# Legacy

## Preservation and Scientific Software

Daina Bouquin

Harvard-Smithsonian Center for Astrophysics

daina.bouquin@cfa.harvard.edu

@dainabouquin

# I'm a librarian.

Harvard University

Smithsonian Institution

Some things that I work on:

- arXiv Next Generation IT Advisory Group
- CfA Scientific Computation Advisory Committee
- Harvard University Science Libraries Council
- Mozilla Foundation Open Leaders Advisor
- Software Preservation Network Steering Committee
- Unified Astronomy Thesaurus Steering Committee

# Semantics

The relationships between signifiers
and what they stand for in reality.

How we understand what something means.

# Lexicon

Vocabulary of a person, language, or branch of knowledge.

(contains the signifiers)

# Legacy

1. Something superseded but difficult to replace.
2. Something received from an ancestor or predecessor.
3. Having a privilege or special status.

NICOLAI COPERNICI

net, in quo terram cum orbe lunari tanquam epicyclo contineri diximus. Quinto loco Venus nono mense reducitur. Sextum deniq; locum Mercurius tenet, octuaginta dierum spacio circũ currens. In medio uero omnium residet Sol. Quis enim in hoc



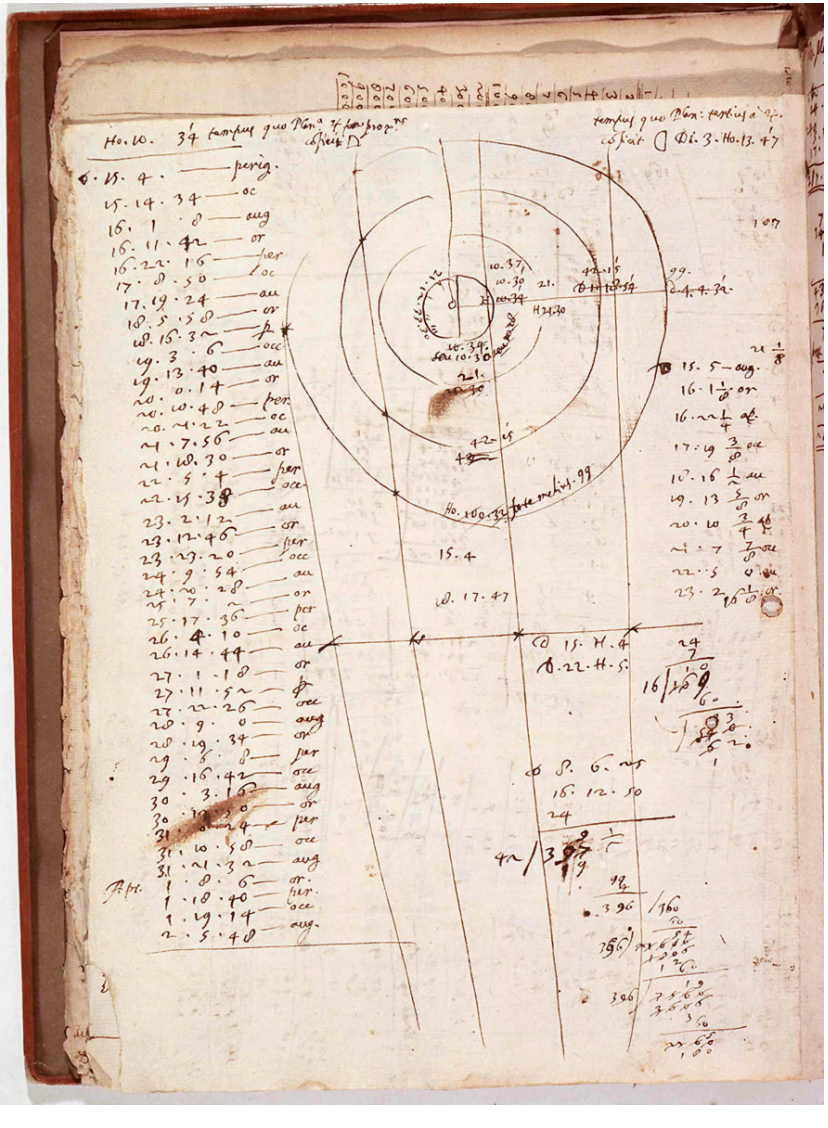
pulcherimo templo lampadem hanc in alio uel meliori loco poneret, quàm unde totum simul possit illuminare? Siquidem non inepte quidam lucernam mundi, alij mentem, alij rectorem uocant. Trimegistus uisibilem Deum, Sophoclis Electra intuentẽ omnia. Ita profecto tanquam in solio regali Sol residens circum agentem gubernat Astrorum familiam. Tellus quoq; minime fraudatur lunari ministerio, sed ut Aristoteles de animalibus ait, maximã Luna cũ terra cognationẽ habet. Concipit interea à Sole terra, & impregnatur annuo partu. Inuenimus igitur sub

hac

# Sometimes all three

- Superseded but difficult to replace.
- Received from an ancestor or predecessor.
- Has a privilege or special status.

Copernicus, N. (1543). *Nicolai Copernici Torinensis De revolutionibus orbium cœlestium libri vi*. Norimbergae: Apud Ioh. Petreium.



Galilei, G. (1610). Osservazioni e calcoli relativi ai Pianeti Medicei.

Galileo (67 years later)

Threatened with torture

Imprisoned for life

Burned his books

## Galileo didn't know his chicken scratch would be important.

(Largely seen as the birth of observational astronomy and the scientific method)

## People didn't care that much about Copernicus' model.

(It was easy to dismiss)

Meaning is collective agreement about a specific thing at a specific time.

**Semantic meaning is not static.**


Humphrey, S.D. Multiple Exposures of the Moon: Nine Exposures, daguerreotype, 1849.

Sometimes it's more about privilege.

Earliest image of the moon extant.

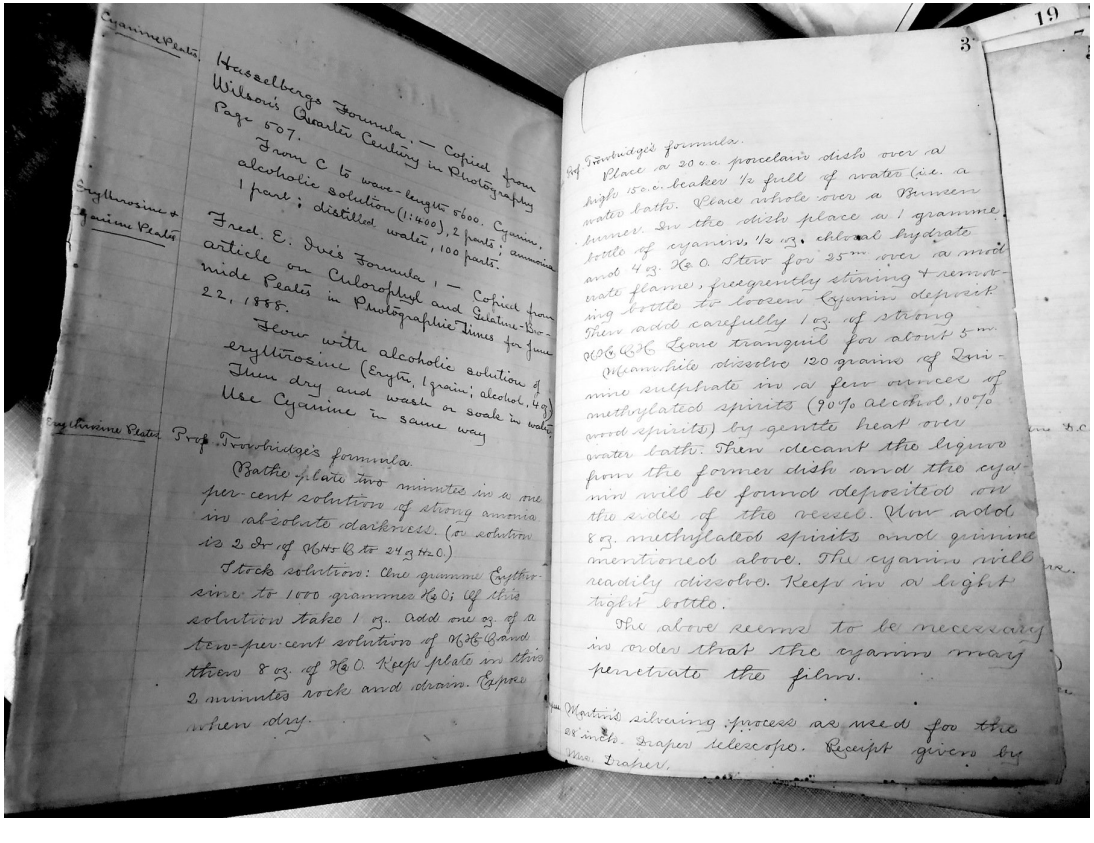
There could have been other images of the moon.

# Gift to the President of Harvard at the time.

(This is it on my desk.)

# Provenance

means **context**



Daguerreotype "Recipe book"

Matters because of its relationship to the daguerreotype.

Provenance guides prioritization for curation.

**Curation is work.**

# All objects need curation.

Everything will break.

Things need to be reformatted.

Entire fields are being developed in response:

# Digital Forensics

Stabilizing and recovering data from digital media.

The creators of these objects did not need to care about the historic meaning of their work.

Provenance could be determined so we gave these things meaning and prioritized them for curation.

We know what to call these things and we know how to take care of them.


# We don't have norms yet for how to give things like this semantic meaning.

- Superseded.
- Received from a predecessor.

Knowledge is more than books and articles.


# I have very little provenance.

When does something like this matter?

Who decides?

How do we semantically link this to anything?

How would someone find it?

(What do I call it?)

# Metadata

Mechanisms for modeling relationships between the information gathered from provenancial sources.

# Schema

Logical framework where
semantic metadata can be recorded.


The fact that a thing exists in a place at a time does not give it meaning or make it identifiable.

I can describe this thing but give it little meaning.

Cultural norms prevent me from throwing this away.

(I would feel bad)

# "I bet there's a paper."

A paper could provide some provenance.

Our schema should definitely have a field where we can identify a relevant paper.

Remember though:

- It would take time and effort to find a paper.
- If the paper exists it is probably behind a paywall (privilege).
- I might not be legally able to own or distribute the paper (publishing models).

# Who got to be an author on the paper?

Who didn't?

Is the "author" of the paper identical to the "author" of this thing?

Who gets credit?

# This object is not a paper.

## Disambiguation

We need to be able to directly identify the object to distinguish between the object and our sources of provenance.

## What are the nodes in our semantic network?

What if this thing was software?

# Some Human Readable Metadata

## corner.py

*Make some beautiful corner plots.*

**Corner plot /ˈkôrnər plät/ (noun):**

> An illustrative representation of different projections of samples in high dimensional spaces. It is awesome. I promise.
>
> *Synonyms: scatterplot matrix, pairs plot, draftsman's display*

Development of *corner* happens on GitHub so you can raise any issues you have there. *corner* has been used extensively in the astronomical literature and it has occasionally been cited as `corner.py` or using its previous name `triangle.py`.

build passing | coverage 27% | license BSD | DOI 10.5281/zenodo.53155

## corner.py v2.0.0

dfm released this on May 26, 2016 · 35 commits to master since this release

Version 2 of corner.py is now tested, documented, and citable.

Assets 2

- Source code (zip)
- Source code (tar.gz)

## corner.py v1.0.2

dfm released this on Feb 11, 2016 · 61 commits to master since this release

Renamed to corner.py and many other updates.

Assets 2

## triangle.py v0.1.1

dfm released this on Jul 24, 2014 · 129 commits to master since this release

This is a citable release with a better name.

Assets 2

## triangle.py v0.1

dfm released this on Jun 19, 2014 · 131 commits to master since this release

This release is citable.

# What makes something citable?

Daina,

I took the liberty of looking you up in the faculty directory. Thank you for looking into the code for the [redacted] computer program. The PI for the study was [redacted].

If you do find the code, I can arrange for it to be loaned to one of my colleagues at SAO.

Thank you, and if you need any more information from me, please let me know.

Hello,

At the end of the attached paper, there is a link to a computer code

[redacted]

The link does not work any more. Is it still possible to get the code?

# I want you to have a scientific legacy.

Software will be the foundation on which future generations must build new knowledge.

## Your work is someone's heritage.

Code is speech.

# "It's on GitHub."

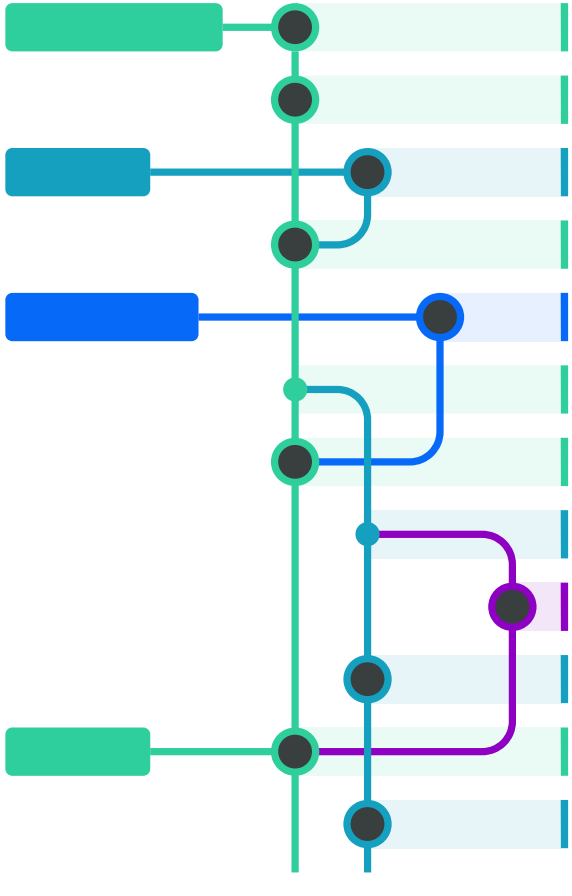
Just means it's in a place right now.

## Identification

Unambiguous way to point at a specific thing in a specific place at a specific time.

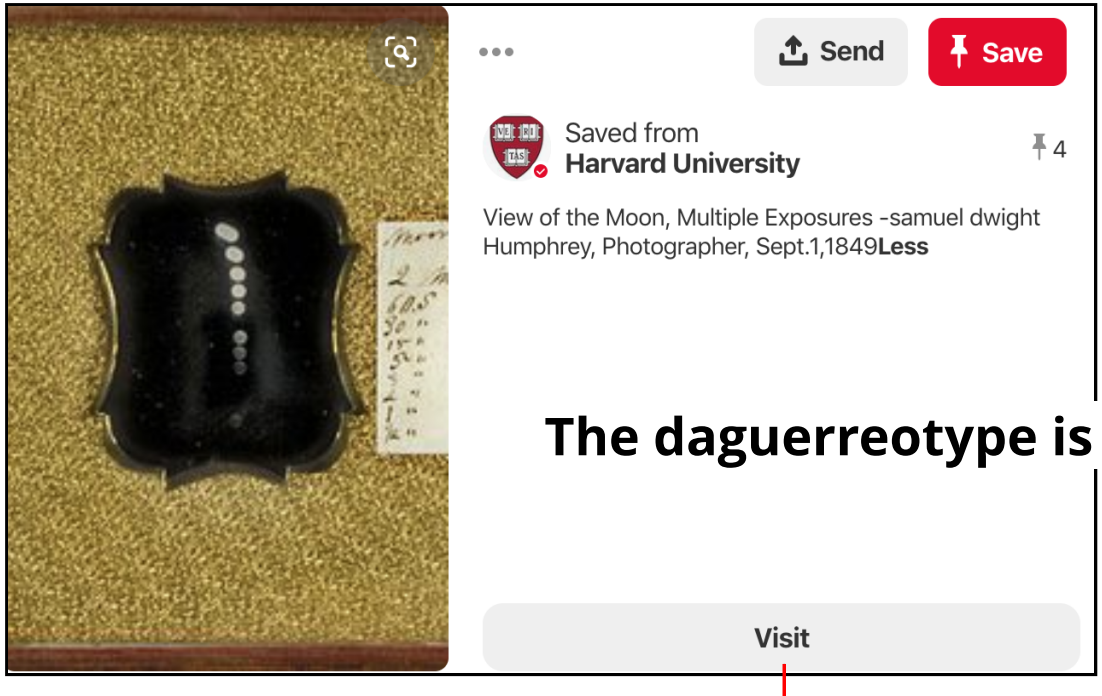
## Location

Where the thing you are pointing at is at a specific time.



# Born Networked

Exists in many ways
in many places over time.




**The daguerreotype is also on Pinterest.**

# Not Found

The requested URL /daguerreotypes/highlight10.html was not found on this server.

This page doesn't exist there anymore.

It also didn't tell me **where the real thing is**.

Is it on my desk or in a vault?

COLLECTION HIGHLIGHTS

< PREVIOUS | NEXT >

View of the moon, multiple exposures
Samuel Dwight Humphrey, photographer
September 1, 1849
sixth plate
vatory

**URL**

**Uniform Resource Locator**

Locations change.

Provenance changes.

Meaning changes.

IMAGE

View of the moon, multiple exposures

Humphrey, Samuel Dwight [photographer]

September 1, 1849

VIEW ONLINE ›

Send to

PERMALINI | E-MAIL | CITATION | EXPORT RIS | EXPORT BIBTEX | PRINT | RESERVES LIST

Details

| | |
|---|---|
| **Title** | View of the moon, multiple exposures |
| **Author / Creator** | Humphrey, Samuel Dwight [photographer] › |
| **Description** | Multiple exposures of the moon. Nine exposures ranging from 2 minutes to 1/2 second.<br>*Materials/Techniques:* daguerreotype<br>*Dimensions:* sixth plate |
| **Notes** | Inscription: Embossed on case: "Daguerreotype of Moon taken by S. D. Humphrey. Canandaigua. Sp. 1, 1849." Paper label on mat: "Moon 2 M, 60 S., 30 S, 15 S, 5 S, 3 S, 2 S, 1 S, 1/2 S." |
| **Subjects** | moon › |
| **Form / genre** | photographs › |
| **Use restrictions** | Harvard College Observatory Library: This image may not be reproduced or transmitted in any form or by any means, electronic or mechanical, without permission in writing from the Harvard College Observatory. |
| **Repository** | Harvard College Observatory Library OB-1 |
| **Creation Date** | September 1, 1849 |
| **HOLLIS number** | olvwork124646 |
| **Permalink** | http://id.lib.harvard.edu/via/olvwork124646/catalog |
| **Source** | HVD - Images |

View source (MARC) ›

**Identification**

attached to machine actionable metadata

olvwork124646.xml


```xml
Source record page
<?xml version="1.0" encoding="UTF-8"?>
<viaRecord images="true" numberOfImages="1" numberOfSubworks="0" numberOfSurrogates="0" originalAtHarvard="true" recordSize="1764" sortCreator="Humphrey, Samuel Dwight" sortDate="1849" sortTitle="View of the moon, multiple exposures" sortWorktype="photographs">
    <recordId altRecordId="ss_8000175539">olvwork124646</recordId>
    <work>
        <image xmlns:xlink="http://www.w3.org/TR/xlink" altComponentID="4091964" componentID="W124646_1" restrictedImage="true" xlink:href="http://nrs.harvard.edu/urn-3:FCOR.HCO:31341">
            <thumbnail xlink:href="http://nrs.harvard.edu/urn-3:FCOR.HCO:164987"/>
        </image>
        <title>
            <textElement>View of the moon, multiple exposures</textElement>
        </title>
        <workType>photographs</workType>
        <creator>
            <nameElement>Humphrey, Samuel Dwight</nameElement>
            <role>photographer</role>
            <namedates>Humphrey, Samuel Dwight</namedates>
        </creator>
        <structuredDate>
            <beginDate>1849</beginDate>
            <endDate>1849</endDate>
        </structuredDate>
        <freeDate>September 1, 1849</freeDate>
        <description>Multiple exposures of the moon.  Nine exposures ranging from 2 minutes to 1/2 second.</description>
        <dimensions>sixth plate</dimensions>
        <topic>
            <term>moon</term>
        </topic>
        <materials>daguerreotype</materials>
        <notes>Inscription: Embossed on case: "Daguerreotype of Moon taken by S. D. Humphrey.  Canandaigua.  Sp. 1, 1849."  Paper label on mat: "Moon 2 M, 60 S., 30 S, 15 S, 5 S, 3 S, 2 S, 1 S, 1/2 S."</notes>
        <useRestrictions>Harvard College Observatory Library: This image may not be reproduced or transmitted in any form or by any means, electronic or mechanical, without permission in writing from the Harvard College Observatory.</useRestrictions>
        <repository>
            <repositoryName>Harvard College Observatory Library</repositoryName>
            <number>OB-1</number>
        </repository>
    </work>
</viaRecord>
```

# Identification

Identifier

DOI

URI

Bibcode

arXiv ID

etc.

# Location

Locator

URL

# URL

https://github.com/dfm/corner.py

was

https://github.com/dfm/triangle.py

Changes over time.

The meaning you are trying to express now will be different from what will be located at this URL later.

**This is not what you cite because this has no unambiguous meaning.**

Cite the **DOI** for the specific version of the thing you want to cite.

# corner.py

*Make some beautiful corner plots.*

**Corner plot /ˈkôrnər plät/ (noun):**

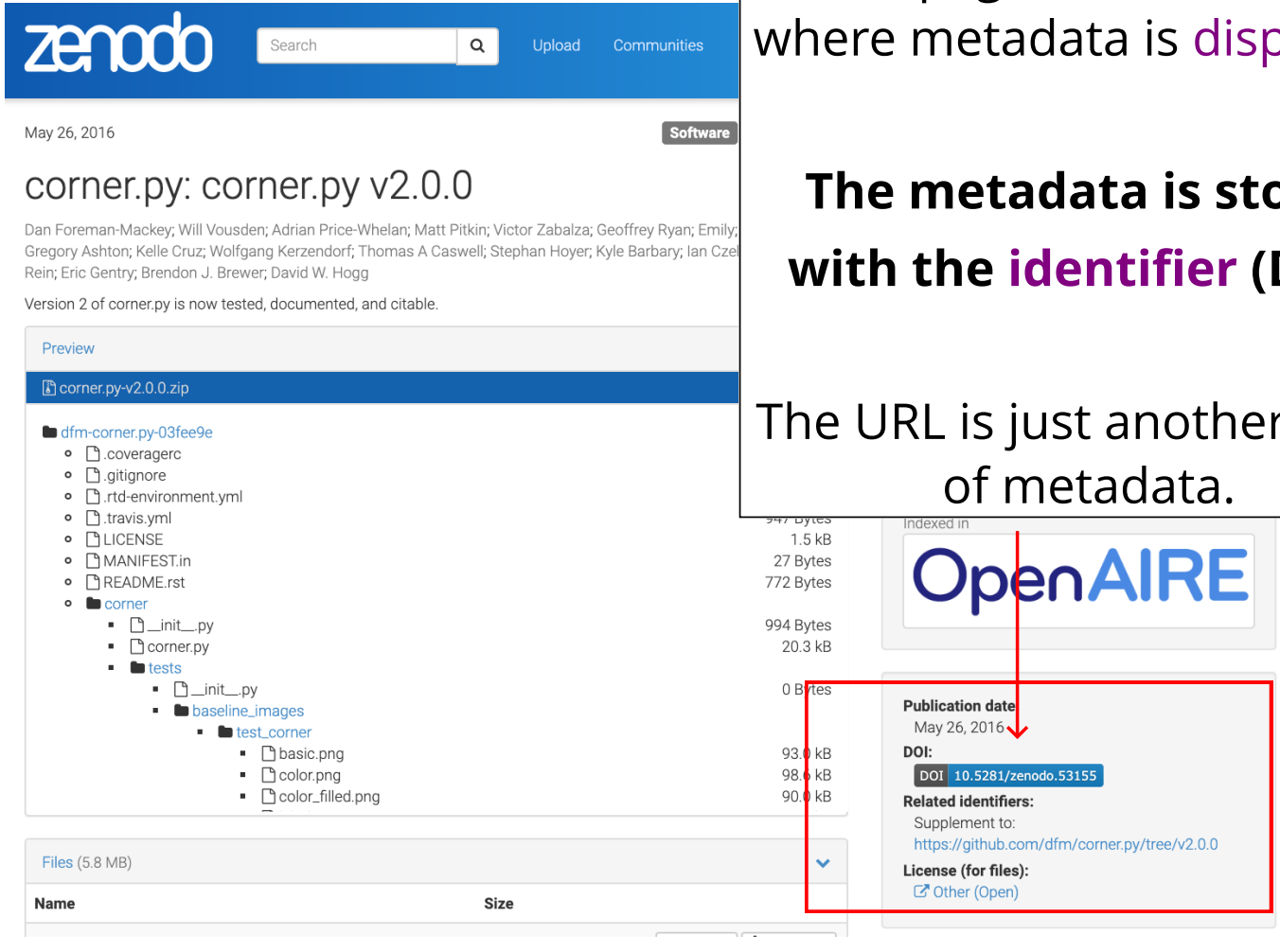
> An illustrative representation of different projections of samples in high dimensional spaces. It is awesome. I promise.
>
> *Synonyms: scatterplot matrix, pairs plot, draftsman's display*

Development of *corner* happens on GitHub so you can raise any issues you have there. *corner* has been used extensively in the astronomical literature and it has occasionally been cited as `corner.py` or using its previous name `triangle.py`.

build passing | coverage 27% | license BSD | DOI 10.5281/zenodo.53155

You already do this with papers.

This page has a URL: https://zenodo.org/record/53155




This page is an interface where metadata is displayed.

**The metadata is stored with the identifier (DOI).**

The URL is just another piece of metadata.

# DOIs are not magic

DOIs are resolvable.

They are bound to metadata.

Minted by a registry responsible for curating location metadata.

Resolves to a tombstone.

## Export

BibTeX CSL DataCite
Dublin Core JSON
JSON-LD MARCXML

## Versions

| | |
|---|---|
| Version v2.0.0<br>10.5281/zenodo.53155 | May 26, 2016 |
| Version v1.0.2<br>10.5281/zenodo.45906 | Feb 11, 2016 |

**Cite all versions?** You can cite all versions by using the DOI 10.5281/zenodo.591491. This DOI represents all versions, and will always resolve to the latest one. Read more.

Archives mint identifiers and curate metadata to ensure your work is findable and has meaning that can change over time.

Summary: Identifiers let us unambiguously point and assign semantic meanings with metadata.

# We can use metadata to make it clear that this is a record for software and not a paper

```
<resourceType resourceTypeGeneral="Software"/>
```

# ADS needs to index curated metadata about your work.

They can only work with the metadata they are given.

When we enrich metadata new connections are possible.

# Who does the work?

Libraries and archives aren't the direct
stewards of your work anymore.

We need to be able to find your work though.
You need to be able to make informed choices about it.
Our bibliographies represent your work.
We need to work together.

# software authors

- You control your metadata.
- You are your own cataloger.

We can give you tools but you need to make choices.

# You need to know when you're making choices that will impact your legacy.

## Acknowledging or Citing Astropy

### In Publications

If you use Astropy for work/research presented in a publication (whether directly, or as a dependency to another package), we ask that you please cite the Astropy papers:

- Astropy Paper II (ADS - BibTeX)
- Astropy Paper I (ADS - BibTeX)

Copy BibTeX to clipboard

We provide the following LaTeX/BibTeX acknowledgment if there is no specific place to cite the papers:

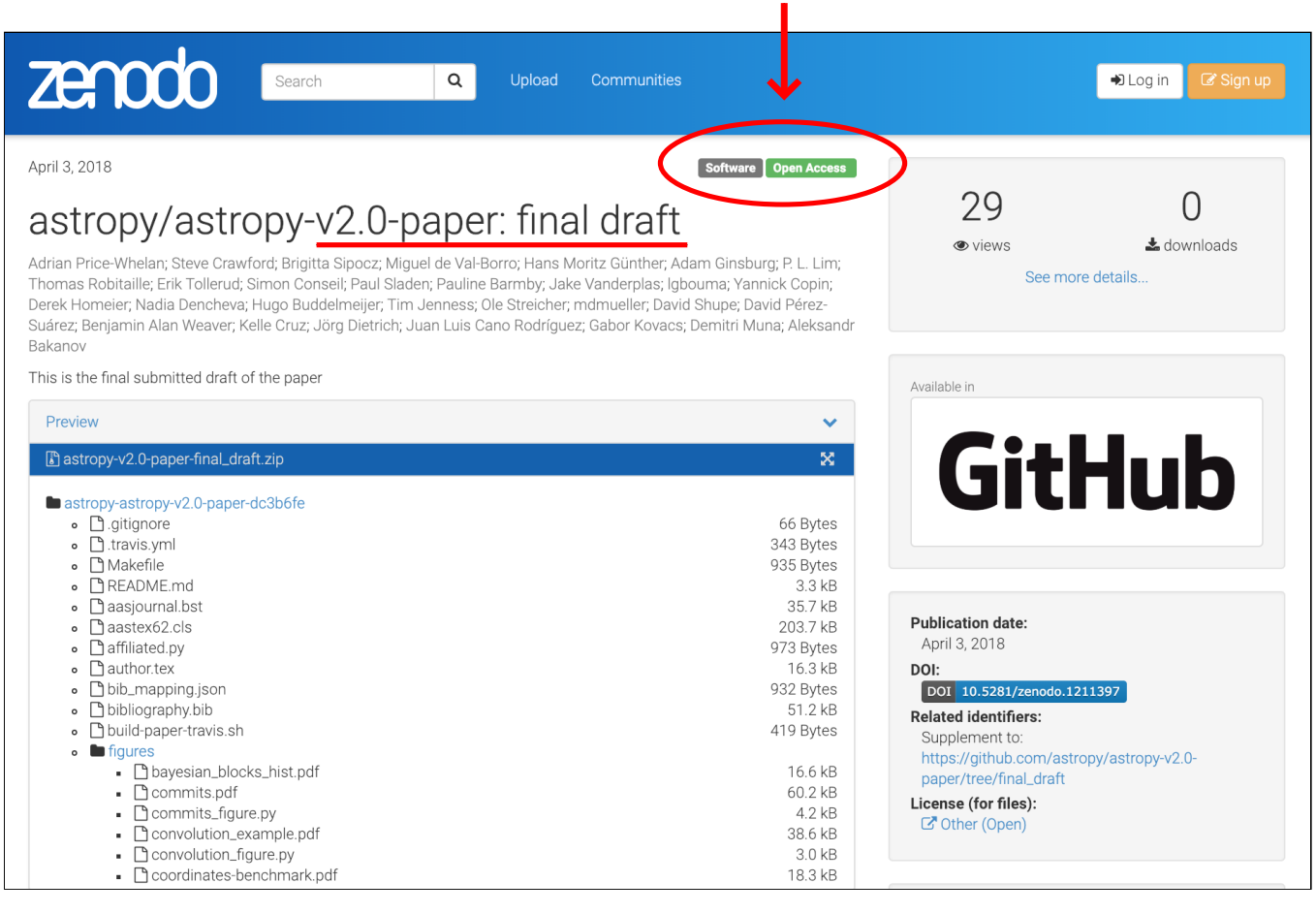
```
This research made use of Astropy,\footnote{http://www.astropy.org} a community-developed core Python
package for Astronomy \citep{astropy:2013, astropy:2018}.
```

Two different papers.

(Not the code)

# Software DOIs don't guarantee software citation

```
<resourceType resourceTypeGeneral="Software"/>
```



# complicated / conflicting author instructions

## ASCL Code Record

[ascl:1109.015] WCSTools: Image Astrometry Toolkit

Mink, Jessica

WCSTools is a package of programs and a library of utility subroutines for setting and using the world coordinate systems (WCS) in the headers of the most common astronomical image formats, FITS and IRAF .imh, to relate image pixels to sky coordinates. In addition to dealing with image WCS information, WCSTools has extensive catalog search, image header manipulation, and coordinate and time conversion tasks. This software is all written in very portable C, so it should compile and run on any computer with a C compiler.

Code site: http://tdc-www.harvard.edu/software/wcstools/

Appears in: http://adsabs.harvard.edu/abs/1999ASPC..172..498M

Bibcode: 2011ascl.soft09015M

Preferred citation method:
Depends on usage; see http://tdc-www.harvard.edu/software/wcstools/publications/ for information

# PlasmaPy: an open source community-developed Python package for plasma physics

PlasmaPy Community; Murphy, Nicholas A.; Leonard, Andrew J.; Stańczak, Dominik; Kozlowski, Pawel M.; Langendorf, Samuel J.; Haggerty, Colby C.; Beckers, Jasper P.; Mumford, Stuart J.; Parashar, Tulasi N.; Huang, Yi-Min

## BibTeX Export

```
@misc{plasmapy_community_2018_1238132,
  author       = {PlasmaPy Community and
                  Murphy, Nicholas A. and
                  Leonard, Andrew J. and
                  Stańczak, Dominik and
                  Kozlowski, Pawel M. and
                  Langendorf, Samuel J. and
                  Haggerty, Colby C. and
                  Beckers, Jasper P. and
                  Mumford, Stuart J. and
                  Parashar, Tulasi N. and
                  Huang, Yi-Min},
  title        = {{PlasmaPy: an open source community-developed
                   Python package for plasma physics}},
  month        = apr,
  year         = 2018,
  note         = {{This work was partially supported by the U.S.
                   Department of Energy.}},
  doi          = {10.5281/zenodo.1238132},
  url          = {https://doi.org/10.5281/zenodo.1238132}
}
```

Slide Deck

# You cannot assume archival repositories know what to ask you for.

Systems need to change.

People who write software
need to decide what matters.

But we have started to define our lexicon.

# Citation File Format

human- and machine-readable file format that provides citation metadata for software.

## Example

If you want to make your software easily citable, you can put a file called `CITATION.cff` in the root of your repository. This file should provide at least the minimally necessary metadata to cite your software. For example:

```yaml
cff-version: 1.0.3
message: If you use this software, please cite it as below.
authors:
  - family-names: Druskat
    given-names: Stephan
    orcid: https://orcid.org/0000-0003-4925-7248
title: My Research Tool
version: 1.0.4
doi: 10.5281/zenodo.1234
date-released: 2017-12-18
```

# CodeMeta

more than citation metadata

## Creating A CodeMeta Instance File

A CodeMeta instance file describes the metadata associated with a software object using JSON's linked data (JSON-LD) notation. A codemeta file can contain any of the properties described on the CodeMeta terms page.

When creating a CodeMeta document, note that they contain JSON name ("property" in linked-data), value pairs where the values can be simple values, arrays or JSON objects. A simple value is a number, string, or one the literal values *false*, *null true*, for example:

```
"name" : "R Interface to the DataONE REST API"
```

A JSON array is surrounded by the characters `[` and `]`, and can contain multiple values:

```
"keywords": [ "data sharing", "data repository", "DataONE" ]
```

Some properties, such as `author`, can refer to other JSON objects surrounded by curly braces and can contain other JSON values or objects, for example:

```
"author": {
   "@id":"http://orcid.org/0000-0003-0077-4738",
   "@type":"Person",
   "email":"slaughter@nceas.ucsb.edu",
   "givenName":"Peter",
   "familyName: "Slaughter"
}
```

The JSON-LD "@type" keyword associates a JSON value or object with a well known type, for example, the statement `"@type":"Person"` associates the `author` object with `http://schema.org/Person`.
It is good practice to always provide the `@type` for any property which specifies a node (JSON object). The terms page indicates these node types.

The "author" JSON object illustrates the use of the JSON-LD keyword "@id", which is used to associate an IRI with the JSON object. Any such node object can be assigned an `@id`, and we may use the `@id` to refer to this same object (the person, Peter), elsewhere in the document; e.g. we can indicate the same individual is also the `maintainer` by adding:

```
"maintainer": "http://orcid.org/0000-0003-0077-4738"
```

Crosswalk for WikiData Properties

Crosswalk for DOAP Ontology

Crosswalk for DataCite metadata

Crosswalk for Debian packages

Crosswalk for GitHub API

Crosswalk for Java's Maven metadata

Crosswalk for NodeJS package.json

Crosswalk for Python distutils

Crosswalk for R Packages

Crosswalk for Ruby gems

# CodeMeta uses JSON-LD (JSON linked data)

Lets us translate our lexicon from one schema to another.

Enables interoperability and further contextualization.

Identifiers can be mapped to other identifiers.

# We're working on defining new metadata architectures

e.g. SigMF (Signal Metadata Format)

**Recording**

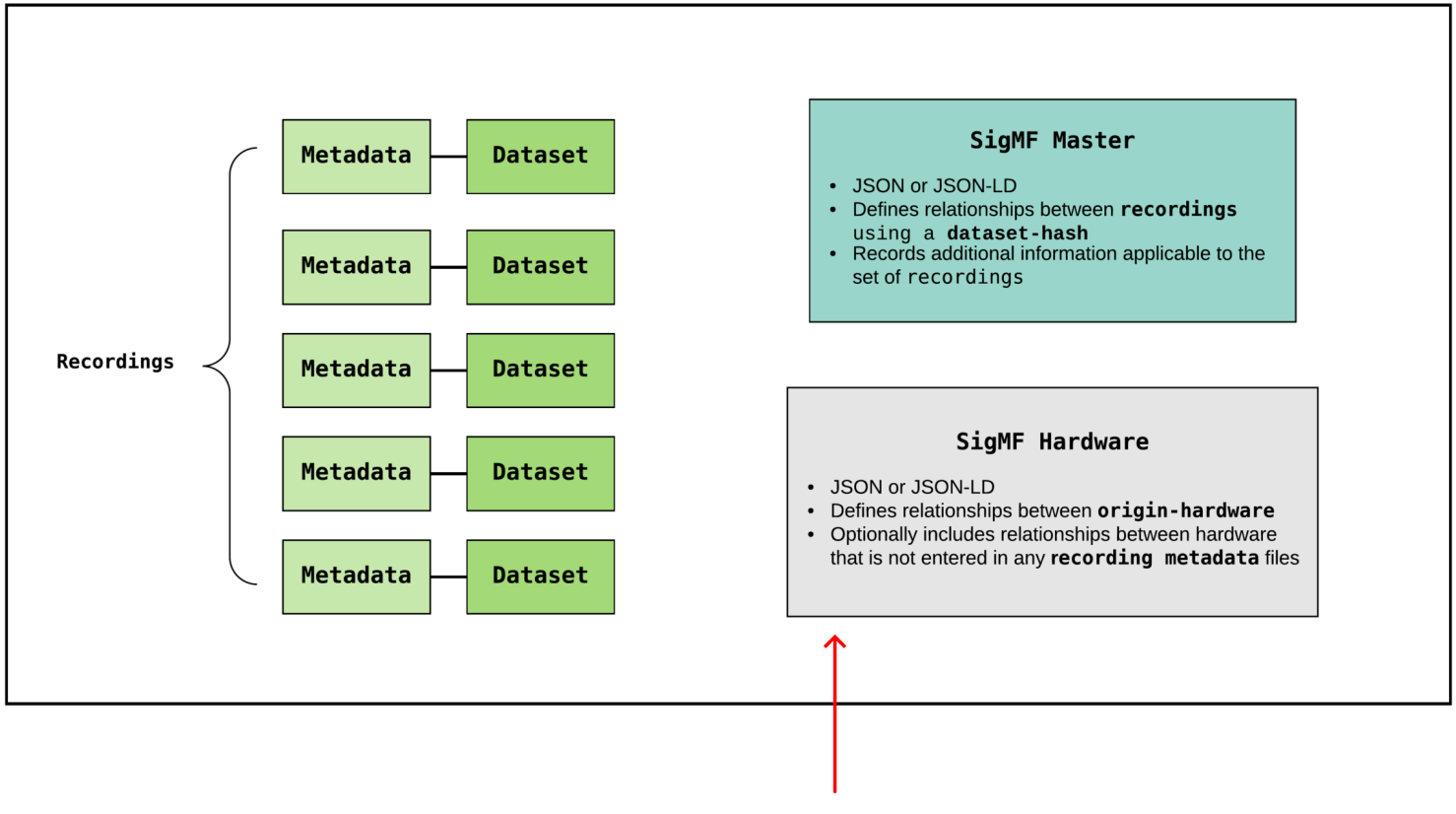
**Metadata**
(.sigmf-meta)

- JSON-LD
  - Allows you to reference external metadata schemas for *context*
  - Structure - "namespace:name": value
- Object types:
  - **global**
  - **captures**
  - **annotations**
- Objects have namespaces:
  - `core:<name>`
    - Includes
      - `core:`**dataset-hash**
      - `core:`**origin-hardware**
      - `core:`**extensions**

**Dataset**
(.sigmf-data)

- contains **Samples** of digital measurements
- referenced by a **dataset-hash**

**SigMF Archive** (.tar.sigmf) OR **SigMF Directory**




Hardware is provenance

Working on Guidance

(building discipline specific resources too)

## SSI/Jisc Guidance for Software Deposit

Jackson, M. (2018b). Software Deposit: What to deposit (Version 1.0). http://doi.org/10.5281/zenodo.1327325

## Example: Jupyter Notebooks

Bouquin, D., Hou, S., Benzing, M., Wilson, L. (2019). Jupyter Notebooks: A Primer for Curators (Version v1.0).

http://doi.org/10.5281/zenodo.2591580

Things you can do

**right now**

# Software Authors

- Mint a software DOI
  - deposit a release of your software and metadata files (Zenodo, Figshare, an institutional repository, etc.)
- Create a CFF file (minimal metadata for identification)
  - Consider making a CodeMeta file (more context)
- License your data and code explicitly
- **Update and check your metadata**
  - Check it again
- Link documentation to the source code directly
- Ensure preferred citations/any instructions about attribution **enable direct access to the software itself using your DOI**
- If you have many versions of software, decide who the authors are for each version (also get ORCiDs).

# article authors

- Cite archived software directly.
- No one else will catch mistakes.
- You are your own copy editor.

# Article Authors

- **Unambiguous, direct software citation**
  - If the preferred citation is not to the software, cite the software **and** the other thing.
  - Always **cite the archival copy** of the code when it exists
    - You might need to look for it.
- Consider the **version** that you are citing.
  - Who are you trying to give credit?
- **Put software citations in the references section**
- **Cite your own code in a software paper**
  - tells others how you want it cited

# And yet it moves.

We have a complete history of nothing.
Some things get a legacy and some things don't.
Your work matters.