -119.37249833123325,
                                72.09587373217158
                        ]
                },
                "properties": {
                        "depth": 108.9,
                        "time": "2015-08-06T22:00:00.000Z"
                }
        }
  ]
}
```

The cURL request below should be run from the directory that contains the file to be submitted. Alternatively, the absolute path for the file can be used.

In this cURL request, the *uniqueID* should replace the given **<uniqueID>** parameter. The uniqueID in the header must match the uniqueID in the record.

The *authentication token* provided by the DCDB should replace the **<valid token>** parameters.

The *file name* should replace "**geoJson.json**" and should be located in the same directory. Alternatively, the absolute path for the file can be provided.

Note that in this case the metadata is included within the file rather than in the POST request.

```
$ curl 'https://www.ngdc.noaa.gov/ingest-external/upload/csb/test/geojson' -i -X POST \
    -H 'x-auth-token: <valid token>' \
    -H 'Content-Type: multipart/form-data' \
    -F 'file=@<geoJson.json>;type=multipart/form-data' \
    -F 'metadataInput={
    "uniqueID": "<uniqueID>"
}'
```

Submission Response

Below is the expected response if the data is successfully posted.

```
HTTP/1.1 201 Created
Content-Length: 75
Content-Type: application/json;charset=UTF-8

{"message":"Submission successful.","submissionIds":["123"],"success":true}
```

For questions regarding the submission of crowdsourced bathymetry data to the IHO DCDB, please contact bathydata@iho.int. If possible, please reference the submission ID listed in the submission response.