# Darwin Core Archive Assistant

## User Guide

**Version 1.1**

**April 2011**

Complete user guide to the Darwin Core Archive Assistant. This service supports the publication of biodiversity data using the Darwin Core Archive format. It supports the creation of an XML data descriptor file.

**Suggested citation:** GBIF (2010). Darwin Core Archive Assistant, version 1.1, released on 1 April 2011, (contributed by Remsen, D, Sood, R.), Copenhagen: Global Biodiversity Information Facility, 23 pp,.

**Persistent URI:** http://links.gbif.org/gbif_dwc-a_guide_en_v1.1

**ISBN**: 87-92020-21-6

**Copyright © Global Biodiversity Information Facility, 2010**

**Language: English**

**License:**

This document is licensed under a Creative Commons Attribution 3.0 Unported License

**Document Control:**

| Version | Description | Date of release | Author(s) |
|---|---|---|---|
| 1.0 | Initial Draft and Proofing | 1 Dec 2010 | David Remsen |
| 1.1 | Consolidated update | 1 April 2011 | David Remsen |

**Cover Art: Greg Basco**

Chestnut-mandibled toucan, *Rhamphastos swainsonii*

## About GBIF

The Global Biodiversity Information Facility (GBIF) was established as a global mega-science initiative to address one of the great challenges of the 21st century – harnessing knowledge of the Earth's biological diversity. GBIF envisions 'a world in which biodiversity information is freely and universally available for science, society, and a sustainable future'. GBIF's mission is to be the foremost global resource for biodiversity information, and engender smart solutions for environmental and human well-being . To achieve this mission, GBIF encourages a wide variety of data publishers across the globe to discover and publish data through its network.

# Table of contents:

# About this user guide

The purpose of this user guide is to explain how to use the **Darwin Core Archive Assistant** tool in order to facilitate biodiversity data publishing using the Darwin Core Archive format. This format utilizes text files as the basis for data exchange and is simple enough that biodiversity data can be transformed to this format without any locally installed software.

This tool assists data managers in the creation of an XML document that may be required to be included in a DarwinCore Archive, known as a *metafile*, for two typical data management configurations:

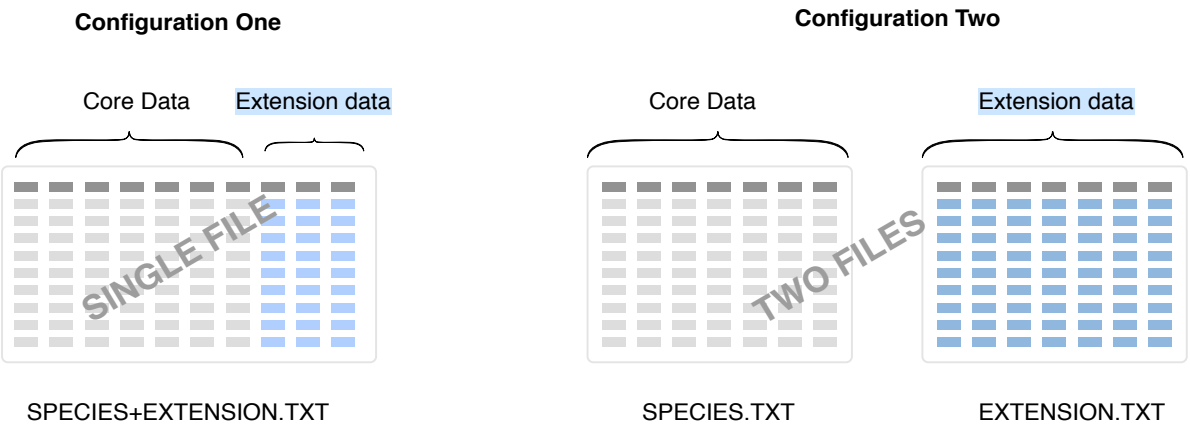


Figure 1 - User Data configurations used in this guide

- Configuration One: Scientists (biologists) who manage biodiversity data using spreadsheets such as **MS Excel®**.
    - Documentation for this user configuration will focus on creating a metafile for a single data file that may contain data corresponding to, both, a core data file and one or more extensions.

- Configuration Two: Database administrators who manage biodiversity data using relational databases such as **MS Access®**, **FileMaker®**, **SQL Server®**, **MySQL®**, etc.

- Documentation for this user configuration will focus on creating a metafile for multiple data files where the admin is able to export files that correspond to distinct core and extension files.

Note: The tool described in this guide helps to produce a properly formatted DarwinCore Archive. It does not, however, register the resultant file to the GBIF network. To register a DarwinCore Archive file with GBIF, please see the document,

Registering Your Dataset with GBIF, a How-to Guide
http://links.gbif.org/gbif_how_to_register_guide_en_v1

# Introduction – the Darwin Core Archive Format

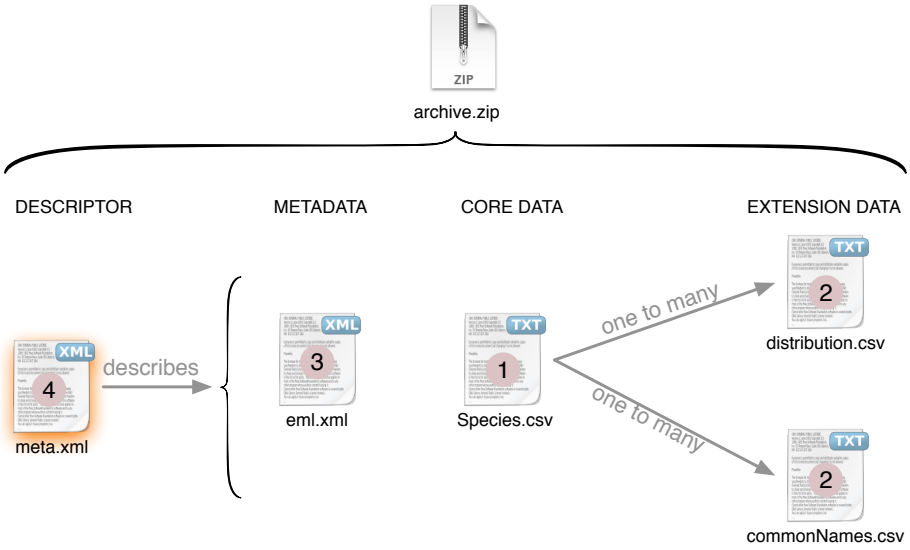


Figure 2 – DarwinCore Archive parts

## *Darwin Core Archive Parts*

A Darwin Core Archive (Figure 2) is a set of text files, typically grouped into a single compressed file, an *archive*, for portability using a compression format such as Zip[1] or GZIP[2]. The set of files in an archive may vary but consists of the following:

A required **core data** file consisting of a standard set of DarwinCore terms.
A data file is formatted as *fielded text*, where data records are expressed as rows of text, and data fields (columns) are separated with a standard delimiter such as a tab or comma

[1] http://en.wikipedia.org/wiki/ZIP_(file_format)

[2] http://en.wikipedia.org/wiki/Gzip

(commonly referred to as CSV or 'comma-separated value' files). Many users are familiar working with data in spreadsheets and understand the basic concept behind fielded text. In a DarwinCore data file the first row of the file *may* contain the names of the DarwinCore terms represented in the succeeding rows of data.

A row in a DarwinCore Archive data file *must* represent one of two biodiversity data types: a distinct taxon such as a species or a specific *occurrence* of a taxon such as a specimen. In general, this requires the inclusion of a unique identifier in the core data file, known as the *core id*.
The Darwin Core text guidelines support the use of **optional "extensions"**.
An extension is a linked data file that contains additional non-DarwinCore terms that 'extend' the core set of terms and enable a more enriched set of data to be shared by a user. Extensions are defined and registered with the GBIF registry so they are 'known' to the network. A row in an extension file always includes a reference to the core id corresponding to a taxon or a taxon occurrence in the core data file. Multiple rows in an extension may refer to the same core data record.

A simple example is a vernacular names extension that supports the description of common name properties that might be related to a species described in the core data file. The extension allows such properties as the language, location, and source of the common name to be expressed while also supporting the publication of multiple common names for a single taxon as distinct rows in the extension file. The overall topology of one or more of these extensions to the core table is referred to as a "star schema".

The DarwinCore Archive format relies on a special file – an XML descriptor file, called the "*metafile*" (typically named *meta.xml*). The metafile is used as a map to describe the core taxon file and any extensions that collectively form the specific data profile that will be produced by the user.

An **optional metadata document** is used to describe the overall resource in the archive. It provides details such as a title for the resource, contact information and general scope

and description. GBIF supports multiple metadata formats that includes a GBIF-specific profile expressed in the Ecological Metadata Language (EML).

> ***Composing this XML descriptor file (#3) is the function of the DarwinCore Archive Assistant.***

## Getting started – The parts of the web application

The **DarwinCore Archive Assistant** is an open source application and the source code is freely available for anyone wishing to access it. The most recent version will always be located at http://tools.gbif.org/dwca-assistant/. Most users do not need to install their own version of the application and should use the version at this address.

The source code and technical documents can be found on the project site at http://code.google.com/p/gbif-meta-maker/. 4

### *Screen Layout*

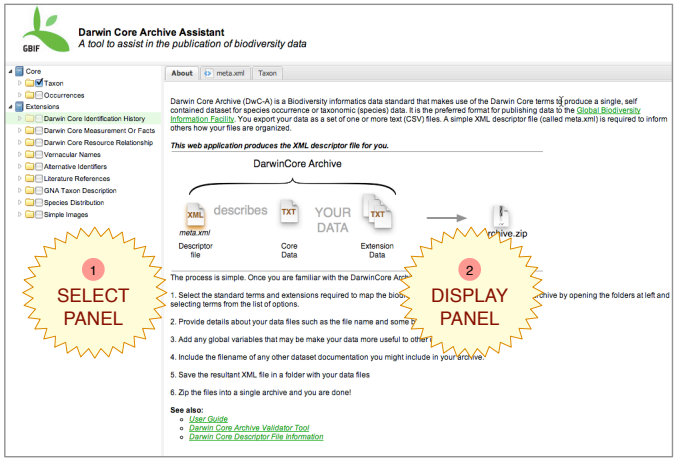


Figure 3 - The startup view with the "About" page displayed

When accessing the Darwin Core Archive Assistant, note that, aside from the header information, the screen is divided into two columns or panels (see Figure 3 above).

1. The left ***Select Panel*** **(1)** is for browsing and selecting data fields that will be published by the user. This panel displays a nested set of items corresponding to core and the extensions data elements. This panel is always displayed while the application is active.

2. The right ***Display Panel*** **(2)** occupies most of the application space and is for displaying pages of information. Pages are distinguished by tabs arranged horizontally on the screen. At startup, the user will notice three initial tabs: *About*, *meta.xml*, *Taxon (a core data file)*. The active tab is in lighter grey shades. The first two pages, "About" and "meta.xml" are always displayed and cannot be removed. The "Taxon" page may be replaced by the "Occurrences" or other core data file. The remaining numbers of pages are variable, based on selections made by the user.

   The *About* page provides the initial startup information and links to relevant information.

   The "meta.xml" page displays the primary output of the web application: a validated DarwinCore Archive descriptor file.

At startup, a basic XML document is presented (as can be seen in Figure 4 below). Most of the content of this page will be generated, depending on the settings and modifications made by the user. The XML generated in this panel can be selected, but is not directly editable. The panel provides buttons for saving and validating the generated metafile as well as an entry field for specifying the name or location of a metadata document.


Figure 4 – The meta.xml page

The third page, "Taxon", represents the default selection of core data file. Any remaining pages represent selections of extension data made by the user. A DarwinCore Archive requires a core file corresponding either to a core data type which currently are either a Taxon or a taxon Occurrence ("Occurrences"). Other extension pages can be created or deleted by checking the box beside the different folders and sub-folders from the left select panel (1).

While it is possible to uncheck the boxes from the *Extensions* folder, it is not possible to uncheck both *Taxon* and *Occurrences* – one of these two pages must always be selected and displayed.

## *Select Panel Details*

### Selecting core terms and extensions

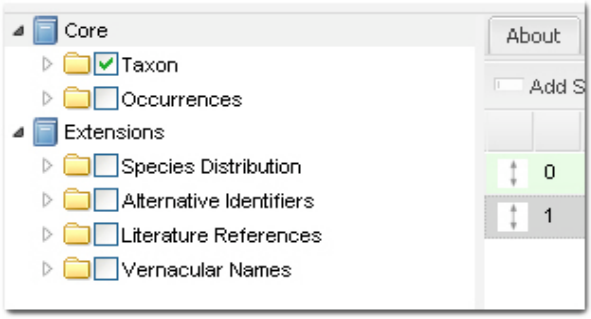


Figure 5 – Select panel display

As it can be seen in Figure 5- **Select panel display** above, there are two main "volumes": Core and Extensions, listed in the Select Panel. Clicking on the arrow beside each volume will open the menu and reveal a series of sub-folders. There are two sub-folders for *Core* files: *Taxon* and *Occurrences*; and there are sub-folders for *Extensions* files: *Vernacular Names*, *Alternative Identifiers*, *Literature References*, GNA Taxon Description and *Species Distribution*. The specific number of Extensions displayed may vary and depends on:

1) If the extension is associated with the selected core data type.
2) The number of extensions registered in the GBIF registry.

The process of selecting data fields for both core data types or for extensions is exactly the same: the user selects a folder in the Select Panel to display a list of terms.

A complete list of extensions, terms and definitions can be found on the GBIF schema repository[3]. A quick reference guide may be found at: http://rs.tdwg.org/dwc/terms/.

A short description of a term from the select panel appears in a pop-up window when the user hovers over it. The description may be accompanied by some examples, as it can be seen in Figure 6 below.

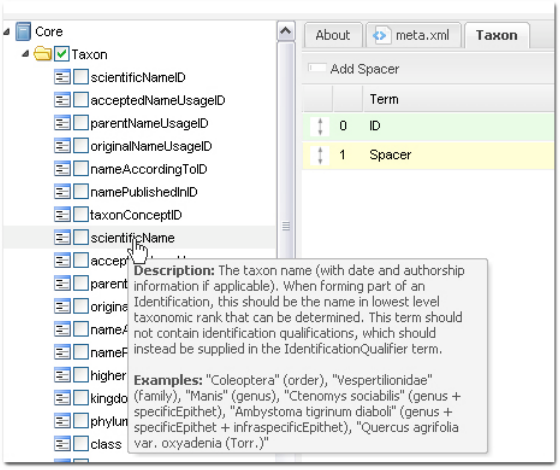


Figure 6 – Mouse over a term to view more information

[3] http://rs.gbif.org/

### *File Settings*

The application provides the following file setting options:

1. CSV File
2. Tab Separated File
3. Custom Format

In addition to the above file formats, the application also allows to specify the following delimiters for the selected file setting:

1. Field Delimiter
2. Fields enclosed by
3. File Encoding
4. Ignore header row
5. Line ending

Users can select the appropriate value for the above from the drop down box.

## General workflow for using the DarwinCore Archive Assistant

To effectively use the application, the user should have a basic understanding of the DarwinCore Archive format and the scope of the core data and extensions available for mapping their source data. A user may choose to export one or more data files from a source first, and then specify the data fields in each file using the application second, or generate a profile and metafile first and use this as a guide to exporting one or more data files. Whether before or after launching the application, the user must save the published data file or files as text files, comma-separated or tab-separated output being recommended. Other delimiters are acceptable and can be described in custom file settings.

Find below the sequence of steps to be followed for generating a validated metafile:

1. Identify and select the core data type (Taxon or Occurrences)
2. Specify the name of the published data file.
3. Provide specific file formatting details. Select appropriate file options under File Settings.
4. Identify and select the core data fields. Expand the tree under Taxon or Occurrences and check the boxes to select the fields.
5. Add spacers if needed
6. Review and rearrange if needed the columns added in the core data file page layout.
7. Review and select relevant extensions. Expand the tree under selected extension to select fields. The process of selecting extensions is similar to the process of selecting the core data file, the only exception is that, while there can be only one core data file selected, the user may select all or none of the extensions. Extension pages can be created or deleted by checking the box beside the different folders and sub-folders from the left select panel.

For each selected extension, a new page will be created in the right display panel. The order of selecting the extension data files is not relevant for the final output.

## Publishing a DarwinCore Archive to GBIF

### *Mandatory Data Elements*

Some data fields are mandatory when selecting Core data types and Extensions. Mandatory data are required either to provide relational integrity between core data and an extension.

**The core ID**

The term "core ID" is a general descriptive name and does not refer to a specific DarwinCore term. The actual identifier used as the core ID is dependant on the core data type selected.

- When the core data type is Taxon, the core ID is the *taxonID*.
- When the core data type is Occurrence, the core ID is the *occurrenceID*.

A core ID is not required when:

- No extensions are used
- The first row (the header row) contains field names that exactly match the Darwin Core terms

A core ID is required when:

- One or more extensions are used in the publication of a data profile
- A data file does not contain a header row with column names that do not exactly match their corresponding DarwinCore term names.

**Other required fields**

Additional required terms will be indicated when an extension is selected. For example, the Vernacular Names Extension requires that a *vernacularName* term be among the elements published in the data file. Require fields are highlighted and are check marked under the "Required" tab.

**Source file name**

In the core data page, and in any extensions, the user must specify the name of the data file that will contain the selected fields.

The data file should be located in the same folder as the *metafile* so that no file path information is required. Filenames should follow typical file-naming conventions (e.g., no spaces, using hyphen/underscore if needed (example: "vernacular_names.tab")).

A data file name is required for the core data page and for *each* extension that is used to create a data publication profile. This is true even if all data (core+extensions) resides in the same file.

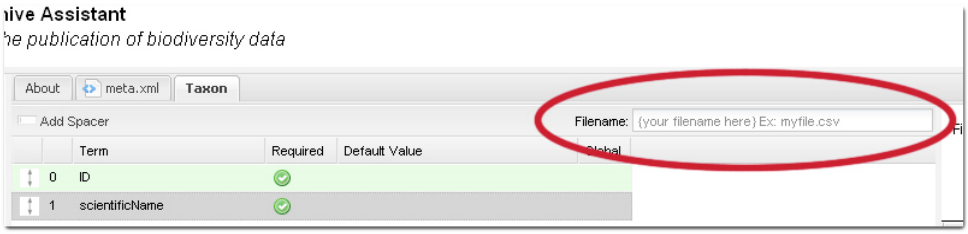


Figure 7 – Datafile name entry form

## Field Ordering

Fields are listed in the Display Panel in the order they are selected by the user in the Select Panel with required IDs listed first. The objective is for the order of data elements to match the field order in the corresponding data file. Note that numbering in the application starts with field 0 (zero) for the first field because this is how it will be referred in the metafile.

After selecting the data elements, the columns may be re-ordered, by a simple "drag and drop" action: (in the Figure 8 below the "occurrenceStatus" is dragged from the first row to the second).


Figure 8 – Re-ordering rows

A data element can be deleted from the display panel by unchecking the box beside it. When the box will be checked again, the selected element will appear the last in the displayed list.

### Add a blank “spacer” column

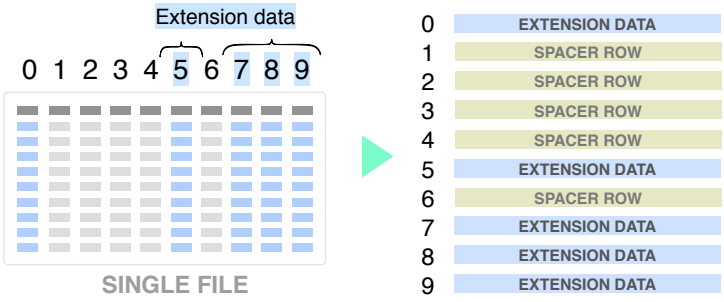


Figure 9 - Extension and core data in one file

It is possible to publish a single data file that contains elements that conform to both core and extension data. Since core and extension information is distinct in the application, each is processed separately and the columns that do not conform to the core or extension are skipped. In Figure 9, for example, extension data is located in columns 5, 7, 8 and 9. Since the application numbers fields according to their order in the list, columns 1-4 and column 6, which are mapped in the core, are skipped by adding blank or spacer rows, to the element list in the extension page. This allows the metafile to put the proper column number into the metafile when it is generated.

The user can add spacers by clicking on the “Add Spacer” button found in the header of the display panel (Figure 10).

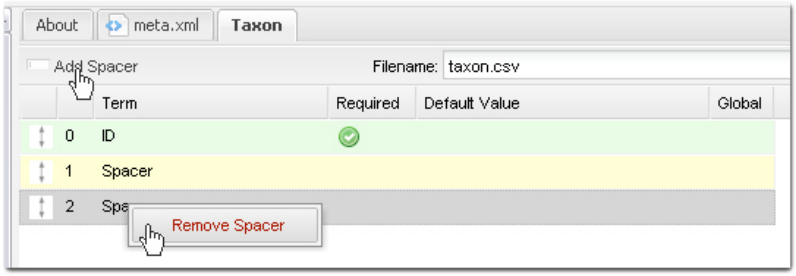


Figure 10 – Adding and removing spacers

The Spacer field may be moved by "drag-and-drop" action and it can be deleted by mouse right-click on the field (for Windows) and selecting "Remove Spacer" or mouse click to select the spacer on the screen - hold down Ctrl - and a "Remove Spacer" appears (for Mac). (Figure 10)

## Configuration One: Scientists (biologists) who manage biodiversity data using spreadsheets such as MS Excel®

Configuration One refers to scientists (biologists) who manage biodiversity data using spreadsheets such as **MS Excel®.** In this example, a single worksheet contains a data table that can be exported as a single text file, but contains data elements that correspond to both DarwinCore core elements and extension elements. From a DarwinCore Archive perspective, this single file must be described in the context of the separate core and extension components.

This configuration REQUIRES an ID, which may not be present in the source data at the onset but is needed if extensions are used.

A data file from an actual species list on a public government site provides a working example.[4] The information was copied in a spreadsheet and then saved as "*species.csv*" as it can be seen in Figure 11 below:

[4] The Massachusetts Division of Wildlife: Rare Species for town of Falmouth
*http://www.mass.gov/dfwele/dfw/nhesp/species_info/town_lists/town_f.htm#falmouth*

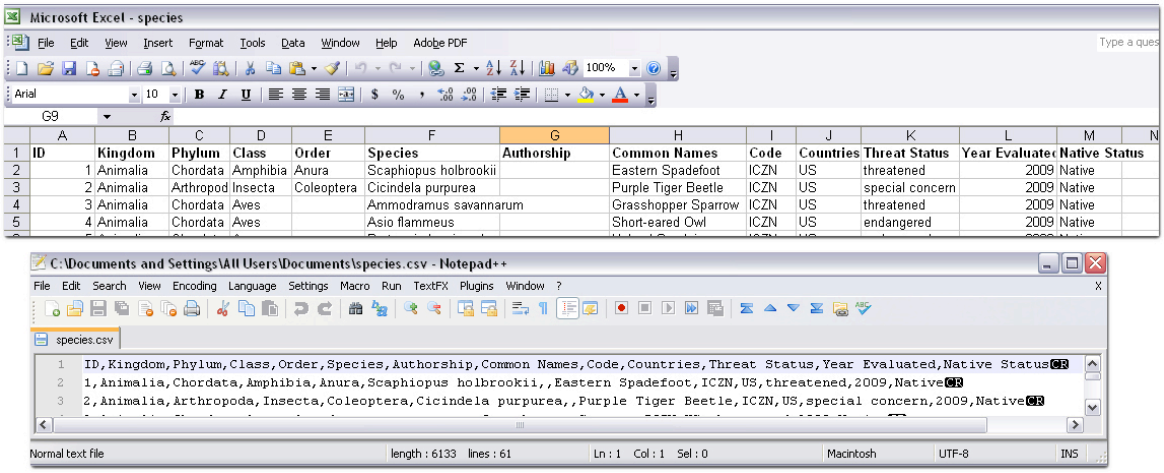


Figure 11 – A sample data file

Notice that the columns from this table are named in the first row and that these names sometimes, but often do not, match the element names in the core and extension definitions. The first step, therefore, is to review the data columns and identify those which can be matched to core or extension data elements.

In this case, the source data file, *species,csv* contains data elements that are associated with the taxon core data type and two extensions

- "Taxon" core data type
- "Vernacular Names" extension
- "Species Distribution" extension:

Documenting this set of data requires the creation and completion of the corresponding three pages in the Darwin Core Archive Assistant:

- Taxon core page,
- Vernacular Names extension page
- Species Distribution extension page

*Mapping the core data fields*

*This configuration requires a special attention paid to the number of columns and remember that column numbering starts with zero, not one.*

1. Select the Taxon core data volume from the left select display.
2. Register the file name (in the example, “species.csv”) in the header of the newly created page.
3. Record the file settings for the files, “species.csv”
4. Select the core terms elements in the select panel that match the data fields in the source data file.
    A. First column, column 0, is the ID and maps to the core ID
    B. The next columns are Kingdom, Phylum, Class, Order, ScientificName, ScientificNameAuthorship, nomenclaturalCode.
5. Note in Column “H” the spreadsheet in Figure 11 – A sample data file is labeled Vernacular Names and falls between the “Authorship” in Column G and “Code” in Column I. The vernacular name element is part of the Vernacular Names extension and is therefore mapped in the “Vernacular Names” page. In order to map the Code (Column I in the spreadsheet) properly in the metafile, a spacer must be added to “skip” the Vernacular Name column and account for the blank column.
    A. Click the Add Spacer button to add a spacer to the Display Panel, taxon page.
    B. Move the Spacer to the position between scientificNameAuthorship and nomenclaturalCode on the field list (index 7).
6. The final output of the Taxon page may be seen in Figure 12 below:

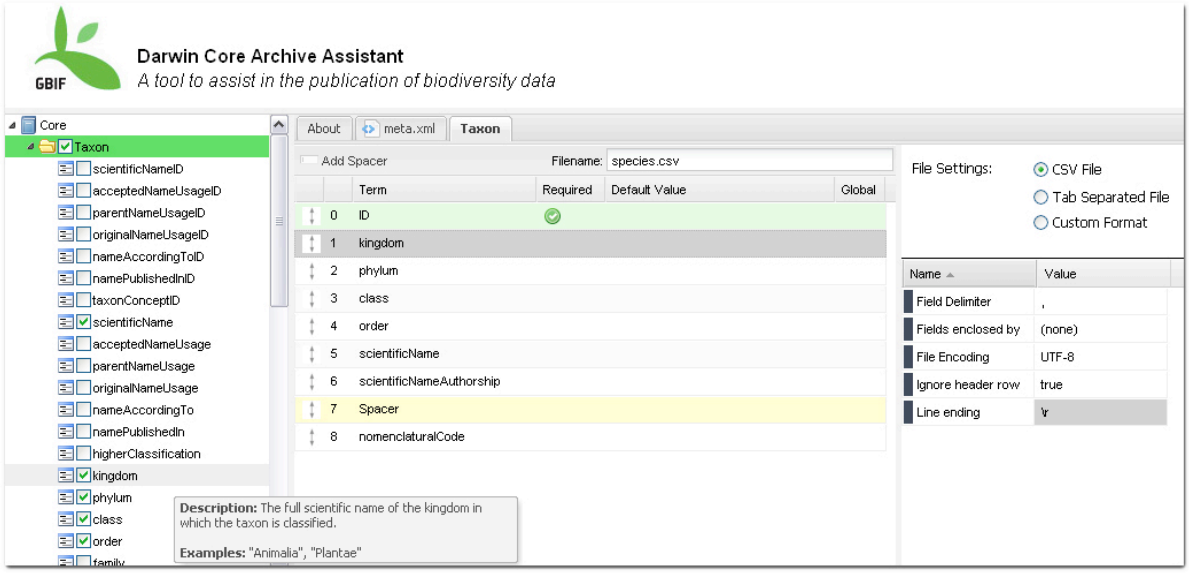


Figure 12 – The final output of the Taxon page

### *Mapping the extensions data fields*

In the example chosen for Configuration One, there are two extension data files: “Vernacular Names” and “Species Distribution”.

For each of the extensions, a new page will be created, by checking the box next to the respective volumes.

#### Generating the metafile for “Vernacular Names” extension

In this example, there is only one column belonging to this extension data file, “Common Names”, found in the eighth column in the spreadsheet.

7. Select the Vernacular extension from the left select display. Since the information for the Vernacular Names extension is found in the same data file as the information for the Taxon core data file, the file name for this page will be identical as the file name written in the Taxon page: “species.csv”. Note that in addition to the core ID, the vernacularName element is already included in the file, as it is required.

8. Register the file name (in the example, “species.csv”) in the header of the newly created page.
9. Record the same file settings for the files, “species.csv”
10. An **optional metadata document** is used to describe the overall resource in the archive.

    It provides details such as a title for the resource, contact information and general scope and description. GBIF supports multiple metadata formats that includes a GBIF-specific profile expressed in the Ecological Metadata Language (EML).
11. Refer back to the spreadsheet in Figure 11 – A sample data file. The vernacular names column occupied Column H, the eighth column in the spreadsheet. The first column is ID used as the core ID and represents, field 0, the core ID, on the Vernacular Names page in the Display panel. All the other columns up to Column H represent fields associated with the Taxon page we identified in the previous steps. The goal is to specify that one can find the matching vernacular name field in Column H, the eighth column. This means we need to add six (6) spacers to the page to assign the vernacular names element an index of seven (7).
12. Add the spacers by clicking the “Add Spacer” button six times.
13. Position the vernacularName element to the end of the list and ensure that is has an index ID of 7.
14. Refer back to the spreadsheet in Figure 11 – A sample data file. Note that the vernacular names in the source file are all in English, a piece of information not included in the source data but a property that may be of value to later users of the data, who may merge the data with data files originating in other languages, as an example. This information may be added as a global value, indicating that all vernacular names in the list may be assigned a language property with the value of “English”
    A. Select the “language” element from the vernacular extensions list in the Select Panel
    B. Click on the “Default Value” entry field in the corresponding row in the Display Panel.
    C. Enter “English”
    D. Click the Global checkbox to the right of the entry field

The final output of the “Vernacular Names” page may be seen in Figure 13 below:

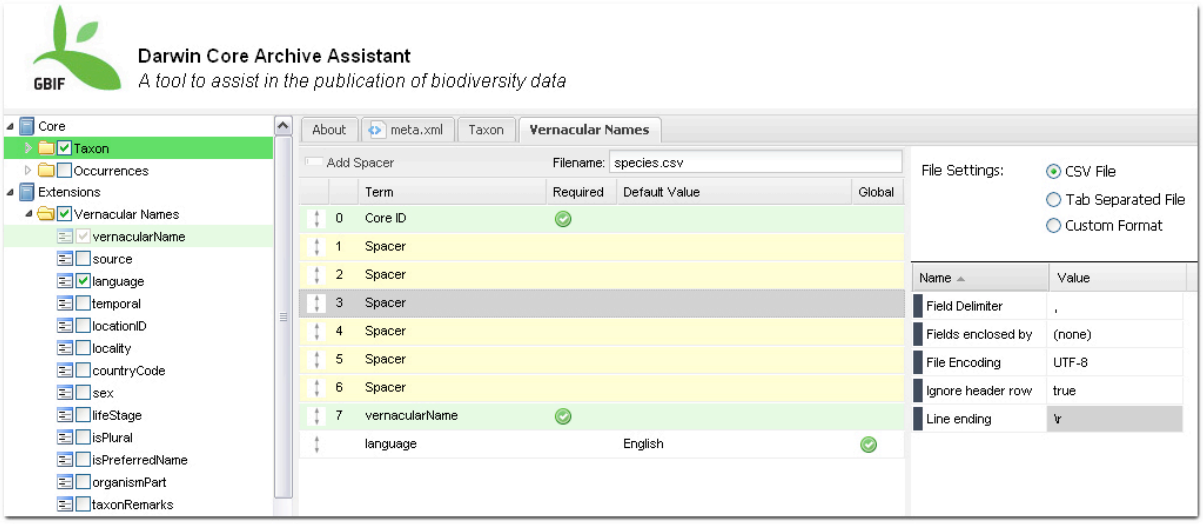


Figure 13 - The final output of the “Vernacular Names” page

The meta.xml will now also contain information about the extension data files:

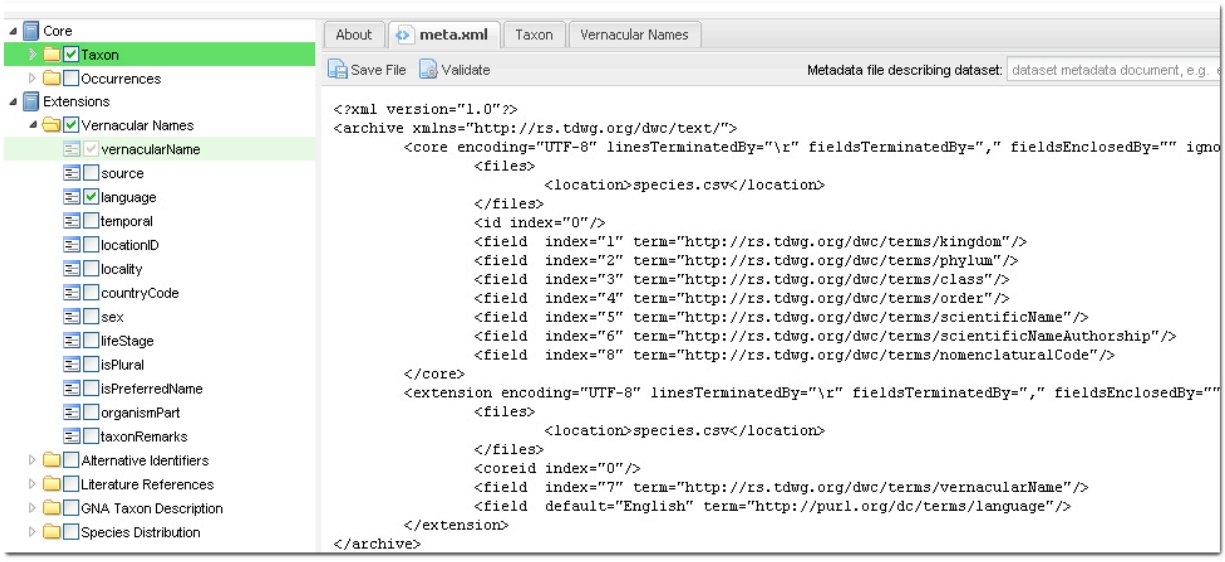


Figure 14 – The updated metafile. Note the extension data at the end of the file

### Generating the metafile for "Species Distribution" extension

The last 4 columns in "species.csv" data file labeled, Countries, Threat Status, Year of evaluation, Native Status. These columns will be mapped to data elements in the Species Distribution extension, specifically

| Column Name | Species Distribution element |
|---|---|
| Countries | Country |
| Threat Status | threatStatus |
| Year of Evaluation | eventDate |
| Native Status | establishmentMeans |

By default, "Core ID" column is selected, matching the "ID" column from the Taxon page and from the original spreadsheet.

After selecting the "Species Distribution" extension data file from the select panel, fill in the file name and make the file settings. As this information is found in the same data file as the taxon core data and the vernacular names extension data, these settings shall be the same.

15. Register the file name (in the example, "species.csv") in the header of the newly created page.
16. Record the same file settings for the files, "species.csv"

In this example, the definitions belonging to the "Species Distribution" are found in columns 10 to 13. In order to correctly map this information the user must insert spacers for the columns 2 to 10. This is done using the same procedures as in Steps 10-12 in the previous extension.

The final output of the Species Distribution may be seen in Figure 15 below

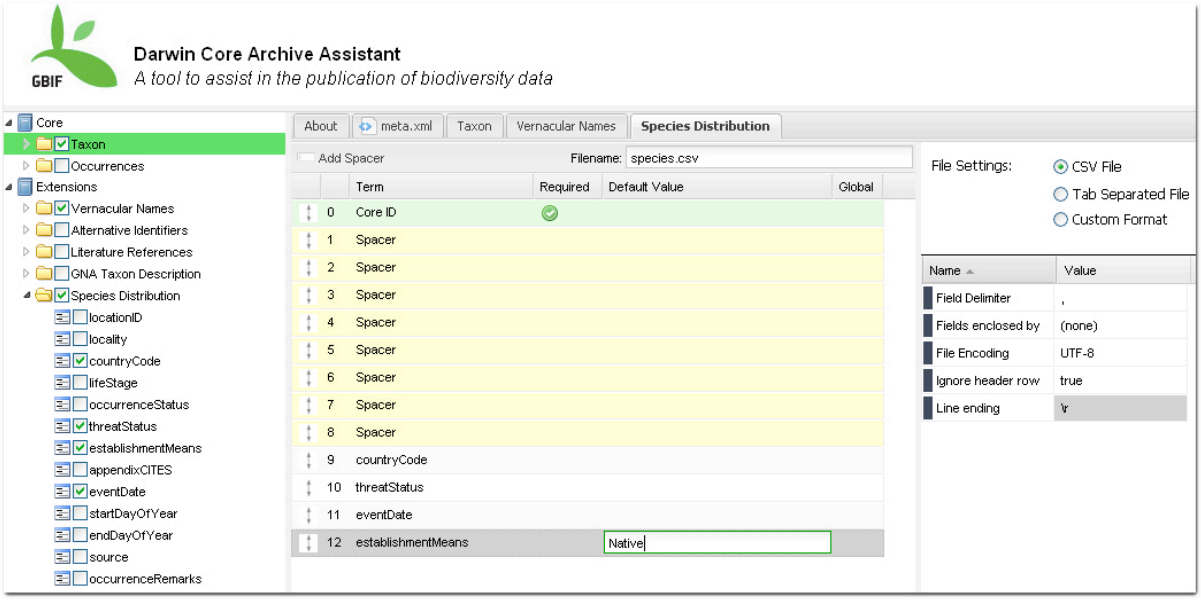


Figure 15 - The final output of the "Species Distribution" page

By writing "Native" in the "Default Value" for the "establishmentMeans", in the "Native status" column from the spreadsheet, if a cell is left blank by the author, the "Native" definition will be automatically inserted.

After mapping the information about the "Species Distribution", the generated meta.xml is valid for the entire spreadsheet.
The meta.xml for the example chosen for Configuration One can be seen in the picture below:

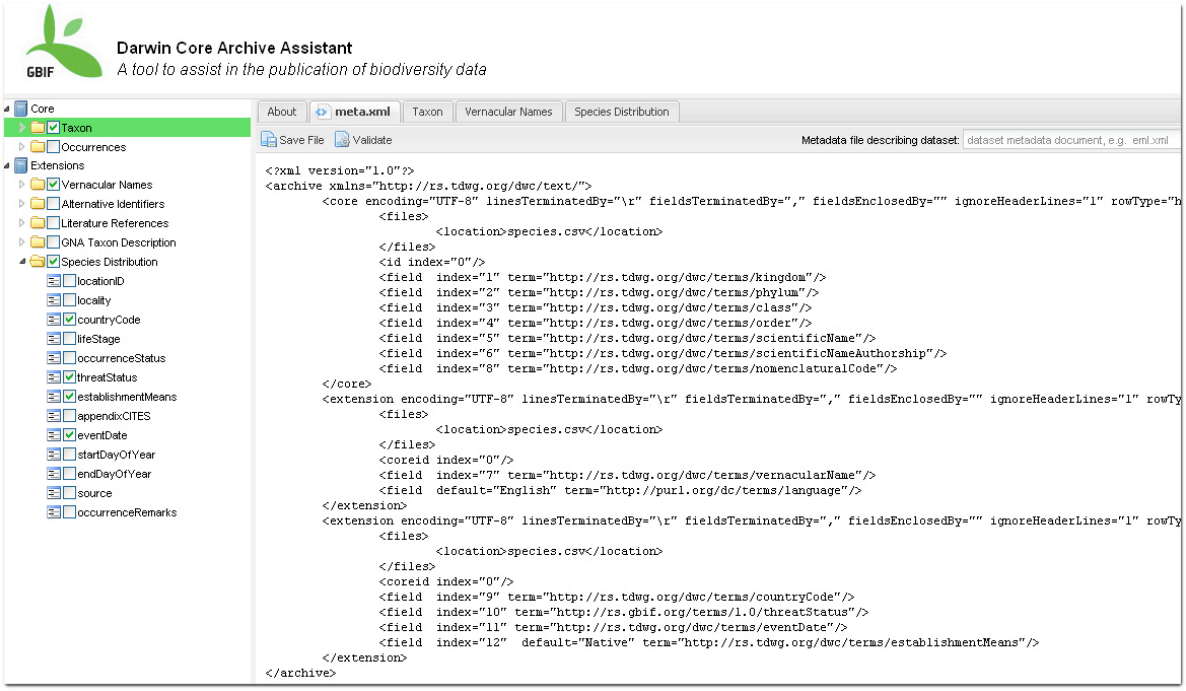


Figure 16 – The final metafile mapping the original spreadsheet to the DarwinCore profile

In summary, this metafile describes the mapping of a single data file (originally a spreadsheet saved as a CSV file) to a core file with a type of "taxon" and two extensions. Note that these three components are described as separate sections in the metafile and this requires the "location" element be repeated three times, and the file description information to also be described three times. Finally, notice that collectively, the three sections, describe fields that occupy positions 0-12, corresponding to the 12 columns in the source data file. Spacers were used to mask columns not relevant in a given extension.

## Configuration Two: Database administrators who manage biodiversity data using relational databases such as MS Access®, FileMaker®, SQL Server®, MySQL®, etc

Configuration Two refers to an example using a simple relational database model or schema that manages data in multiple tables. Such a schema is typical of biodiversity

databases and may be familiar to data managers who use relational database systems such as **MS Access®**, **Filemaker®**, **SQL Server®**, **mySQL®**, etc.

In this example the schema is divided into tables that form natural parallels to the core and extension configuration of DarwinCore Archives. This is no accident as the format was designed to match this common method for modeling data.

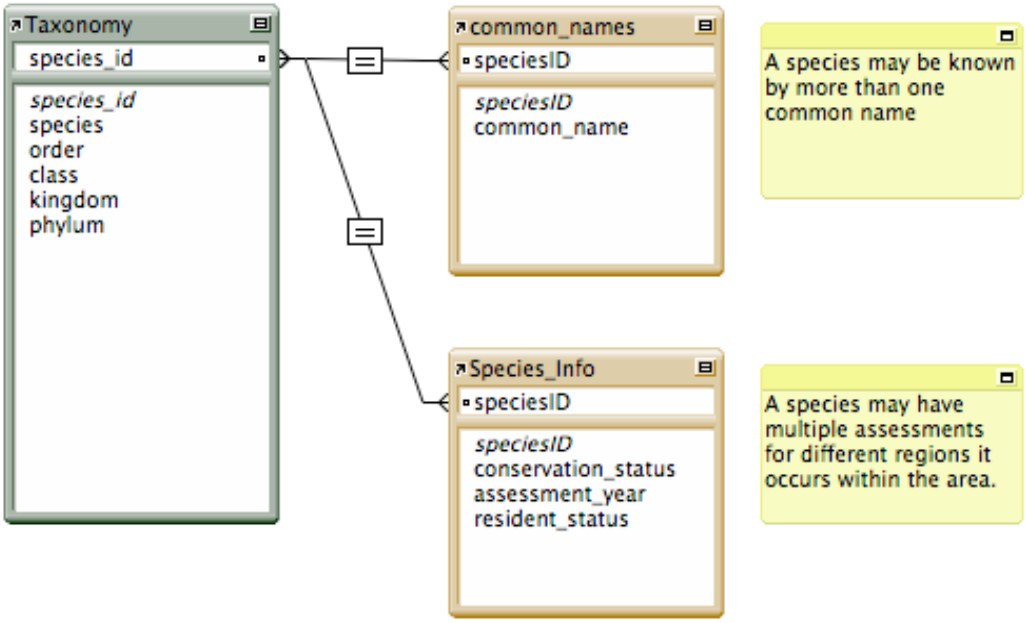


Figure 17 - A simple but typical species-data schema

When working with relational databases, the administrator is able to export files that correspond to distinct core and extension files. Therefore, in this case, the user must create a meta.xml file for multiple data files.

*Relational databases rely on common identifiers between tables to serve as relational links. These primary and foreign keys correspond to the core IDs used in DarwinCore Archive. In this example, species_id is the key identifier for the source database and must be included in each exported data file to serve as the core ID.*

To better understand how to use the DarwinCore Archive Assistant, when working with such a configuration, and also to understand the distinction between Configuraton One and Configuration Two, the same basic data was used as in Configuration One but in this case the data was modeled in the schema shown in Figure 17.

For this configuration, the information was exported from the source database with a file from each table saved as *taxon.csv*, *vernacular.csv* and *distribution.csv* corresponding to the Taxonomy, Common_names and Species_Info tables respectively.

### *Mapping the core data elements*

After selecting the Taxon core data volume from Display Panel, the next step is to register the file name into the header of the newly created page.

1. Set the name of the taxon page file to “taxon.csv”.
2. Record the same file settings for the files, “taxon.csv”
3. Start selecting the terms from Taxon core data volume in the select panel. Often columns names do not match the Darwin Core terms. A short description of a term from the select panel appears in a pop-up window when the user hovers over it. A complete list of terms and definitions can be found on the GBIF schema repository at http://rs.gbif.org/. The first column is the ID, which is also a mandatory field. It is by default found in the first position (zero), but it can be moved around on a different position, just like any other selected term, by a “drag-and-drop” action.
4. Select the terms, Kingdom, Phylum, Class, Order, ScientificName, ScientificNameAuthorship, nomenclaturalCode.

The final output of the Taxon page may be seen in Figure 18 below:

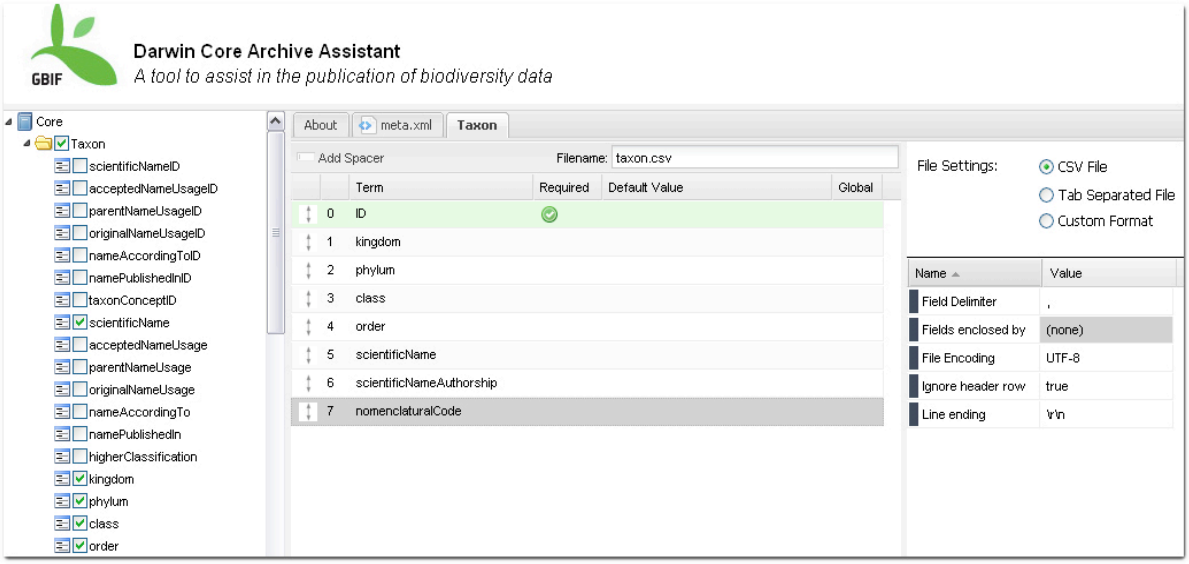


Figure 18 – Field and file settings for the taxon file

The meta.xml for this page can be seen by accessing the meta.xml tab found in the header of the display panel.

## *Mapping the extensions data elements*

Just like in Configuration One, there are two extension data files and for each of them a separate page will be created in the display panel of the DarwinCore Archive Assistant tool, by checking the box next to the respective volumes. In this case, the extension data are stored in separate files.

### Generating the metafile for "Vernacular Names" extension

The vernacular.csv file contains two columns: ID and Common Names. In order to generate the metafile for this document, the user must follow the next steps:

1. Check the "Vernacular Names" box from the left select panel
2. Fill in the name of the published data file – *vernacular.csv*
3. Provide specific file formatting details

4. In this case, both terms, ID and Common Names are already selected by default, as “required fields”.
5. The common names in this database are all in English. This information may be added as a global value, indicating that all vernacular names in the list may be assigned a language property with the value of “English”
    a. Select the “language” element from the vernacular extensions list in the Select Panel
    b. Click on the “Default Value” entry field in the corresponding row in the Display Panel.
    c. Enter “English”
    d. Click the Global checkbox to the right of the entry field

If the language column is also added, the final output for the “Vernacular Names” page will be as follows:

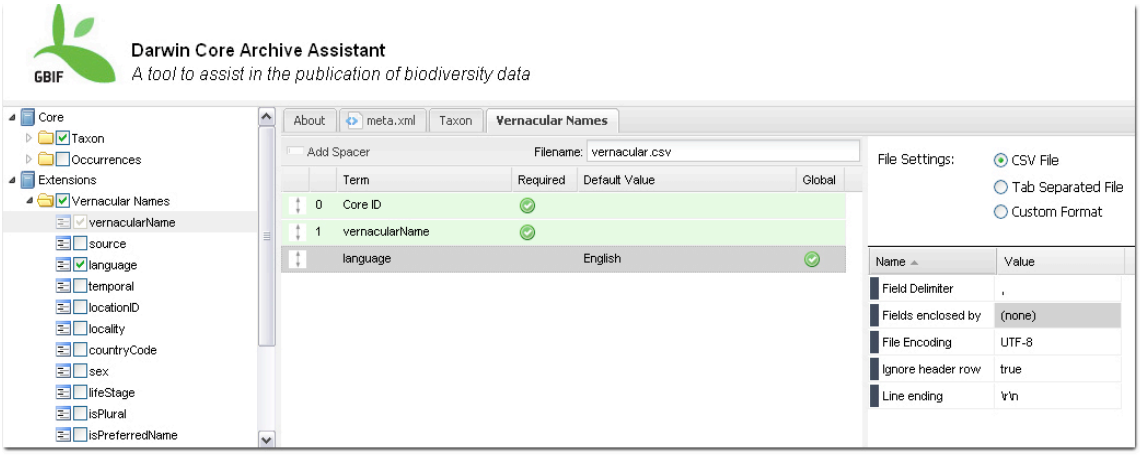


Figure 19 – Mappings and global values for the vernacular names data

The meta.xml will now also contain information about the extension data files. Click on the meta.xml tab to view the xml file.

**Generating the metafile for “Species Distribution” extension**

The distribution.csv file contains five columns: ID, Countries, Threat Status, Year Evaluated, Native Status.

In order to generate the metafile for this document, the user must follow the following steps:

1. Check the "Species Distribution" box from the left select panel
2. Fill in the name of the published data file – *distribution.csv*
3. Provide specific file formatting details
4. Identify and select the terms from the left select panel

After selecting the terms from the *Species Distribution* extension files volume, the final display on the "Species Distribution" page will look as pictured in Figure 20 below. For the "Status" ("establishmentMeans") the "Native" value was set as default value, but the "Global" attribute was not checked meaning that if there is another value in the column, it will not be replaced by "Native".

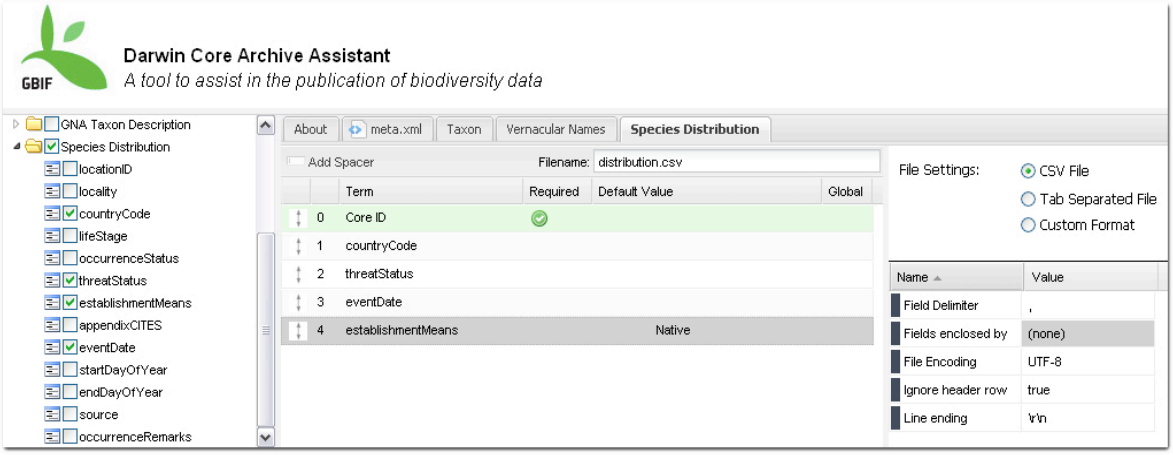


Figure 203 – The final display on the "Species Distribution" page

The final metafile for Configuration Two can be seen by accessing the meta.xml tab from the header of the display panel, as in Figure 21 – the final Configuration Two metafile below:

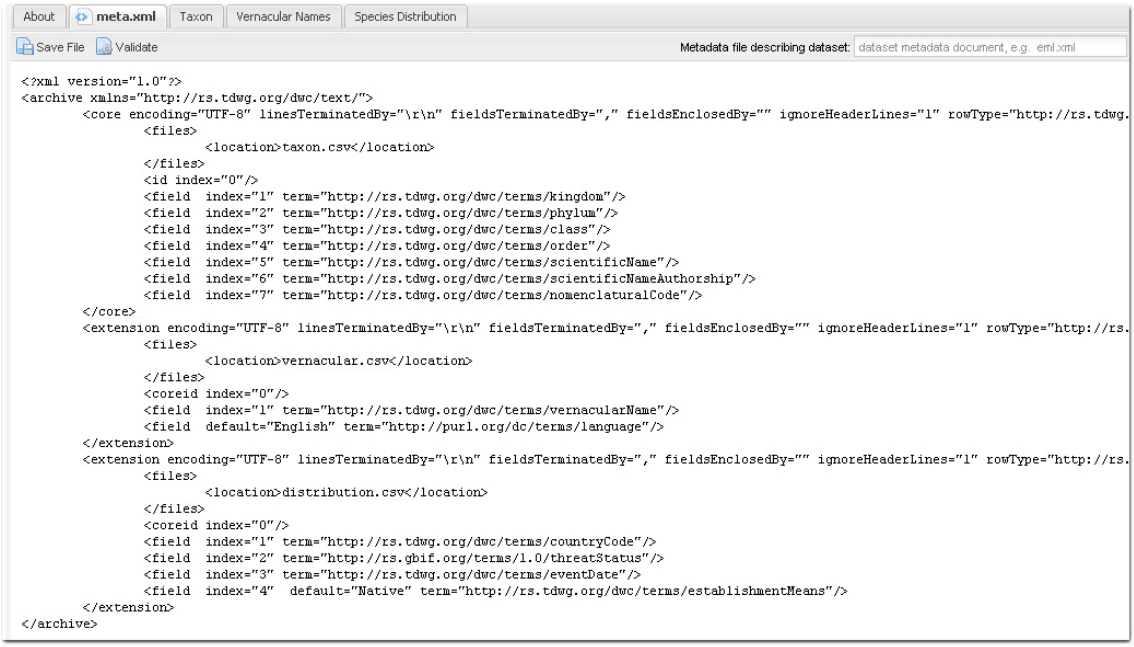


Figure 21 – the final Configuration Two metafile

## Using the meta.xml file to create a Darwin Core Archive

Once the validation of the *metafile* is complete, the user may save a copy of the output metafile via a simple copy and paste operation or by clicking on *Save File* button found in the header of the layout.


Figure 22– The "Save File" button on the meta.xml page

A pop-up window will appear asking to save the file. The file must be stored in the same folder as the other documents that will form the Darwin Core Archive.

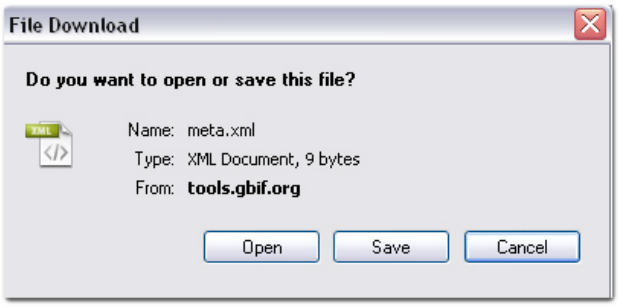


Figure 23 – The pop-up message for saving the file

**Error! Reference source not found.** displays the contents of an example folder containing a typical archive composed of a:

| | |
|---|---|
| Metafile | *meta.xml* |
| Metadata file | *eml.xml* |
| Core data file | *taxon.txt* |
| Extension file | *distribution.txt* |
| Extension file | *vernacularname.txt* |

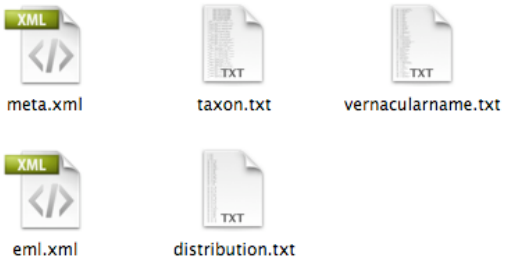


Figure 24 – Example of a typical archive folder

Once all the documents are in the same folder, if can be compressed into an archive using a zip file generator or a TGZ generator.

The newly created archive file may be validated using the **Darwin Core Archive Validator**[5]**.** This process examines all the files in the archive, not just the metafile and ensures the files conform to the standard and will be properly interpreted by tools and services that support this format.

[5] http://tools.gbif.org/dwca-validator/