Specs

# User-facing

## General

1. Command line arguments for major data sources and output
2. –as-copy command line argument functions like test, but is meant to be used if it’s ever run with different zones (not HUCs), or if other parameters change
3. –use-existing-data – meant to be used with –as-copy or another parameter that changes how data will be handled – it creates a new “New\_Data.mdb”, then fills it with the existing data, before beginning processing.

## Importing Data

Generally, it should continue past errors whenever possible, alerting the user to the need to manage that data. For example, if it’s unable to process a dataset, the program should skip it and continue instead of crashing

Changes to the database should not be committed until the last step. That way if an exception that causes a skip is raised, we’re ok and the database isn’t polluted.

## Handling Data

Do a “laugh test” on the data – check if it’s the only HUC in its HUC10 with fish, where the water flows, whether nearby watersheds have the fish (so it could be just an intersection error) and take into account the percentage of the HUC that overlaps when it’s a polygon.

# Technical-side

## General

1. As many portions of it as possible should be callable as functions. though this should be limited by practicality

## Data Imports

1. **Import in Parallel:** Ideally, it forks the program while importing data in order to execute lots of these, mostly CPU-bound operations in parallel. Have a max\_cores parameter that defaults to the 75% of the total cores (rounded down, so that it always leaves one core free, unless it’s a single-core computer. If rounding down hits zero, then change it to 1).
2. **Data should be traceable:**
3. **Certain temporary data should be cached permanently:**
   1. **Other data should not:**