OAK RIDGE National Laboratory

Automated metadata extraction: challenges and opportunities

Tyler J. Skluzacek[1], Kyle Chard[2], Ian Foster[2]

1 Data Lifecycle and Scalable Workflows Group, ORNL

2 University of Chicago & Data Science and Learning Division, ANL

ERROR '22

ORNL is managed by UT-Battelle LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY

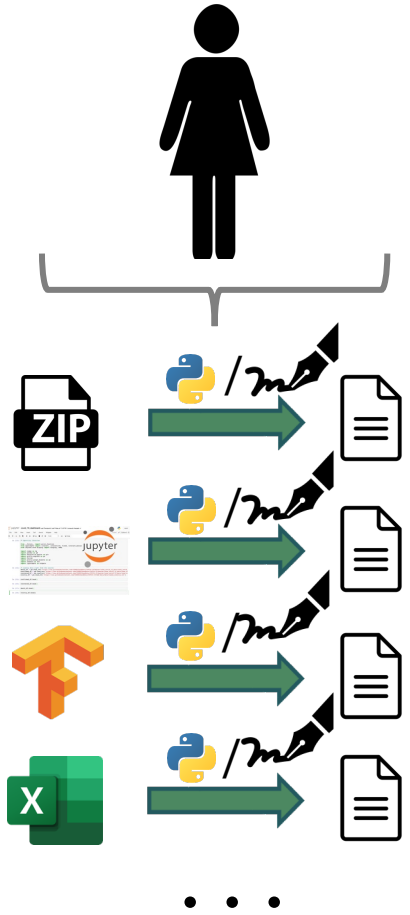
Metadata: *easier said than done*

Metadata: data about data

Research data lifecycle generates a snowball of research artifacts

• • •

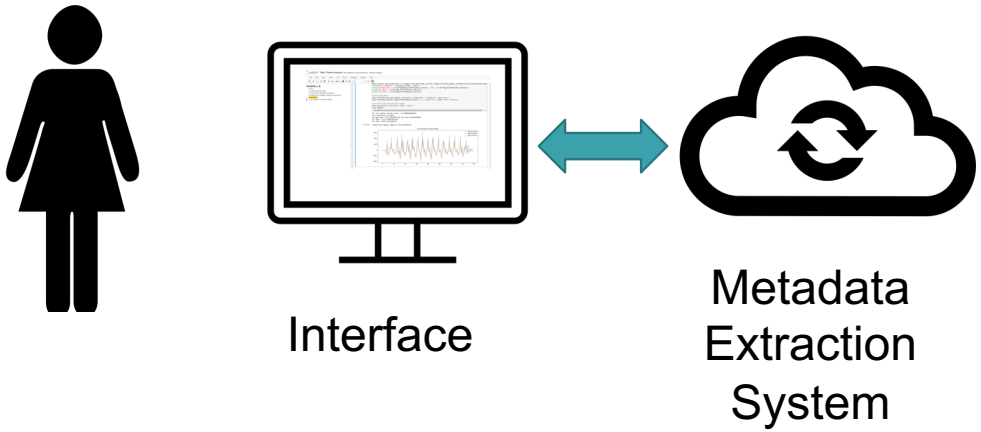
(some time later…)

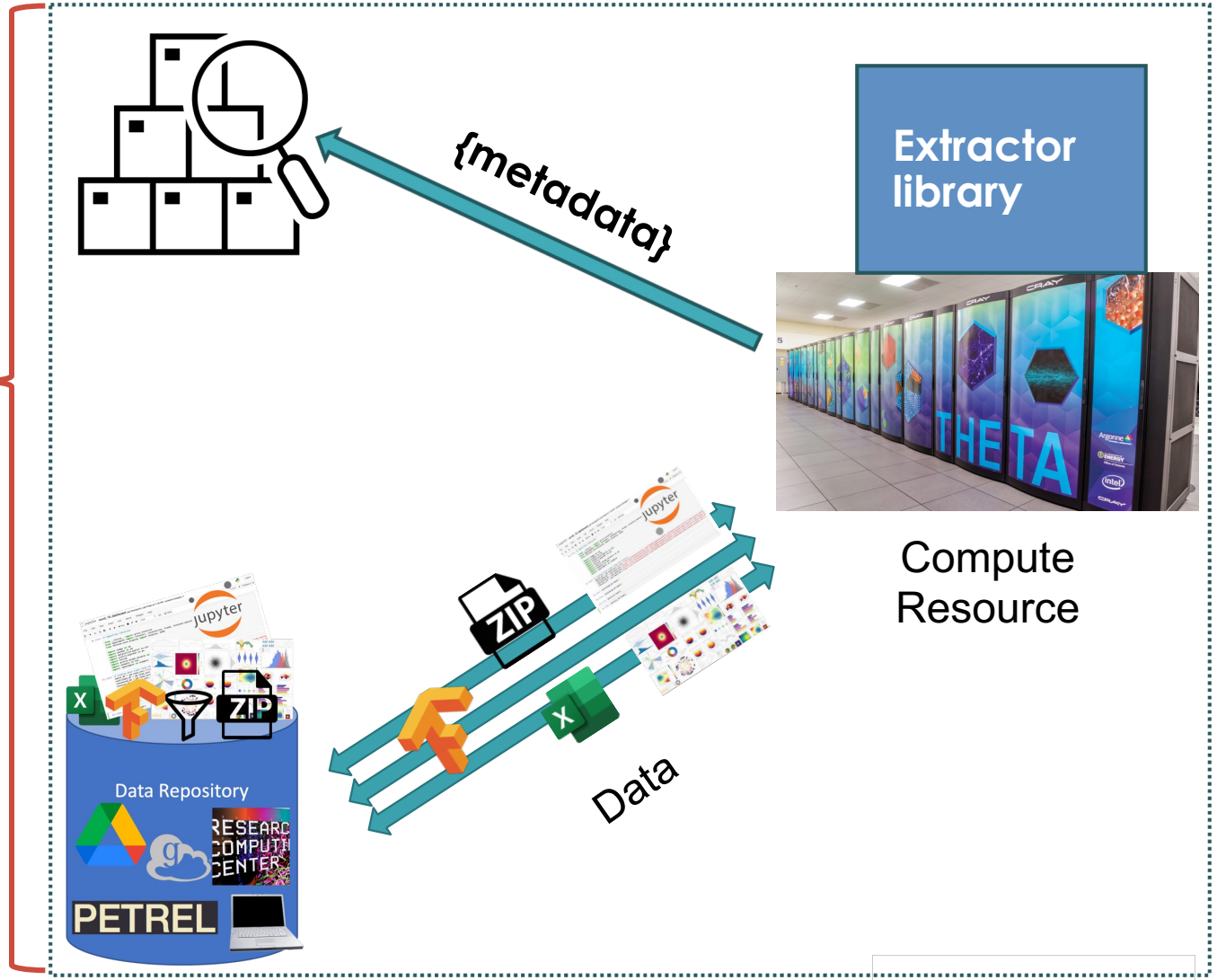


• • •

Extracting their metadata puts significant strain on the scientist

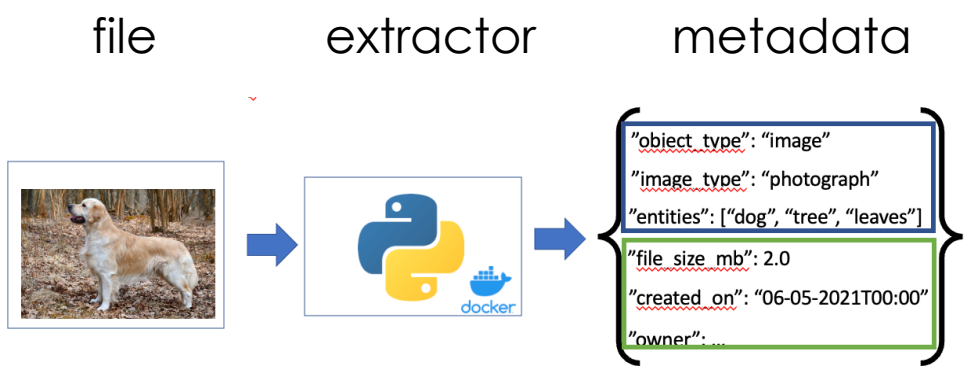
Automated metadata extraction enables scientists to **automatically mine value** from diverse science data

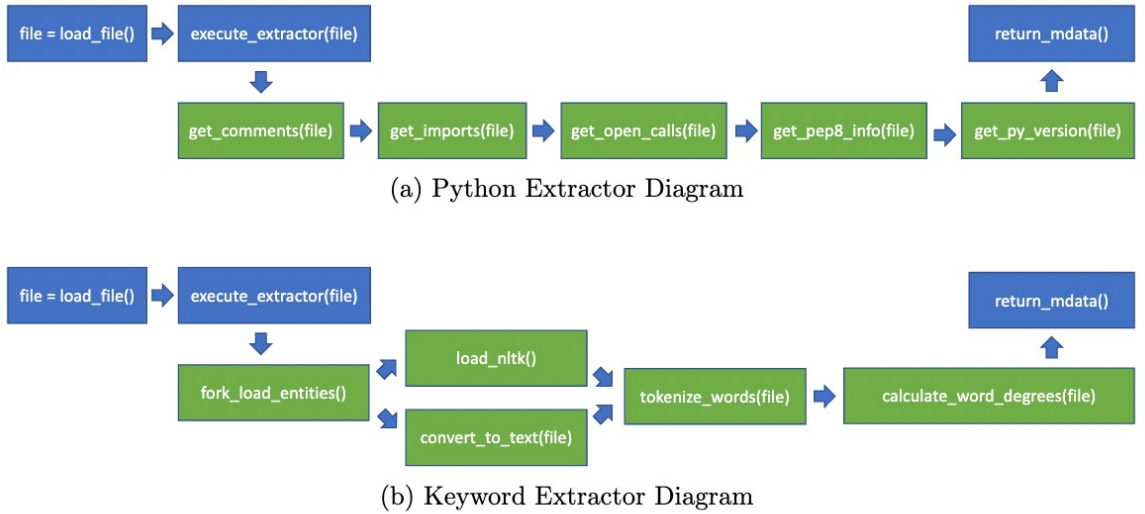




Automated metadata extraction system:
a computing system that mines metadata from data by leveraging computational resources

Extractors are "lightweight" programs that input a file and output metadata, ***for a given type of file***



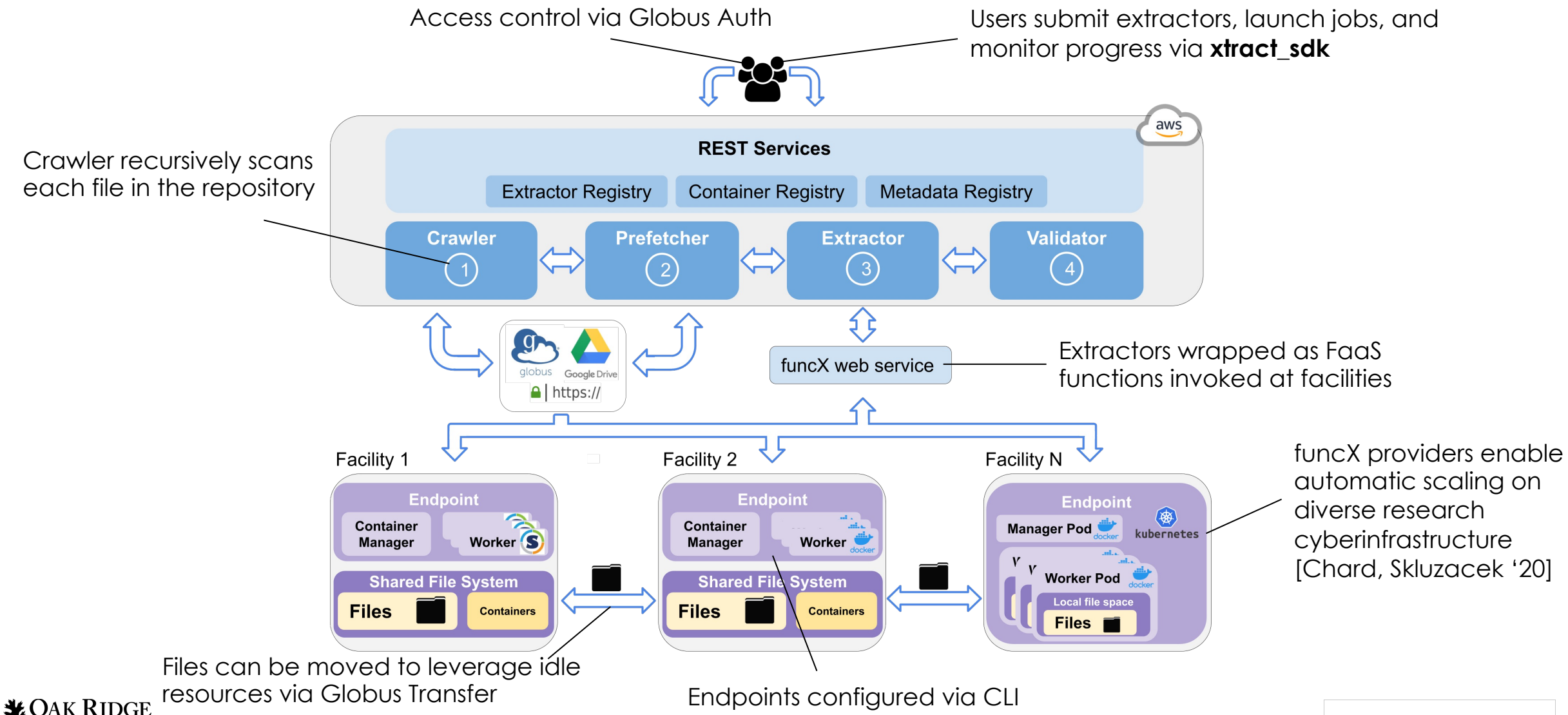


(a) Python Extractor Diagram

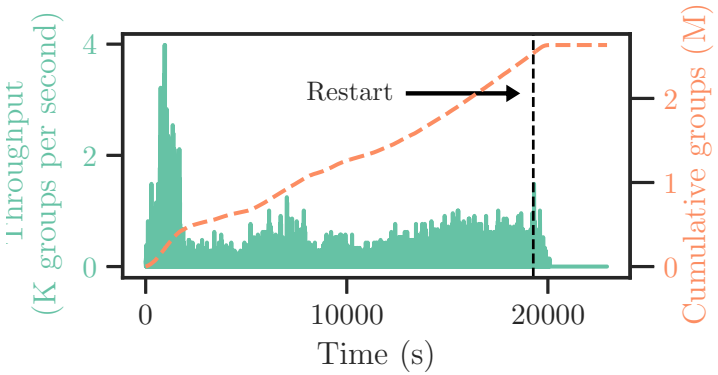
(b) Keyword Extractor Diagram

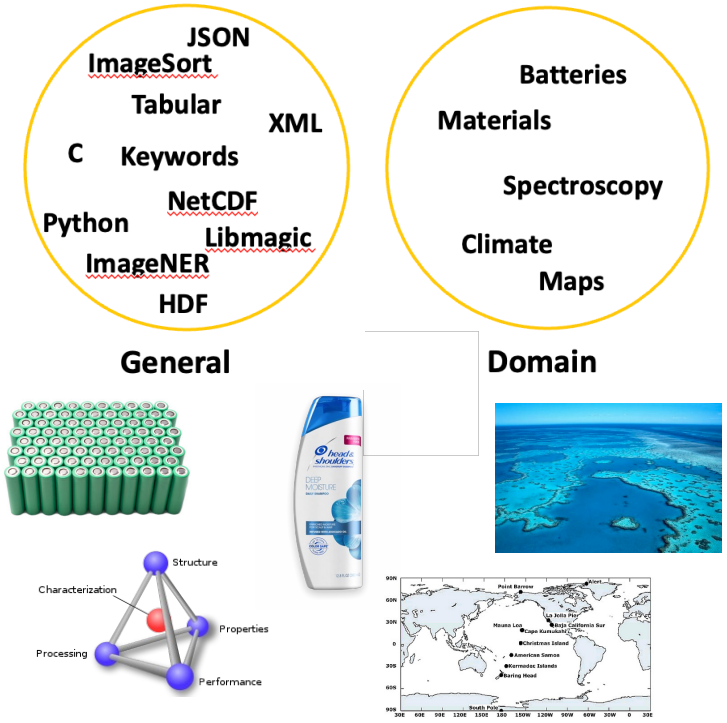
Figure 3.8: Workflow diagrams for `python` and `keyword` extractors. Functionalities present in all extractors (as part of the extractor creation library) are represented by blue boxes; extractor-specific functionalities in green.

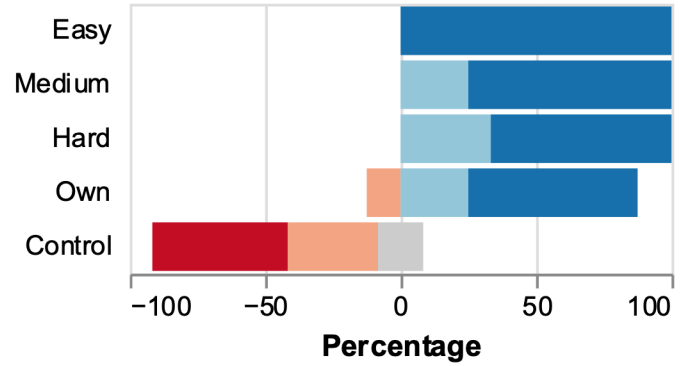
Xtract: the metadata extraction system for science



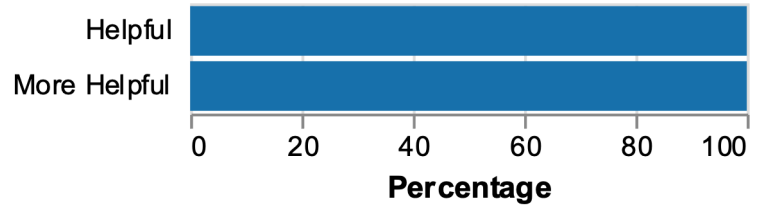
Metadata extraction can be scalable extensible useful







"These metadata collectively contain the attributes necessary to successfully complete this task"



"I have found these metadata ***{helpful | more helpful than my existing approaches}*** in navigating my data"

Xtract can process tens of millions of materials science files (19 TB) in just 6 hours.

Users create extractors to support research across a **breadth of disciplines and file types**.

Automatically extracted metadata enable users to **better navigate** complex data repos.

OAK RIDGE National Laboratory

In evaluating Xtract, we discovered multiple unsolved challenges in automatically extracting metadata

Divergent user perspectives

Users may require different metadata specifications

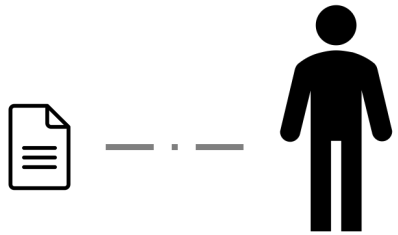
- Precision (decimal points)
- Timeliness (last extracted)
- Representation (graphs)
- Null substitution (NULL, -999)
- … and other quality metrics.

"this is great!" *"I can't use this"*

Evolving user requirements

User requirements will change as a result of new:

- Extraction methods (NLP)
- Standards (FAIR metrics)
- Relevance of data to a new instrument or domain



"Keyword analysis was great, but now there are also great tools for extracting sentiment"

Extractor library growth

As more users use an extraction system:

- Extractors overlap in functionality
- Compute hours are wasted performing 'overlap' tasks

Divergent data perspectives in context

We asked 6 users what metadata attributes were needed for their research workflows:

- **Users 1 and 2:** visually represent metadata on a graph so that users could "pull out quantities for specific parts of a voltage curve"
- **User 3:** "discover data that are similar enough to treat with the same analytical technique"

battery

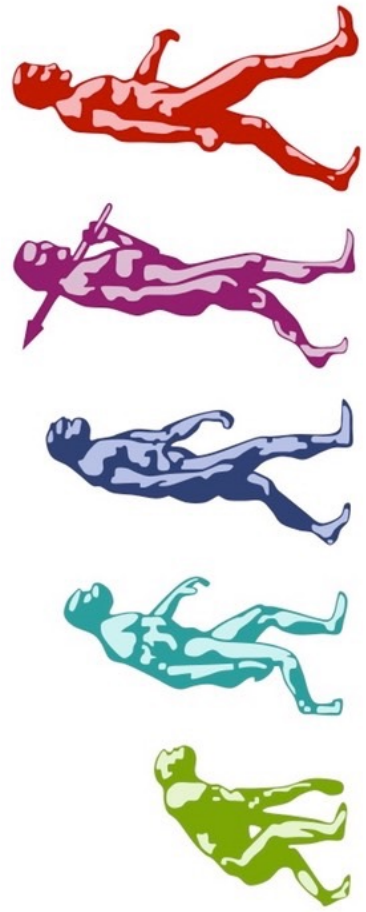
- **User 5:** empty detector field in data should be **auto-populated** in metadata
- **User 6:** empty detector field in data should be **left as "unknown"** in metadata

spectroscopy

Evolving user requirements in context

The value users realize from (meta)data should decrease over time, given new:

- **Data indexing standards**
  Our battery users want their metadata to fit a particular ontology that was published in 2022 [Clark, '22]
- **Use cases for existing data**
  User 3 want to search through old experimental data to find data for training new machine learning models
- **Metadata generation methods**
  Our spectroscopy users want to eventually use computer vision models to perform quality control on generated images



"Skluzacek's law of diminishing (meta)data utility"

Extractor library growth in context

Extractors overlap in functionality

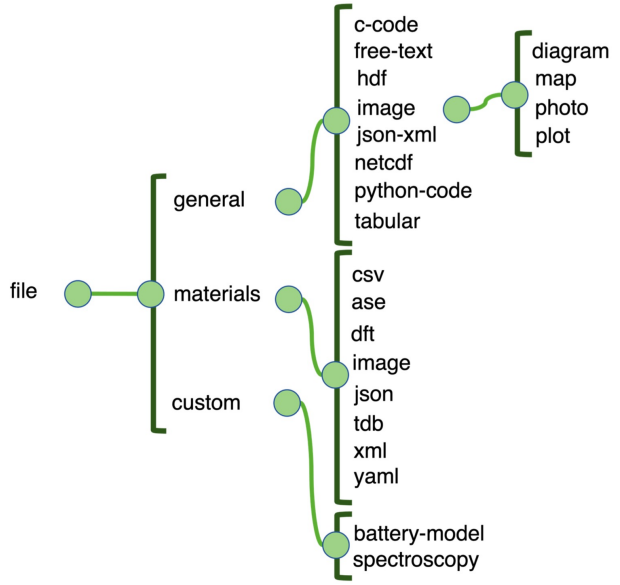
e.g., "netcdf" and "tabular" both calculate aggregates of a data series; adds developer effort and more exposure for bugs/errors

Compute hours wasted

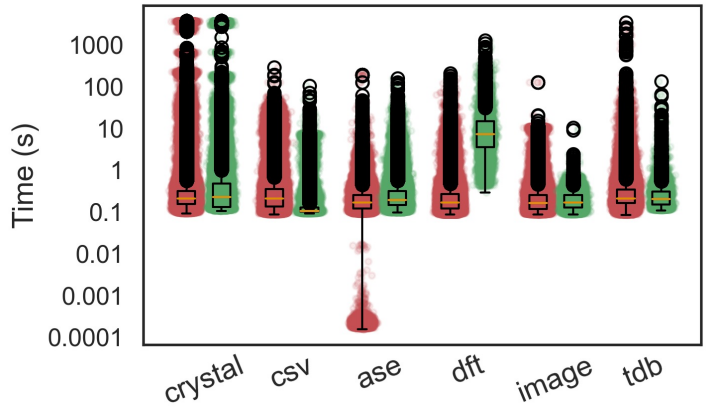
When executing extractors on the 2.2 million files in the Materials Data Facility, we want to find ways to minimize time spent performing redundant calculations

Difficult to orchestrate

How can I prioritize which 'similar' extractor is better given limited budget?



Xtract's extractor library



Time taken to execute all extractors on each file in Materials Data Facility (MDF)

Sum of **correct:**	4,373 core hours
Sum of **incorrect**:	11,898 core hours

Now, two potential research directions to help address these issues…

Direction 1: enable multi-context metadata views

Users often interact with (meta)data via a **search index**

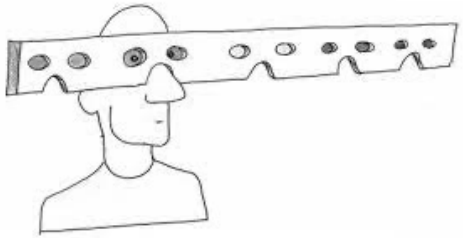
- **User A** wants to search for "birds"
  - Returns any records of birds
- **User B** wants to see records for "gavia immer" (the common loon)
  - Returns only records of a specific type of bird

Hierarchical data models allow varied search specificity for images [Cai, '04], text documents [Kuang, '11], and numeric data [Hoang, '20].

Why should extraction systems prioritize multiple views over the same (meta)data?

Container explosion (and scope) relief
if an extractor can assist both User A and B, then only need 1; decrease programming effort across users

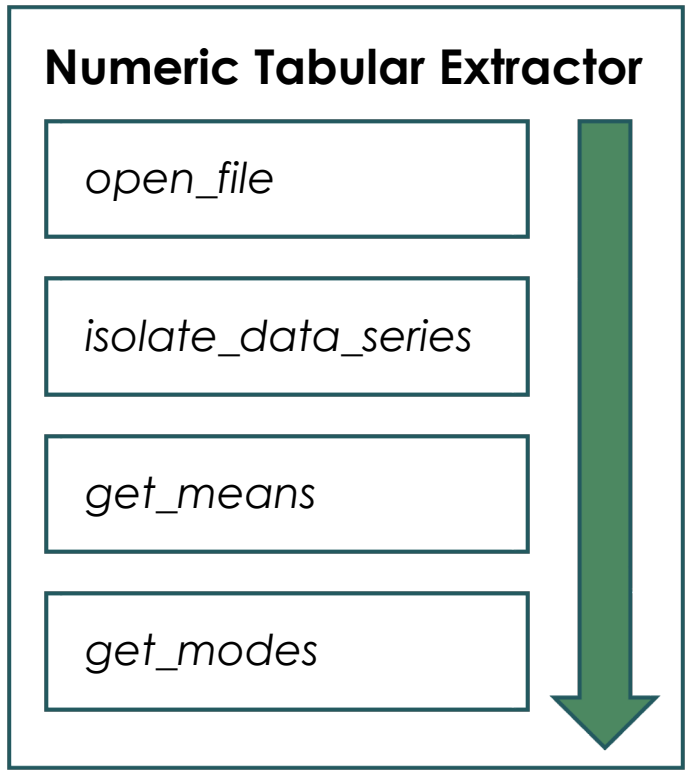
Easily adapt to temporal requirement changes
if new standards are released, could adapt existing metadata for the new use case



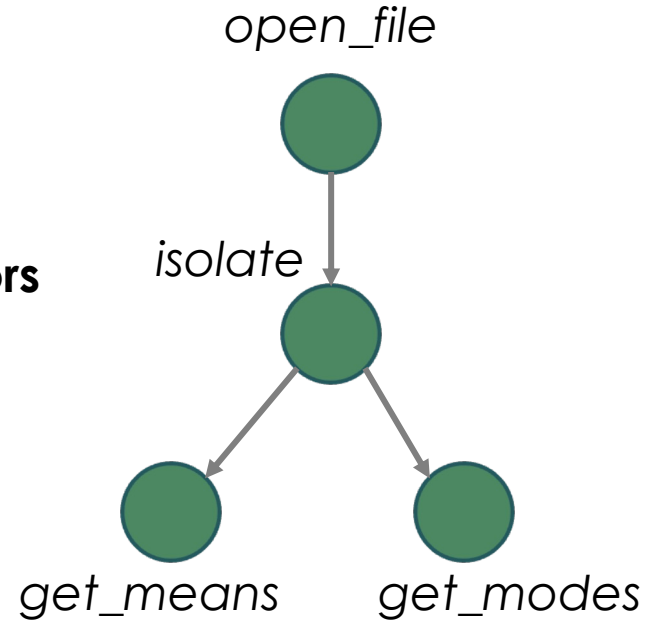
Adoption
if the existing extractor library can appease users, more users will leverage extraction systems

Direction 2: decompose extractors into **microextractors**

Microextractors: modular, shareable, stateful software abstractions for specific extractor functionalities



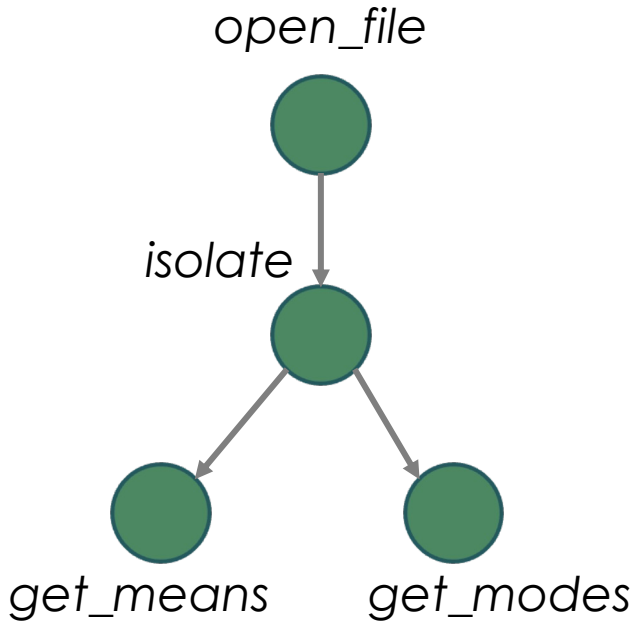
…as microextractors



What exactly does this solve?

Why should metadata extraction systems adopt microextractors (ME)?

1. Shareable, standard extraction logic
2. Clear data flow; programming ease
3. Can easily add or alter one ME and rerun only partial DAG
4. ME enables "merging multiple extractors into 1"
5. Conducive to hierarchical model



Summary

Modern extraction systems hampered by

- users needing ‘different things’ from metadata
- too many extractors (extractor explosion problem)
- fading metadata quality over time

These issues could be alleviated by

- multi-context metadata views
- microextractors
- intelligent extraction methods that minimize user effort

Let’s get to work!

Thank you!

If you would like to learn more, please reach out:

Tyler J. Skluzacek
Research Scientist, Oak Ridge National Lab
skluzacektj@ornl.gov