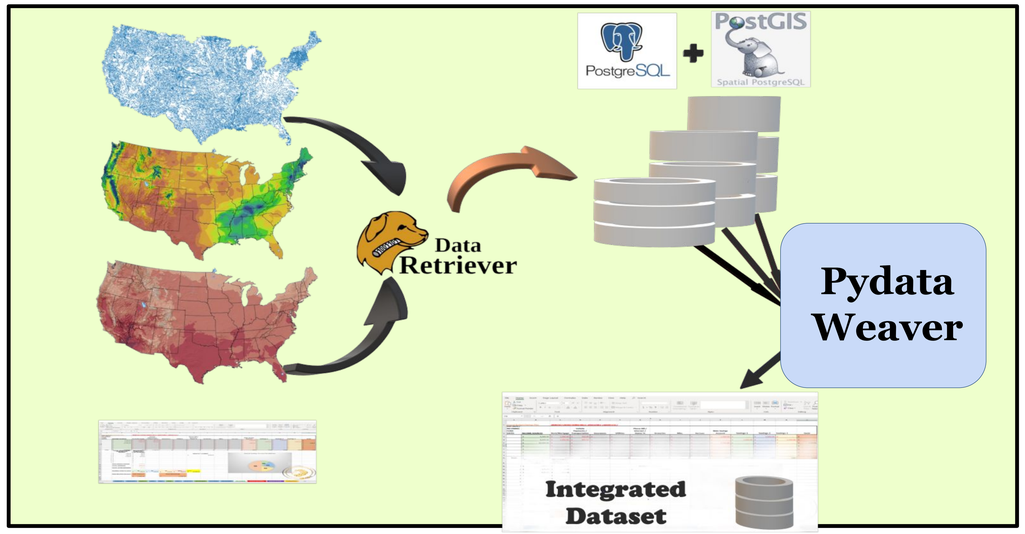
The PyDataWeaver: A data integration platform

Henry Senyondo, Andrew Zhang, and Ethan P. White
University of Florida, Department of Wildlife Ecology & Conservation and The Informatics Institute

A Fast and Clean Data Integration Pipeline

Combine publicly available data into ready-to-analyze datasets. Most research questions require integrating multiple datasets. Researchers typically have to manually create these combined datasets using databases, programming languages and GIS systems. The PyDataWeaver automates the task of integrating multiple data sources to create new comprehensive datasets using JSON based a data integration standard.



Using the Pydata Weaver

Python Package & Command-Line Interface

Installation:

```
$ pip install pydataweaver
```

List available datasets :

```
$ pydataweaver ls
```

Integrate Mammal Community DB (tabular) with Bioclim climate data (raster):

```
$ pydataweaver join postgres mammal-community-bioclim
```

Get the citation for a dataset:

```
$ pydataweaver citation mammal-community-masses
```

Design Considerations

- Out of memory storage & computation:Database management systems backend storage and integration
- Automated data cleaning and install: Uses the Data Retriever data package manager
- Spatial join support: Uses spatial database extensions for the integration and aggregation
- Supports both command line and Python interface

Integration and Package Definition

The PyDataWeaver supports both tabular and spatial datasets and uses the DataRetriever package to clean and install the required datasets into a database. The platform uses a JSON metadata definition protocol, that defines the data and properties required for integration.

Pre-integration: automate cleanup + install

- Use DataRetriever to download & clean data
- Supports tabular and spatial datasets
- Standardizes spatial references
- Installs cleaned datasets into SQL databases
- Resolves SQL keywords in data
- Correct non-standard null values

Integration of installed datasets

- Uses JSON metadata for the integration
- Identifies and resolves overlapping attribute names
- Perform aggregation in the database for out-of-memory support
- Removes unwanted columns during table joins
- Use left joins to ensure complete row information

JSON-based Data Integration Definition

Dataset Definition

JSON data integration package meta file includes:

- Data package name
- Output table name to store the new dataset
- List of tables to be integrated
- Assemble tables in the order of common attribute values
- List of join attributes among tables
- Include supporting data like version and citation

```
{
  "name": "portal-plot-species",
  "version": "1.0.0",
  "citation": ["Ernest, S.K.M. 2019..."],
  "keywords": ["mammals"],
  "licenses": {"CC0"},
  "result": { "dataset": "res_all" ..},
  "main_file": {"path": "portal.main",]},
  "tables": [
       {"database_name": "portal",...],
  "join": [{"table": "portal.plots",
      "table_type": "tabular",
      "fields_to_use": [],
      "join_ocn": {"common_field": [],
        "portal.plots": ["plot_id"],
      }
    },
…}
```

Development

Software Details

- Python 3.5+
- PostgresSQL + PostGIS (using gdal)
- Open Source (MIT)
- GitHub: https://github.com/weecology/pydataweaver
- Retriever: https://github.com/weecology/retriever

Future Development

- Spatial support for SQLite, MySQL and HDF
- Automated data package creation
- Support for hierarchical datasets, HDF5

Funding: The Gordon and Betty Moore Foundation's Data-Driven Discovery Initiative through Grant GBMF4563 and by U.S. National Science Foundation through Grant 0953694 both to E. P. White.