Software to Streamline Sharing of Agricultural Algorithms and Data

David LeBauer, Christophe Schnaufer, Kristina Riemer

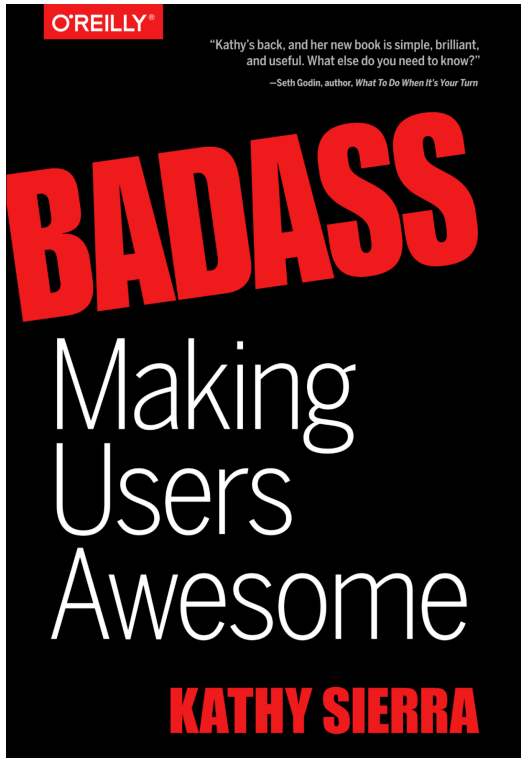
THE UNIVERSITY OF ARIZONA
College of Agriculture & Life Sciences

Introduction

Challenges

How to make appropriate data deposition commonplace?

Solution



UA Digital Agriculture Group

Mission: Providing solutions for computing in agriculture, so that we can engineer crops and sustainable agricultural landscapes.

Vision: Faster, more collaborative agricultural science and engineering through shared software and data.

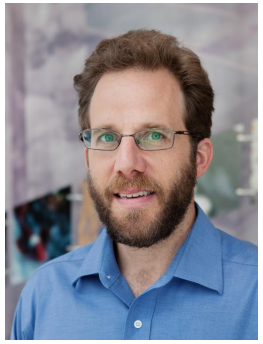

Emily Cain


Kristina Riemer


Julian Pistorius


Jorge Barrios


Chris Schnaufer


David LeBauer

Collaborators, Too Many to List:

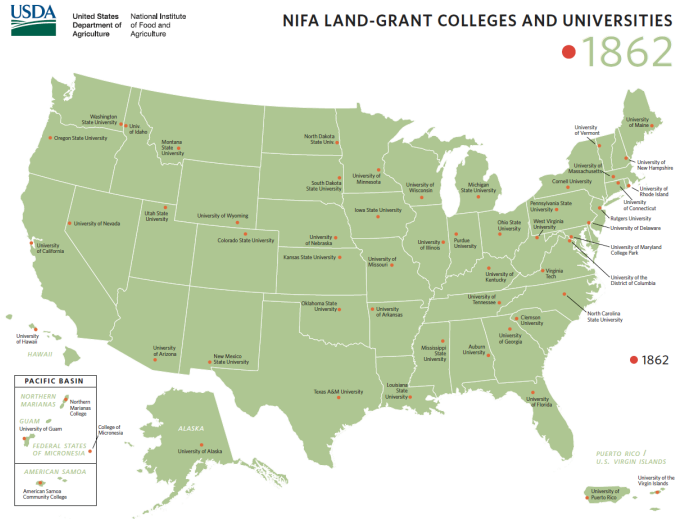
Tyson Swetnam, Brian Heidorn, CyVerse, Todd Mockler, Nadia Shakoor, Geoff Morris, Vasit Sagan, Robert Pless, Rob Kooper, Max Burnette, Steve Long

Others to be listed as we go …

Outline:
Our Approach in 4 Ts

- Tools
- Translators
- Tutorials
- Templates

Open Software as Germplasm:
Aligned with CG and Land Grant Missions



CIMMYT genebank

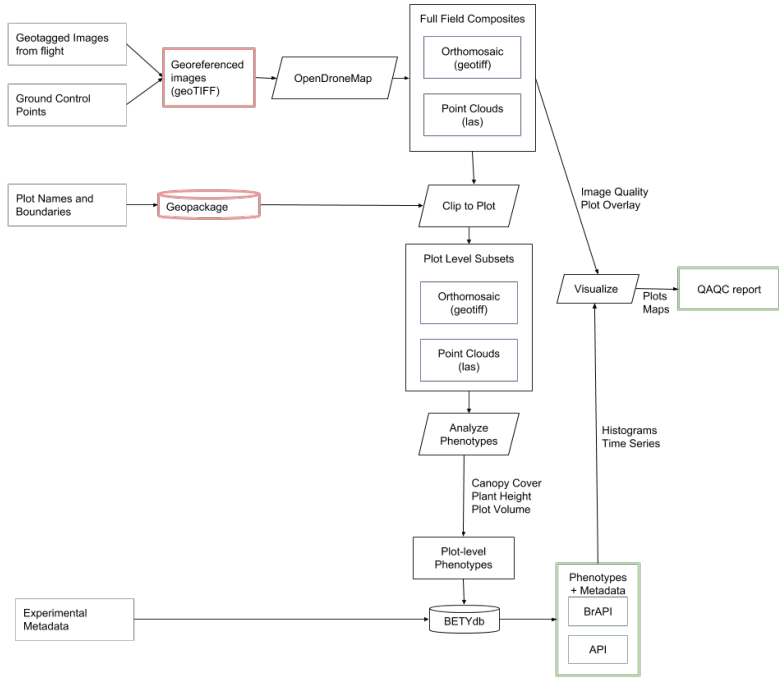
Outline:
Our Approach in 4 Ts

- **Tools**
- Translators
- Tutorials
- Templates

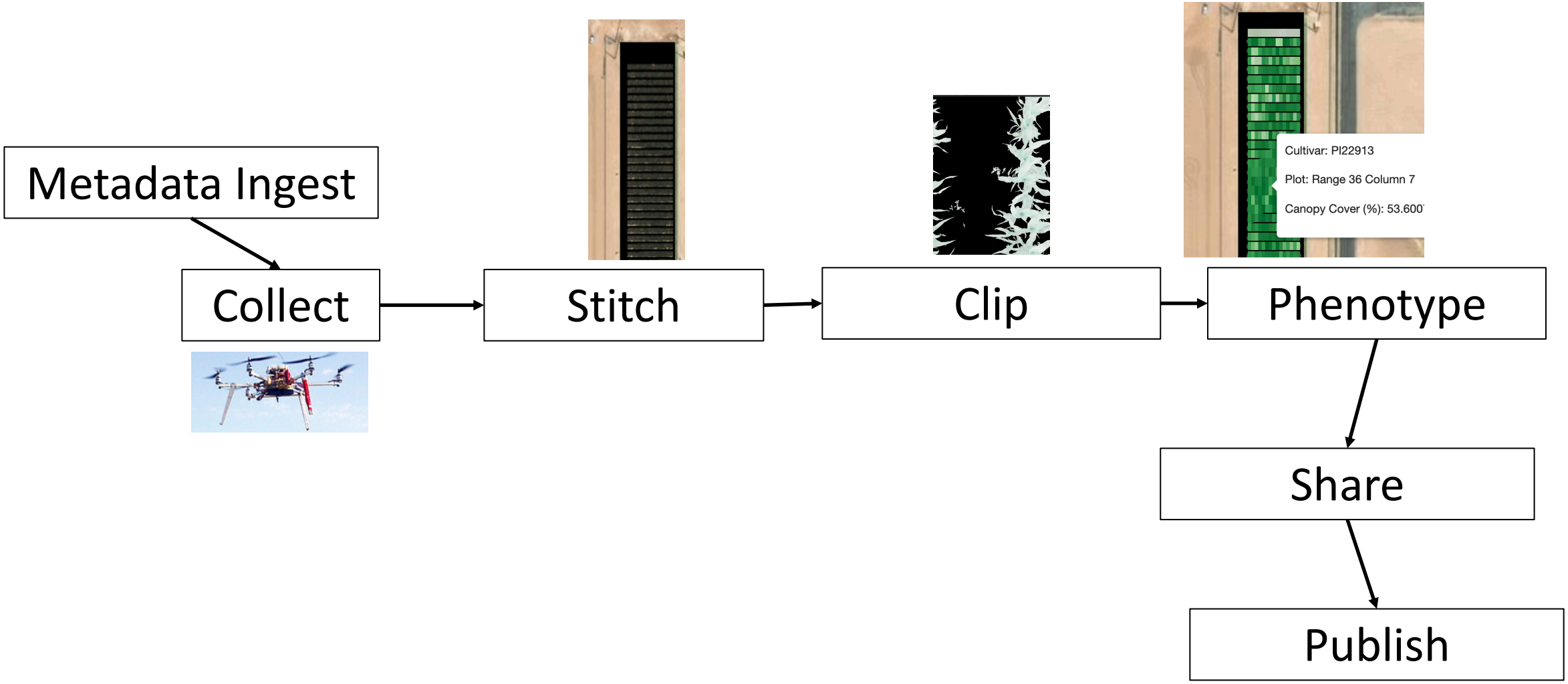
TERRAPHENOTYPING REFERENCE PLATFORM

DPP

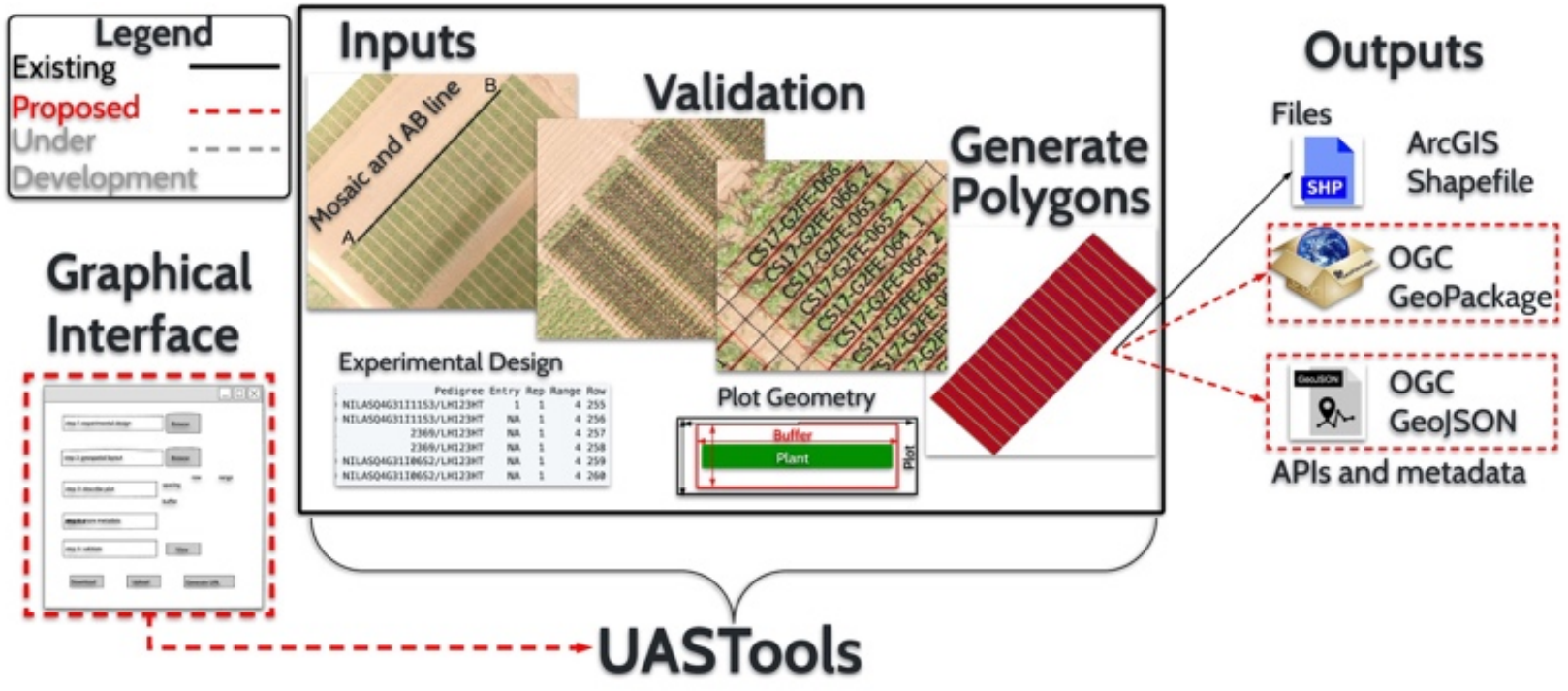
High Throughput Phenomics (images → phenotypes) Pipelines



A Simple Drone Pipeline



Metadata Ingest: Plot Boundaries

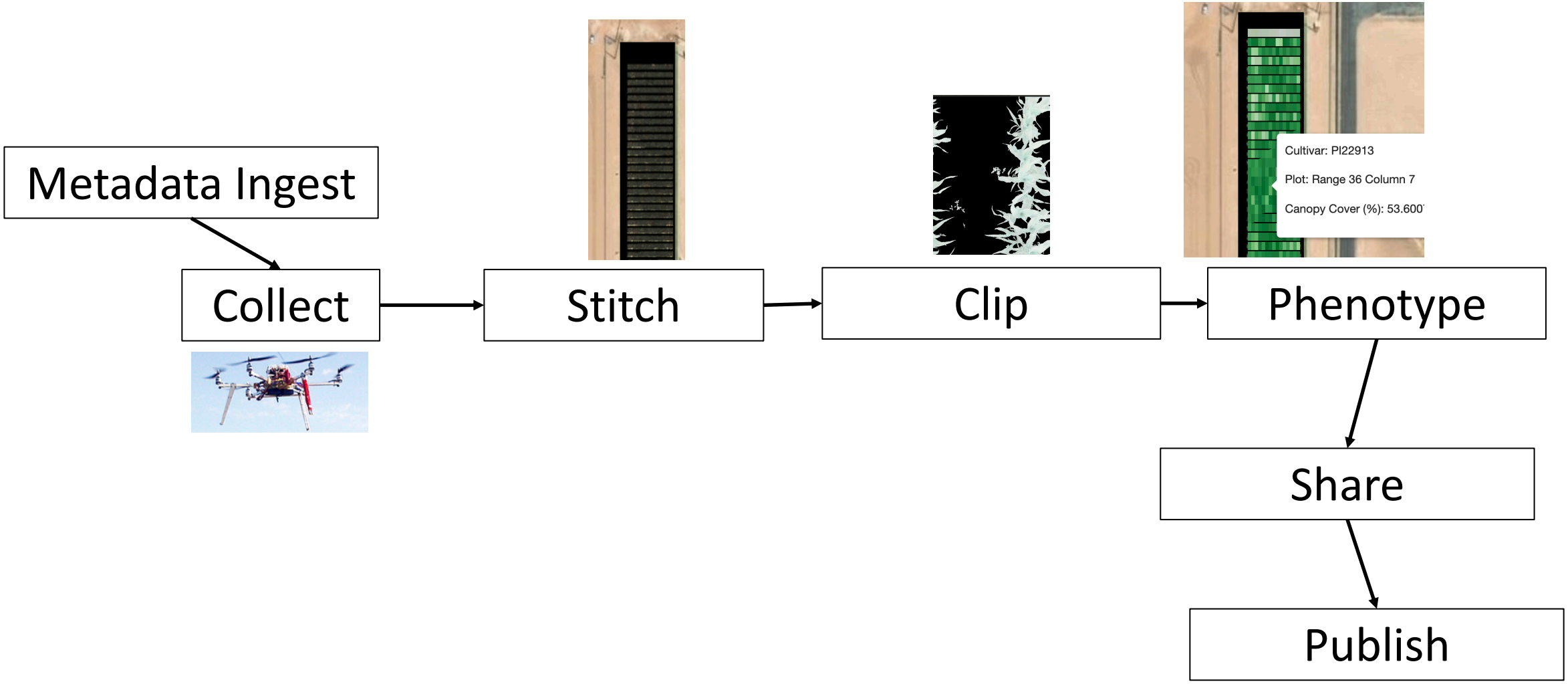


Works ✓

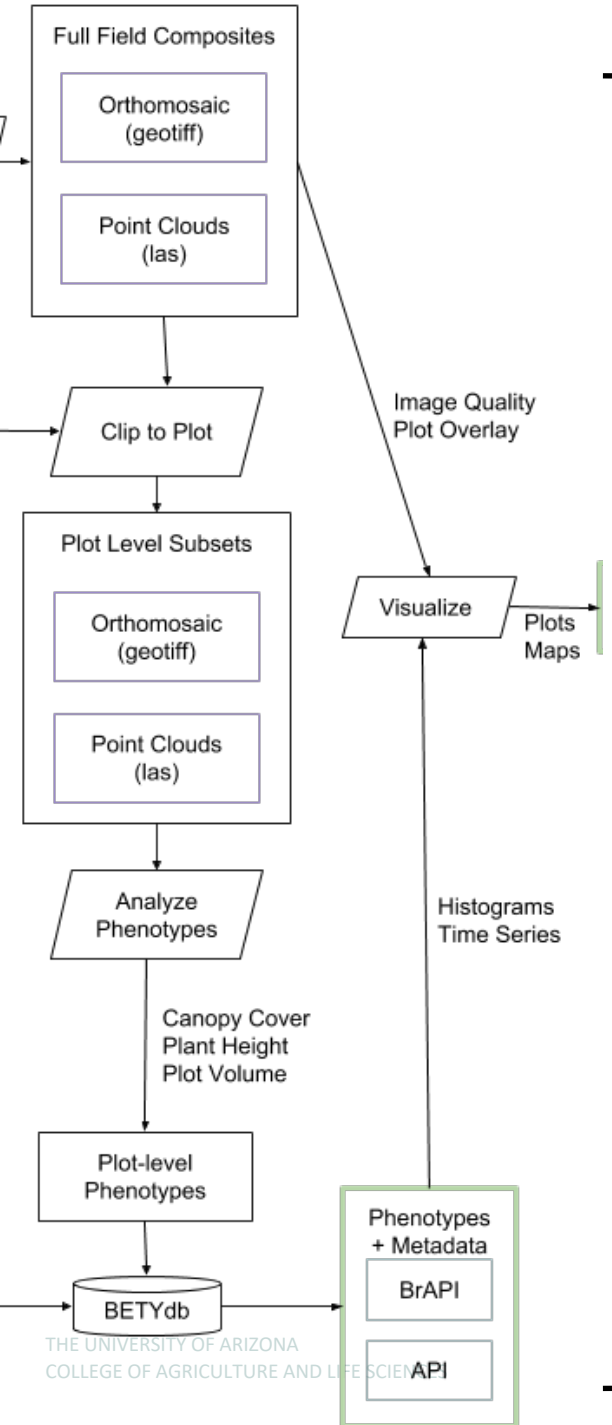
Open Source ✓*

Standard API ✓*

A Simple Drone Pipeline



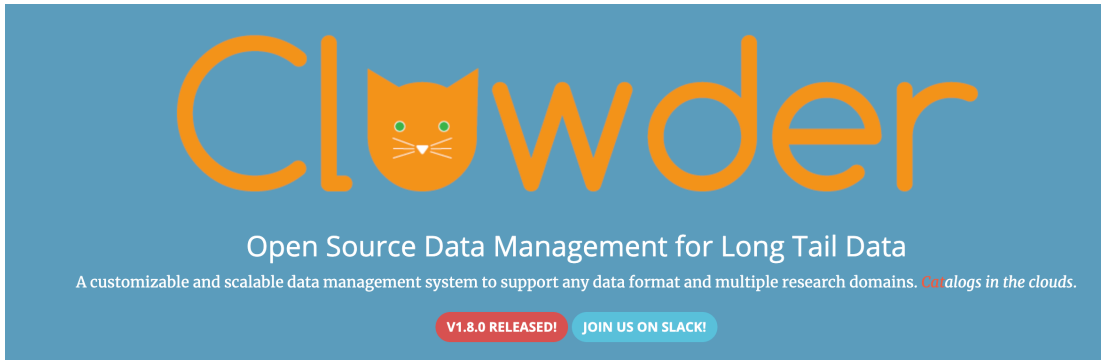
Data Publication

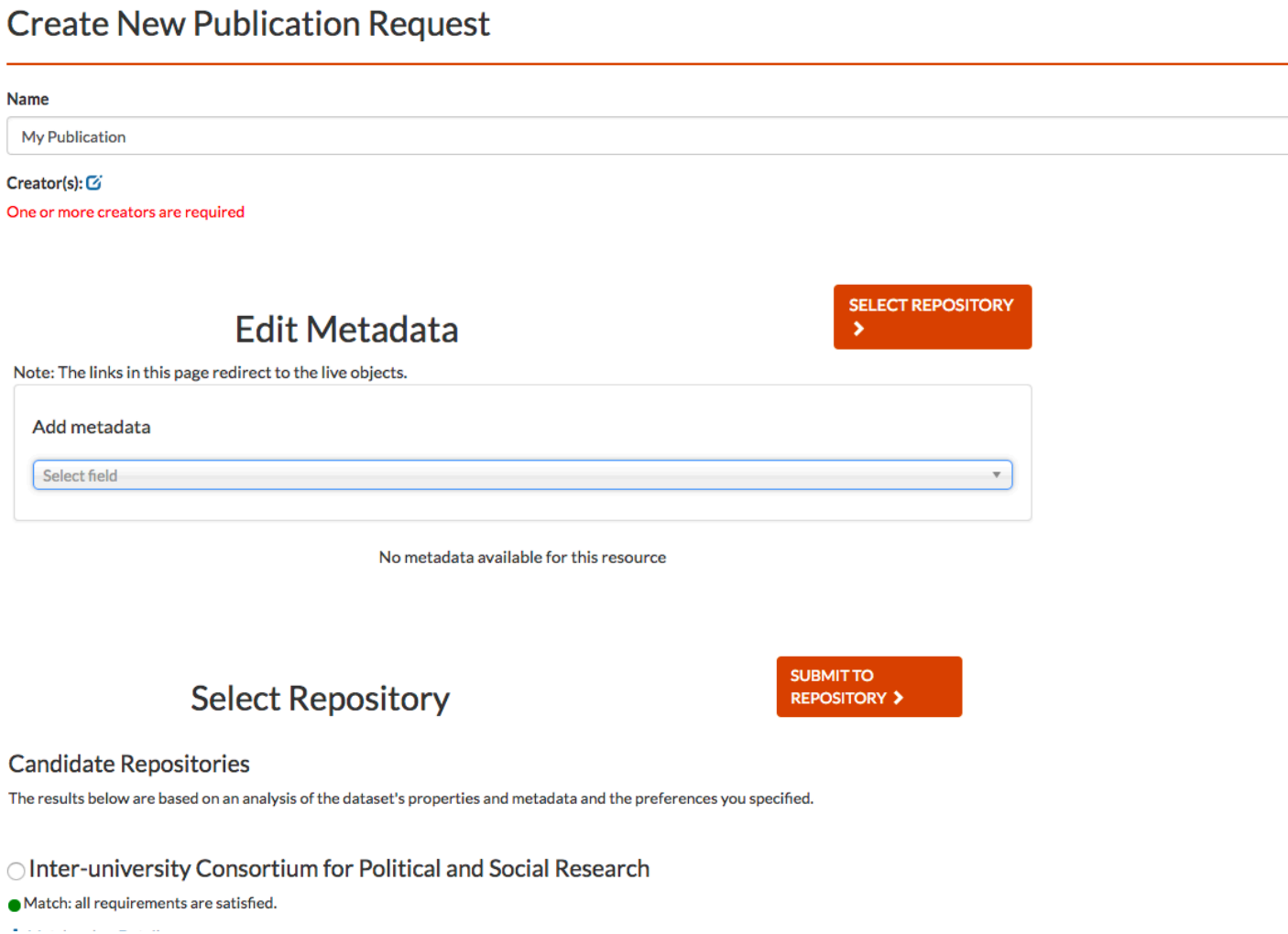


Publishable Object

- My library
- Your Library
- CyVerse
- Dryad

Data Publication





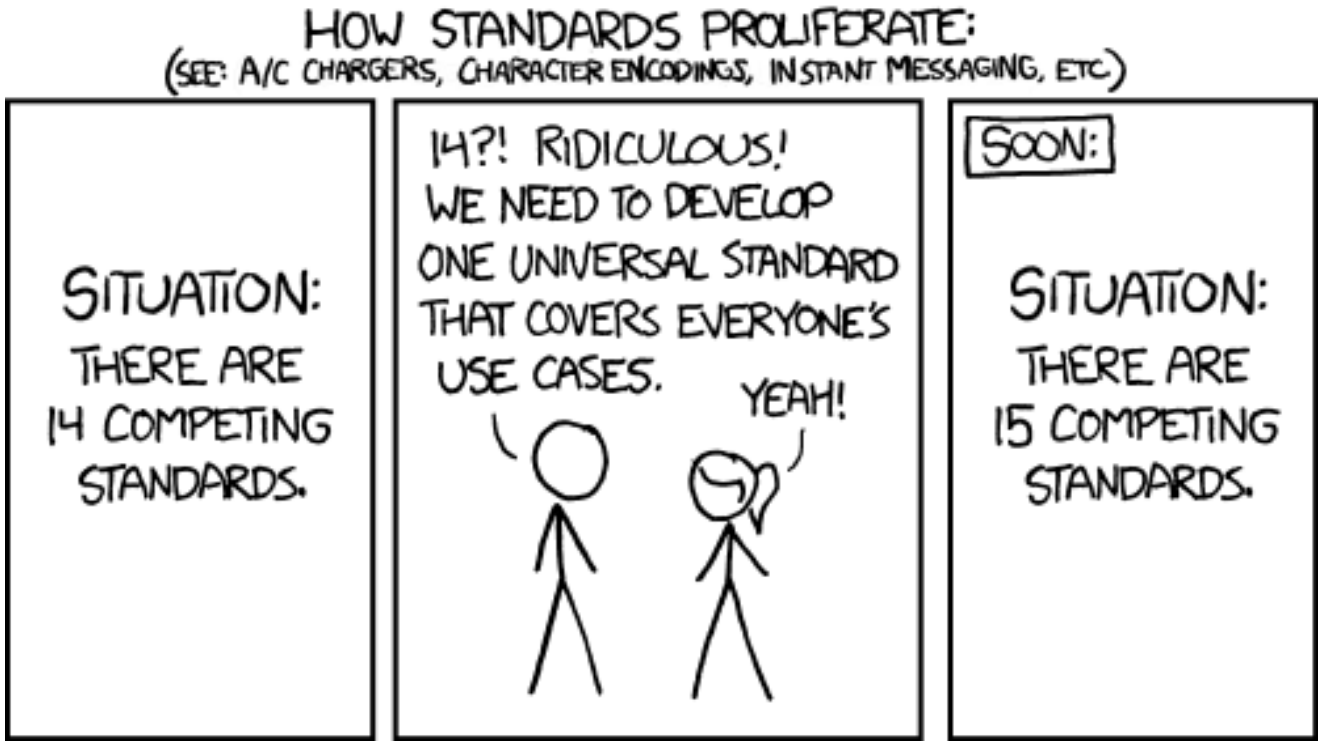
terraref.org/clowder/

clowderframework.org
clowder.ncsa.illinois.edu/docs/publishing_data.html

Outline:
Our Approach in 4 Ts

- Tools
- **Translators**
- Tutorials
- Templates

Translators

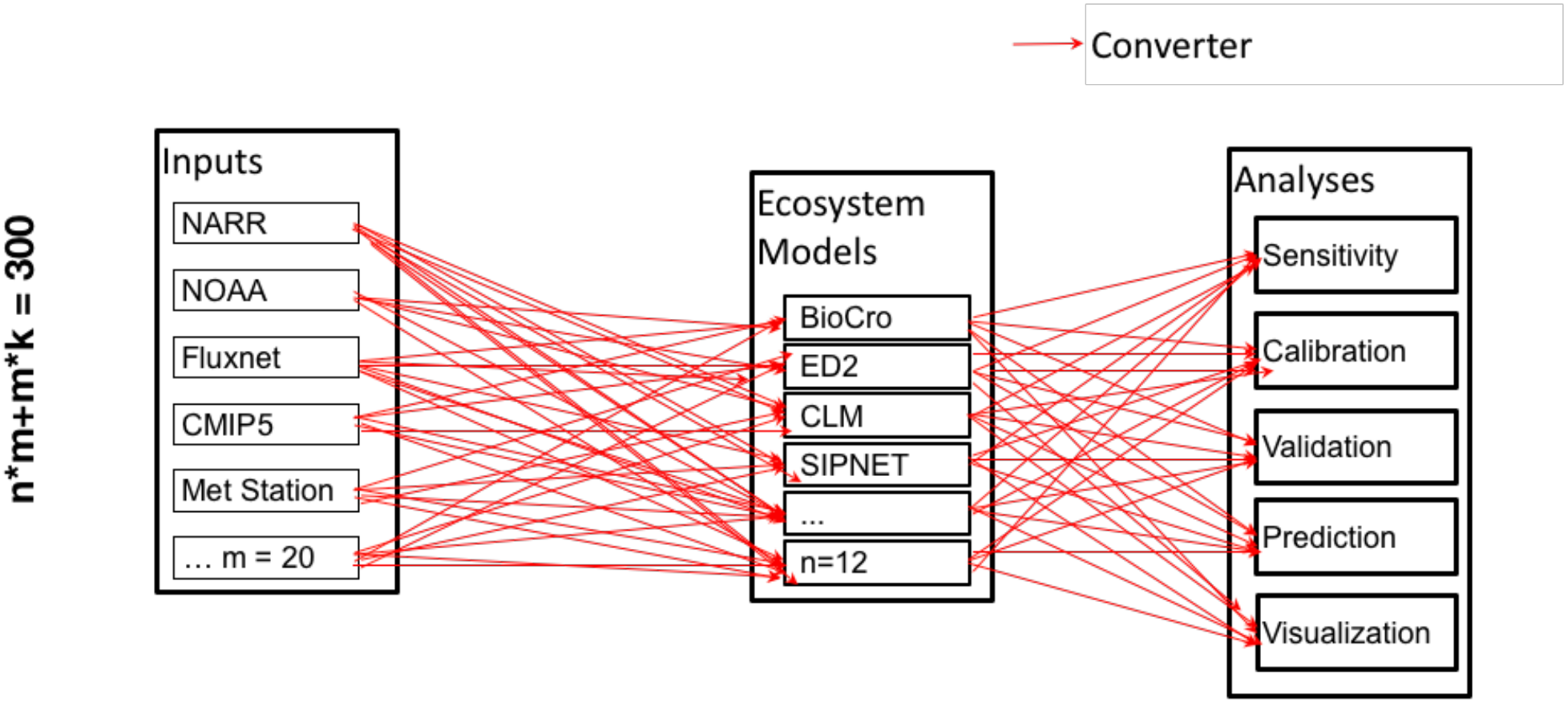


Alternative to a 'new' standard:

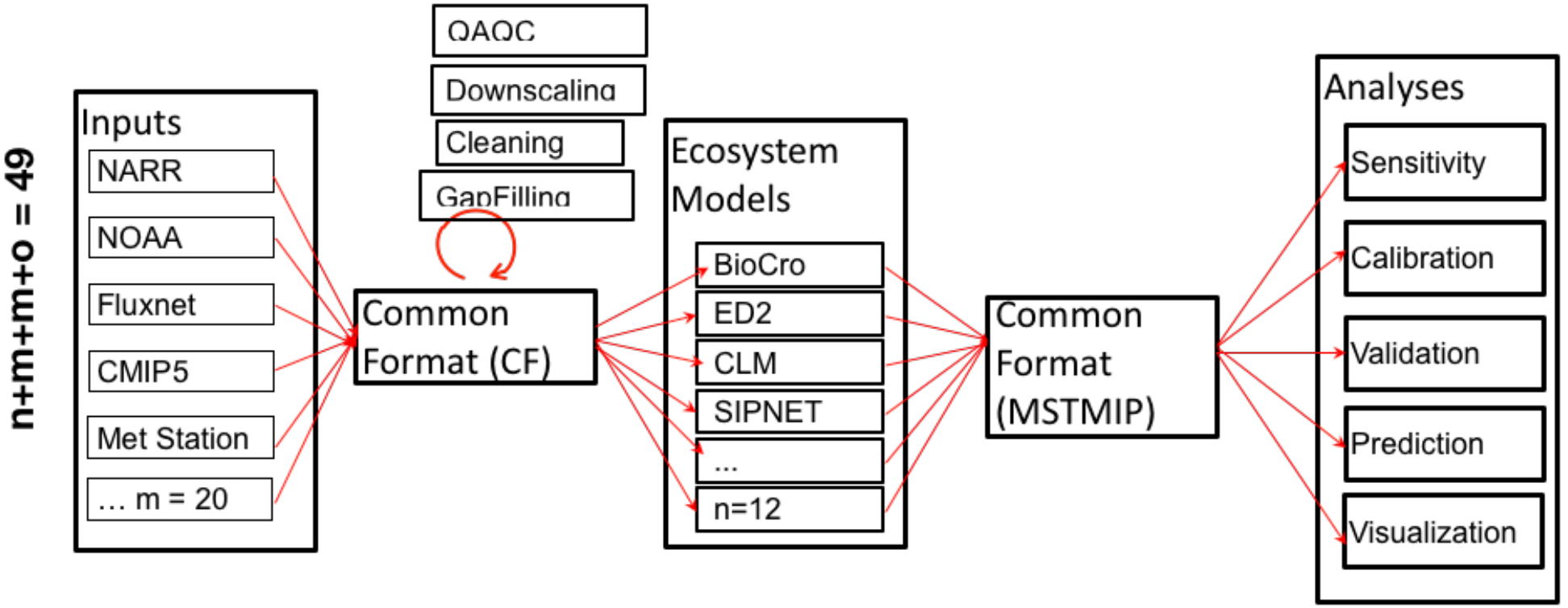
Combine, translate among existing standards.

https://xkcd.com/927/

Ecosystem Models Without Standards



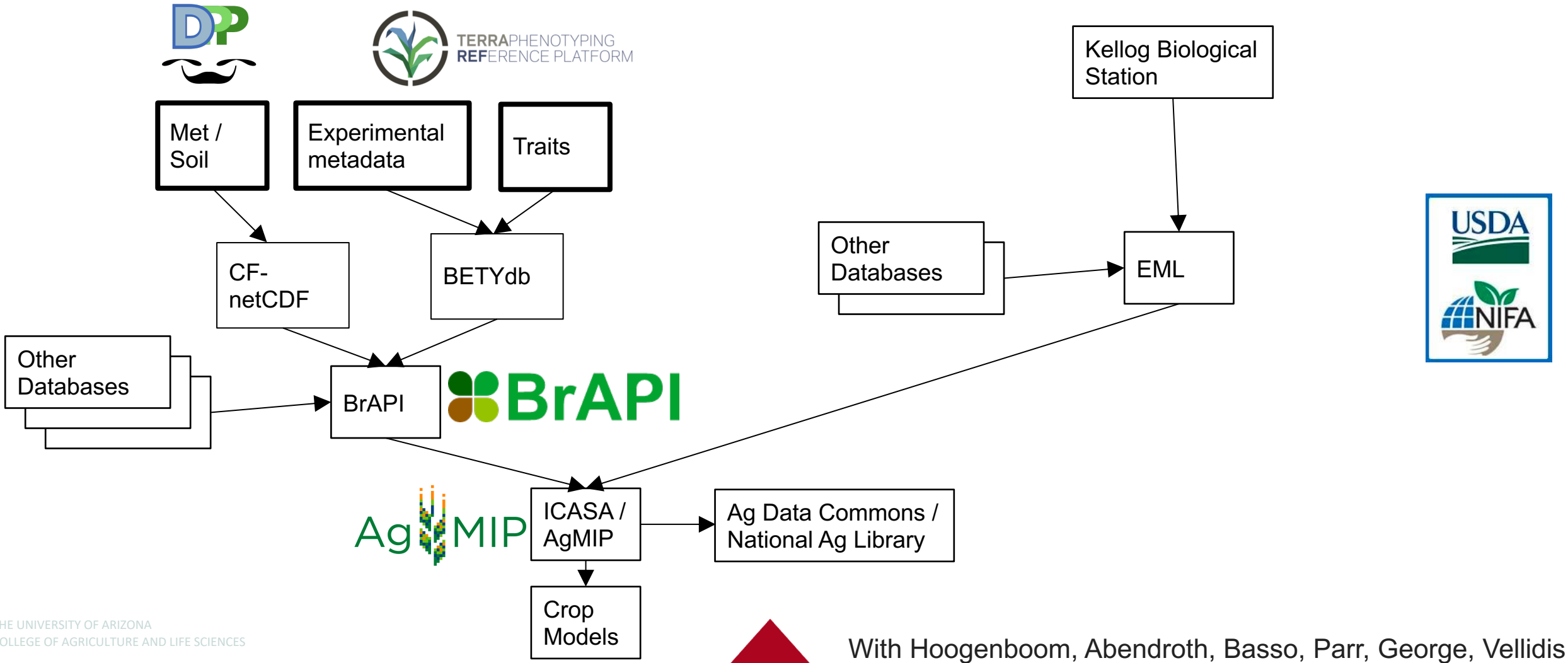
PEcAn + Standards: benefits of interoperability



PEcAn

pecanproject.org

Agricultural Research Data Network (ARDN)



With Hoogenboom, Abendroth, Basso, Parr, George, Vellidis

Outline:
Our Approach in 4 Ts

- Tools
- Translators
- **Tutorials**
- Templates

Tutorials:
Self Guided, Live Coding Webinars, Videos

https://terraref.org/tutorials

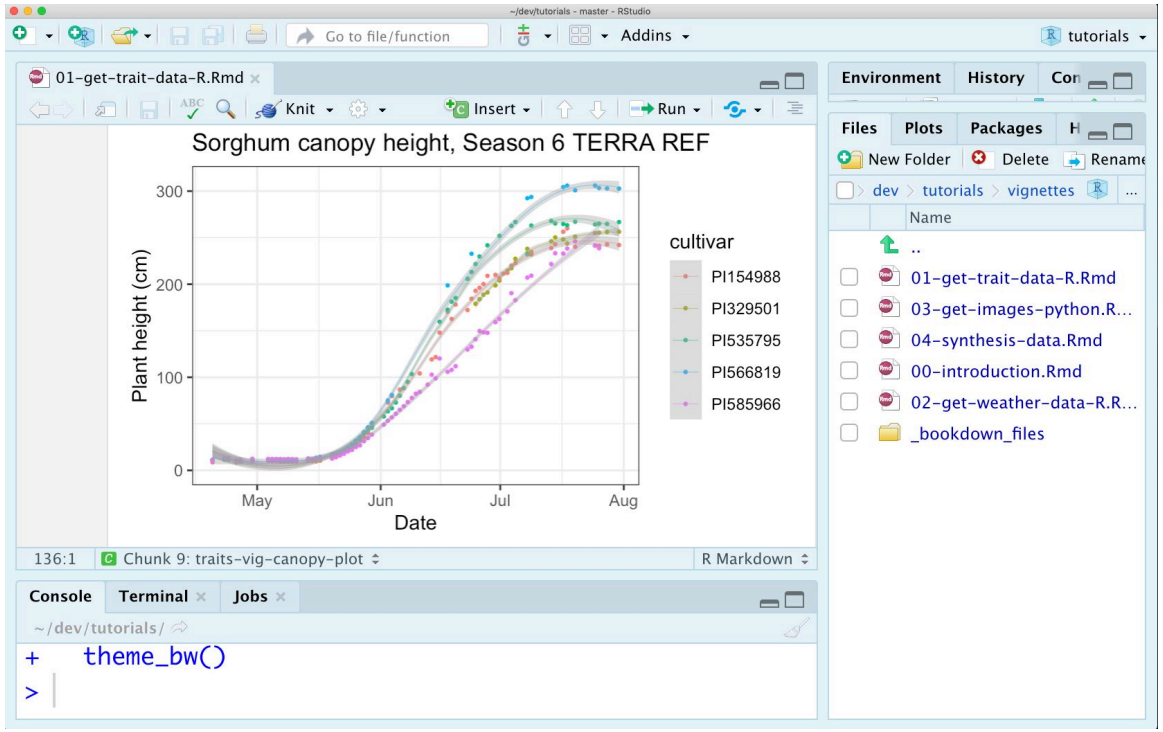
terraref.github.io/tutorials/index.html

TERRA REF Tutorials

David LeBauer and others

2020-01-06

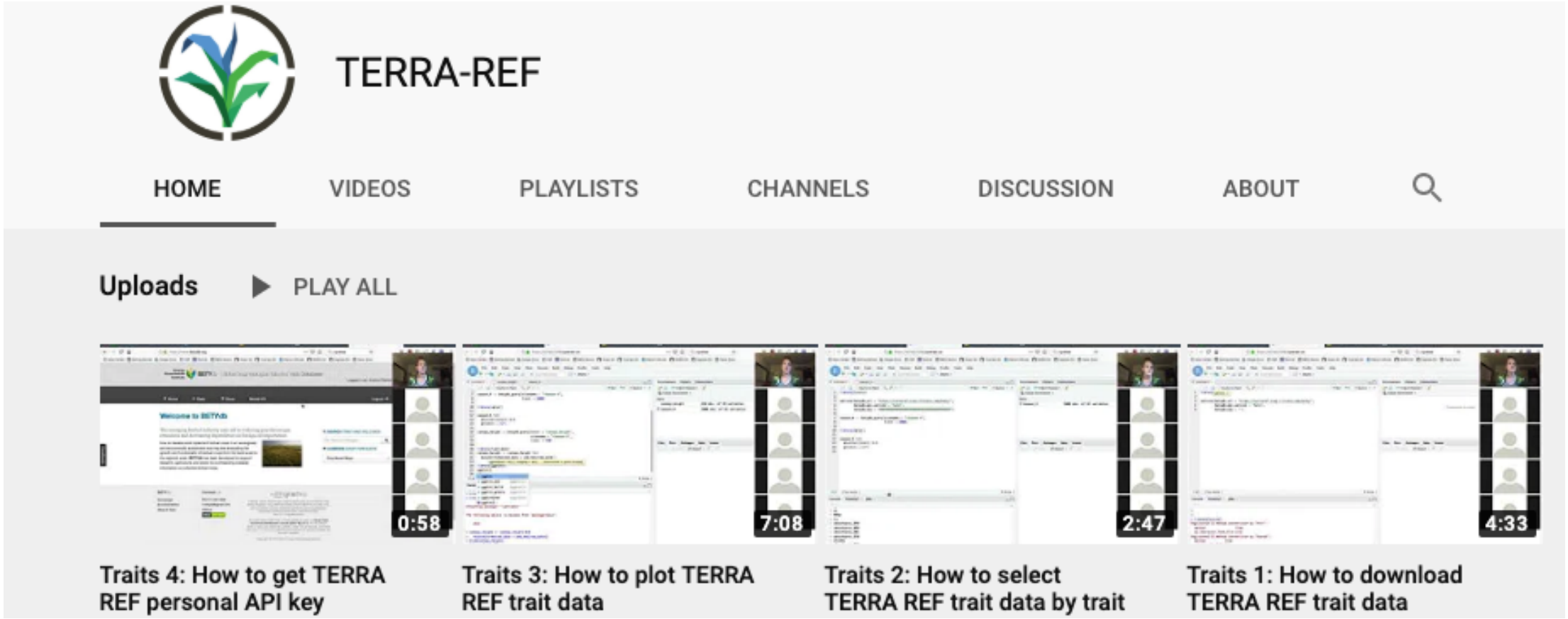
Chapter 1 Overview



Kristina Riemer, UA

CYVERSE

YouTube Videos

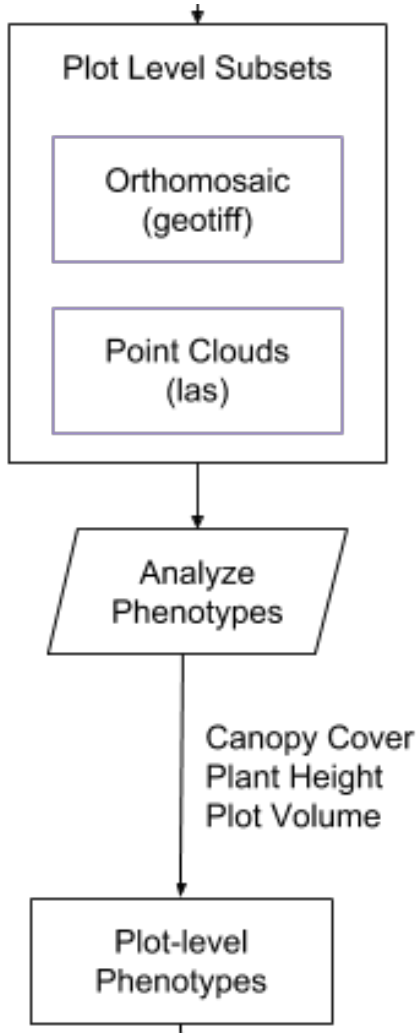


Outline:
Our Approach in 4 Ts

- Tools
- Translators
- Tutorials
- **Templates**

‘Transformer’ libraries

```
def calculate(data, metadata)
    do something
    return data, metadata
```



v1: github.com/terraref

v2: github.com/agPipeline

With Kooper, Burnette, Schnaufer

Templates

Guide users to best practices

- Standard formats and vocabularies
- Best practices
- Testing
- Documentation

Minimize Overhead

- Quickly integrate a new algorithm into a pipeline
- Versioning
- Sharing

Example template

```
def calculate(pxarray: np.ndarray):
    """Calculates one or more values from plot-level RGB data
    Arguments:
        pxarray: Array of RGB data for a single plot
    Return:
        Returns one or more calculated values
    """
    # ALGORITHM: replace the following lines with your algorithm
    channel_size = pxarray[:, :, 1].size

    # RETURN: replace the following return with your calculated values.
    return channel_size
```

github.com/AgPipeline/template-rgb-plot
algorithm_rgb.py

A Template for Plot level RGB data

1. Setup: Click the `Use this template` button in GitHub to make a copy of this repository (or run `git clone`)
2. Definitions: Fill in and modify the definitions in the algorithm_rgb.py file
3. Algorithm: Replace the code in the `calculate` function with your algorithm
4. Test: Run the `testing.py` script to run your algorithm and validate the results
5. Generate: Run `generate.py` to create a Dockerfile
6. Docker: Create a Docker image for your algorithm and publish it
7. Finishing: Finish up your development efforts

github.com/AgPipeline/template-rgb-plot

Conclusions

Reuseable software and data will accelerate science and engineering

Tools, Translators, Templates, and Tutorials enable open, synthetic science

More Information

David LeBauer

dlebauer@arizona.edu

datascience.cals.arizona.edu @digitalaggroup