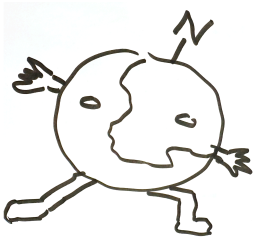
DCN Planetears
Geospatial Data Curation:
an introduction



Module 5: Transformations Instructor Notes

[This handout accompanies a slidedeck with the file name “5_Transformation”]
2024-02-16 1100

Lesson Plan: Transformations (approximately 52 to 76 minutes, depending on the time allotted for activities and group discussion)

Objectives:

- Understand common reasons for recommending transforming file formats
- Assess benefits and downsides of common file formats
- Able to perform transformation of vector data in QGIS
- Bonus: able to reproject a vector file in QGIS

Slide notes: see slides and below

Activities:

- Discussion and brainstorming what transformations to recommend for layers in an example geodatabase (.GDB)
- Demo/Hands-on Practice: Transforming .GDB layers into another file format (e.g. turning a feature class into a CSV)
- Bonus: Reprojecting data into a new coordinate reference system (CRS)

Notes for Slides:

1. **Geospatial Data Curation: an introduction: Module: Transformation (1 minute)**
   Welcome to the next module in the “Geospatial Data Curation: an introduction” curriculum, Transformation. So you’ve checked the files and have a basic understanding of what they are, how they fit within the existing research project. When we are taking these files into a repository, we are helping researchers shift from active development of a dataset to preservation and long-term use. Once we’ve done that initial assessment, it’s time to consider: Are these files in the best format and structure possible? Are they

in a format that will allow them to be accessible to a broad range of people and well into the future?

2. **Module Objective: Transformations (1 minute)**
   This module has three objectives. At the end of the module, learners should be able to:
   1. Understand common reasons for recommending transforming file formats
   2. Assess benefits and downsides of common file formats
   3. Perform transformation of vector data in QGIS

3. **Reason That It Is Hard (1 minute)**
   Decisions around preservation formats and transformation are tricky for geospatial data because there isn't a clear archival standard (especially with vector and database formats). Why is this?

4. **Vector Geospatial File Formats Over Time (2 minutes)**
   One reason is that geospatial file formats have changed a lot over the past 40 years and they continue to change rapidly. This slide shows some major vector geospatial file formats over the past four decades. There has been substantial innovation improving the functionality and storage capacity of formats. But that doesn't necessarily track with which formats are most frequently used.
   a. We haven't talked about coverages (which was a dominant file format in the 1980s) because they are basically never used anymore.
   b. But shapefiles (a format emerging in the 1990s) is one of the most ubiquitous geospatial file formats out there. Despite honestly a lot of limitations (which we will talk about in a little bit) and an active smear campaign by the company that invented them, they can be opened by almost geospatial software.
   c. And then you have Geopackages, which were invented in the mid-2010s, that are open format with solid functionality. But despite being around for a decade, they are not widely used yet.

   So if you are trying to use a file format that is going to be readily opened by a large number of software and stand the test of time, it's hard to know which one to pick.

5. **Proprietary vs Open File Formats (2 minutes)**
   Also complicating things, is a tension between proprietary and open file formats.
   In general, when curating data, we opt for open file formats whenever possible. It's why we ask for CSV instead of Excel Spreadsheet (XLS or XLSX) and recommend PDF instead of Word documents. In geospatial though, some of the most commonly used formats are proprietary. In the last slide, we mentioned that shapefiles are widespread - they are also technically proprietary.

   Many proprietary geospatial file formats can be opened with open source software. Open source geospatial community generally adapts very quickly to new formats. If you have a format that cannot currently open using QGIS, wait a few years and you likely will be able to. [insert personal anecdotes if desired] You often do lose functionality when opening proprietary formats in open source software. But if you transform the files to an alternate file format you may very well end up changing content or functionality as well. It's hard to know how much effort to put into making file formats accessible, because it's a constantly moving target.
   [Next slide]

6. **Questions to Consider (1 minute)**
   In the DCN CURATE(D) workflow, “T” is for “Transform file formats”
   https://docs.google.com/document/d/1RWt2obXOOeJRRFmVo9VAkl4h41cL33Zm5YYny3hbPZ8/edit#heading=h.ax5e5z3ym1pa

   So when you are trying to make decisions about transformation, ask:
   1. How complicated would it be to represent a file in an open format?
   2. How much effort would it take to make that transformation?
   3. And how much more accessible are you actually making that data through transformation?

   [Next slide]

7. **Library of Congress Recommended Geospatial Formats (2 minutes)**
   Turning to recommendations from the Library of Congress, they have some preferred formats including many of the options we've looked at today:
   - Shapefile
   - GeoTIFF
   - Esri File Geodatabase
   - OGC GeoPackage
   - GeoJSON

   The overriding principle they recommend is the "Most complete data...even if proprietary, with a preference for preserving the native format and projection of the data."
   (Note that in this direct quote “projection” is synonymous with “coordinate reference system”) Recommended Format Statement (Library of Congress)

   [Next slide]

8. **Why might you want to transform file formats? (2 minutes)**
   So why might you want to transform files?
   - One of the main reasons would be to simplify the file format. So for example, let’s say you have a single, simple vector layer inside a geodatabase. Geodatabases are built to be able to hold a substantial amount of data, complex relationships, and toolboxes. It’s kind of…overkill for one layer. It’s like having a Microsoft Access Database when really what you need is one spreadsheet.
   - So to increase the number of software that can open the file, you might want to switch to something else. You can open vector data held in a geodatabase with open source software QGIS, but having the data in CSV, GeoJSON, even a shapefile would increase the number of software that could open it.
   - You also might want to transform it to provide access to data in proprietary formats that cannot be opened with open source tools. For example, you cannot open raster data that is in a geodatabase in QGIS or other software. You might provide that data in another format, while waiting for the open source community to catch up and make it accessible.

   [Next slide]

9. **What should you watch out for when transforming files? (1 minute)**
   As you are making a decision about whether to transform file formats, what things do you need to watch out for? At its core, the warning is that some geospatial data formats are complicated and switching between them can introduce unexpected errors and loss. We are going to focus on a few specific issues - the limitations and quirks of shapefiles and embedded metadata.

10. **Watch Out For: Shapefiles (2 minutes)**
    Shapefile is a popular format, many software can open them and they have this perception of being a good format for sharing data publicly. But shapefiles have a lot of limitations and quirks.The thing that is the most visible and frustrating is that variable names are restricted to 10 characters. If they are longer than that they get clipped off during transformation, which leads to very strange names and discrepancies between variable names and documentation. There are a number of less known limitations that can also cause data loss. For example, date fields do not support time values, only day, month and year. Text fields are limited to 254 characters. NULL values don't exist. There are also limitations on the number of fields and overall size you can have. And if you are not aware or anticipating issues, the effects are easy to miss.
    a. There is a custom python script that can warn you about transforming a geodatabase into shapefile. It only works in ArcGIS Pro, so we are not going to look at it now. But just in general we wouldn't recommend transforming things into this format. (Geodatabase Data Curation ArcGIS Toolbox)

11. **Watch Out For: Researcher Transformed Shapefiles (1 minute)**
    One thing to watch out for is that sometimes researchers will send you data that they have already transformed. Preemptively they may have decided to distribute data as a shapefile, because it is such a widely used format. If you see that the field names in the documentation don't match the field names in the data - especially if they have been cut-off or shortened -- it is a sign that the original data format was something else. If the variables seem complex, you might want to follow-up to make sure the researcher knows about the limitations of shapefiles.
    [Next slide]

12. **Watch Out For: Metadata (1 minute)**
    It's very very easy to lose embedded metadata during transformation! This is especially true of:
    - transformations done using QGIS. QGIS is excellent and very effective at transforming the data itself into various formats, but it mostly ignores attached metadata (especially if it was made outside of QGIS).
    - Likewise, metadata is generally lost when transforming from proprietary into open formats

    Check whether metadata is still embedded after transformation and provide alternative access if needed.
    [Next slide]

13. **Watch Out For: Additional Components in Project Files (1 minute)**
    When you are dealing with more complex file format structures like geodatabases or project files, it's important to watch out for "extra" elements that may not be immediately visible. These could be things like custom tools, tasks or workflow models, or layers of metadata. These elements can be hard to represent outside of proprietary software.

Custom toolboxes often require proprietary software to run successfully. Geodatabases and projects may contain multiple-levels of embedded metadata that can be time-consuming and difficult to export into a format that will be interoperable with other systems. If these elements are crucial to the submission and reuse of the data, it might not make sense to transform to another format (even if the original format is proprietary).
[Next slide]

14. Possible Transformations to Consider (1 minute)

Okay - so let's look at a few examples or what this might look like in practice. We recommend that you have the researcher do the actual transformation. There is always a chance that you might fundamentally alter something, so it's better to have the person closest to the data checking its integrity.

First, we are going to look at turning a Shapefile with Point data into a CSV. This would be an example of simplifying the data format
[Next slide]

15. Possible Transformations to Consider: Shapefile Point Data to CSV: 01 (1 minute)

Recall that vector data is basically a spreadsheet with geometry…a spreadsheet where each row represents a physical location such as a point, line, or area. To represent this, shapefiles require a set of related files - a spreadsheet (.dbf) plus several files describing the geometry and how that geometry relates back to the spreadsheet.
[Next slide]

16. Possible Transformations to Consider: Shapefile Point Data to CSV: 02 (1 minute)

But if you are working with point data, you don't really need three separate files to describe locations. That information can be conveyed just as well with a couple extra columns in the spreadsheet for latitude and longitude. One option for simplifying things and reducing the overall number of files in a submission is to export the spreadsheet portion of the shapefile as a CSV with the locations included.

a. Note: Doing this will remove embedded information about the projection and Coordinate Reference System. This information can be captured in the documentation but may not always be the best move.

[Next slide]

17. Possible Transformations to Consider: Geodatabase: 01 (1 minute)

What if you are dealing with a more complex geodatabase, one that includes closely related datasets that have internal organization? There may be a lot of long variable names or special field types. We could convert some of them into CSV - the point data. But we couldn't do that for the polygons, lines, or raster.
[Next slide]

18. Possible Transformations to Consider: Geodatabase: 02 (1 minute)

We could convert them into a shapefile, but there would be a potential for a lot of content and structure to be lost.
[Next slide]

19. Possible Transformations to Consider: Geodatabase: 03 (2 minutes)

What else can we do?

- We could just leave it. While there are only a few platforms that can open a geodatabase, most of the layers can be opened in QGIS.
- We could focus on the layers that are not well-recognized in open source software. QGIS can open vector data in a geodatabase, but not raster data. So we might want to export just the raster layers into another format.
- You also might talk with the researcher and decide to focus on just a few priority layers. So maybe most of the layers have been included to provide context, but they aren't things created by the researcher. Maybe this researcher did a survey of water fountains and that data isn't available elsewhere. We might suggest providing just that data in a more open, accessible format to highlight it/make it more visible.

[Next slide]

20. Possible Transformations to Consider: Geodatabase: 04 (1 minute)

If the researcher involved is set on providing the data in an open format. In this situation, you might suggest transforming to a GeoPackage. We haven't talked about it much today, but it's an open source equivalent of a geodatabase and it could be an option. It has similar functionality, but the process of transforming content always risks change and embedded metadata will likely be lost.

[Next slide]

21. Activity: Discussion in groups (15 to 20 minutes)

Imagine if the data we have been using for this training had been sent to us as a geodatabase. Take a look at this inventory in small groups and talk about what you would recommend for a few minutes, then we will come together as a large group and look at how we would implement it.

- ❖ Have participants review the geodatabase using QGIS and looking at the inventory .TXT file.
- ❖ Possible recommendations:
  - ➢ Have a copy of the raster files outside of the geodatabase since they cannot be accessed without ArcGIS
  - ➢ Have a copy of the two point layers (AmericanWoodcock_nests and GoldenWingedWarbler_nests) outside of the geodatabase because they are unique locations collected by the researcher
  - ➢ Move everything into a .GPKG to use a more open format

[Next slide]

22. Activity: Practice Transformation (10 to 20 minutes)

- ❖ Take action based on the recommendations the group decided on during the discussion.
- ❖ This could include transforming:
  - ➢ Point feature classes into CSV (with CRS information EPSG:26915 noted)
  - ➢ Feature classes into an open vector data format
  - ➢ Moving the raster out of the geodatabase can't be done within QGIS, so would need to be a request to the researcher - You could practice writing this request as a group

- ❖ If you have time, you might also demonstrate how to reproject data starting from the files in the 5_Bad_projection

[Next slide]

23. Final Thoughts on Transformation (1 minute or more if you allow for report backs after the activity)

- Weigh the amount of effort it would take to transform files against improvement to accessibility
- For complicated datasets, it is unlikely that transformation from proprietary into open formats will be possible without losing some essential pieces
- Prioritize elements that will be inaccessible outside of proprietary software
- Educate researcher about open formats for use in future projects
- If files are transformed, include steps taken in the documentation