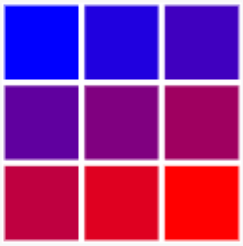


FAIRshake

FAIRshake User Guide V.2

CFDE CC MOD 16 Workplan 2021 Deliverable 5.c

Submitted: April 30, 2022

Authors: Daniel J.B. Clarke, MS; Eryk Kropiwnicki, MS; Avi Ma'ayan, PhD

Table of Contents

I. What is FAIRshake?

As more digital resources are produced by the research community, it is becoming increasingly important to harmonize and organize them for synergistic utilization and reuse. The findable, accessible, interoperable, and reusable (FAIR) guiding principles [1] have prompted many stakeholders to consider strategies for tackling this challenge. The FAIRshake toolkit [2] was developed to enable the establishment of community-driven FAIR metrics and rubrics paired with manual and automated FAIR assessments. FAIR assessments are visualized as an insignia that can be embedded within digital-resources-hosting websites. Using FAIRshake, a variety of biomedical digital resources can be manually and automatically evaluated for their level of FAIRness. The purpose of FAIRshake is not to penalize and judge digital object producers and servers, but to assist them with improving the interoperability of the products they produce and host. FAIRshake was also created to promote the use of community standards so an ecosystem of digital objects can better interoperate.

II. Getting Started

Starting a Project with FAIRshake

FAIRshake can be accessed from https://fairshake.cloud. On the site, the Projects tab lists existing FAIRshake projects. Each project in FAIRshake bundles a collection of registered digital objects that are associated with the project. Examples of such digital objects include software tools, datasets, databases, API, or workflows. Each of these digital objects is associated with one or more FAIR rubrics used to evaluate it. To start your own project in FAIRsahke, you need to first establish a user account (Fig. 1) and sign in (Fig. 2). FAIRshake support account set up and sign in with ORCID, GitHub, or Globus. Accounts in these environments are not required. The user can sign up with their own username and e-mail exclusively with FAIRshake.

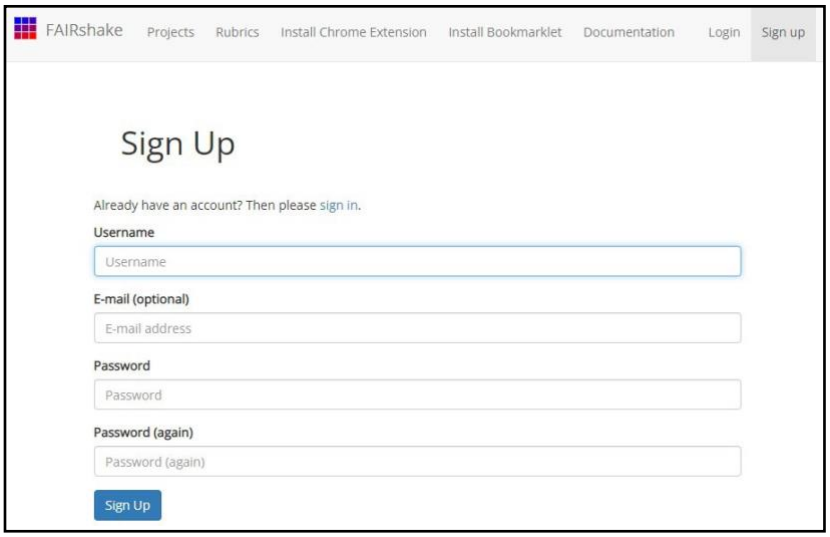


Fig. 1 FAIRshake sign up page

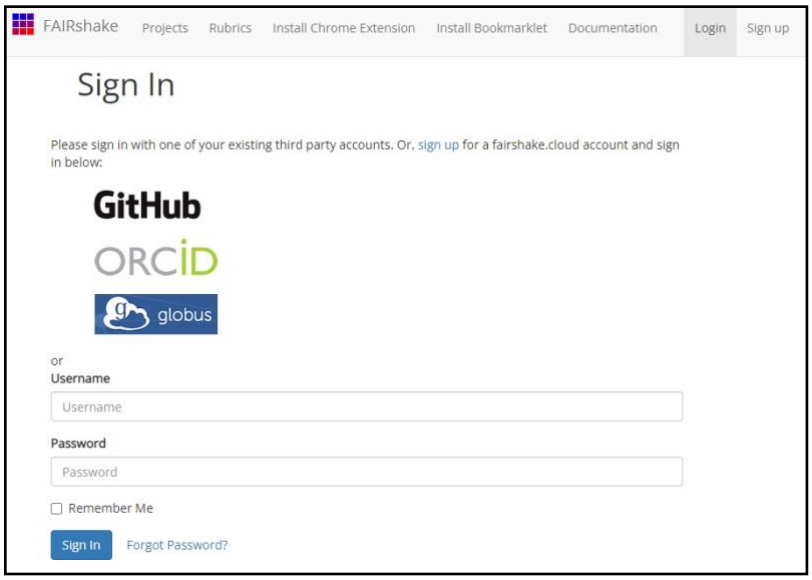


Fig. 2 FAIRshake sign in page

Next, on the Projects page, click the “Create New Project” card. This will invoke the presentation of an input form for submitting metadata about the project. Once you are done filling out the form, press submit to establish the project in the FAIRshake database (Fig. 3).

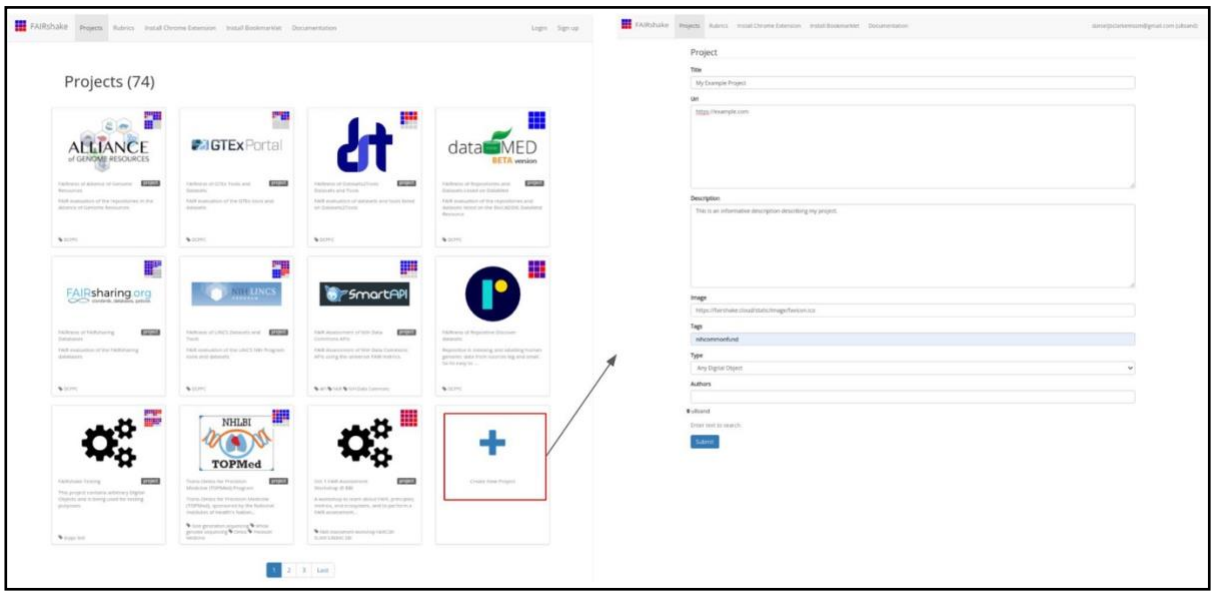


Fig. 3 Operations for creating a new project

Associating digital objects with a project in FAIRshake

Navigate to your project’s page. Under the “Associated Digital Objects” header, click the “Create New Digital Object” card. You will be presented with an input form for entering metadata about your digital object (Fig. 4).

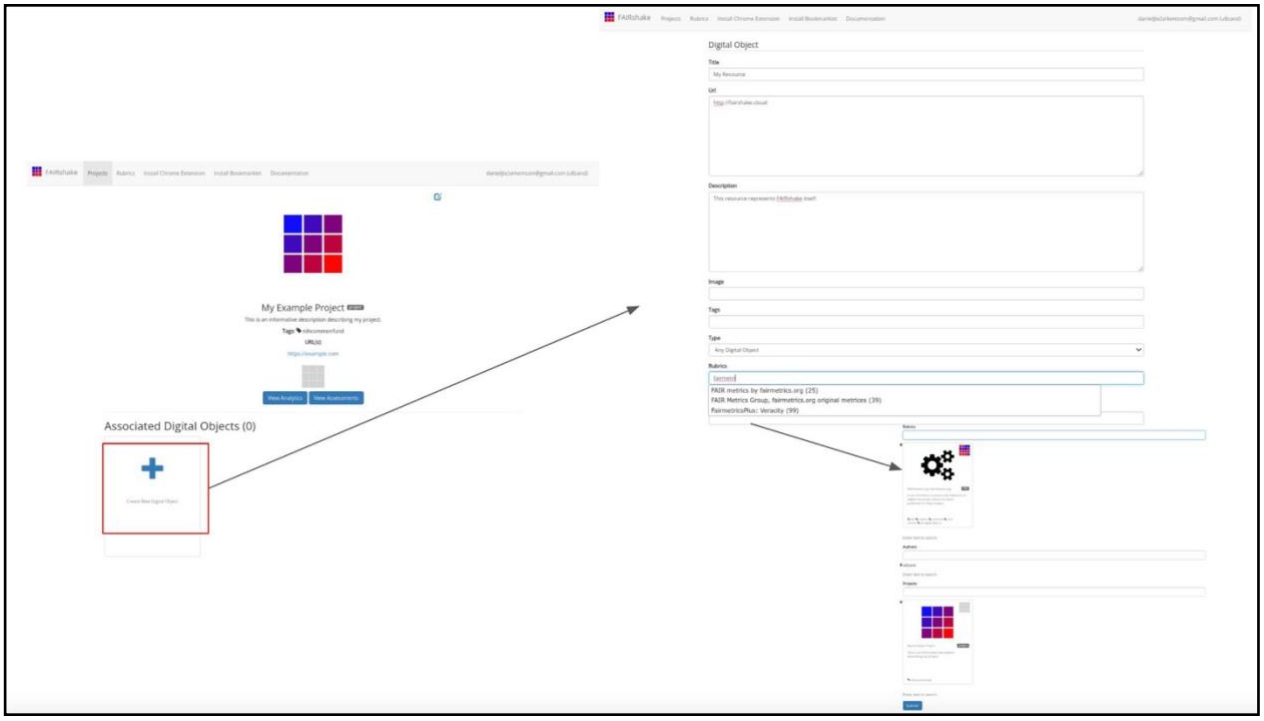


Fig 4. Associating a new digital object with a project

In the rubrics autocomplete field, enter text to search for available rubrics. You will be presented with a list of potentially relevant rubrics that you might want to be associated with your digital object. Once the form is submitted, you will be redirected to a page that is created specifically for your digital object with the associated projects and rubrics (Fig. 5).

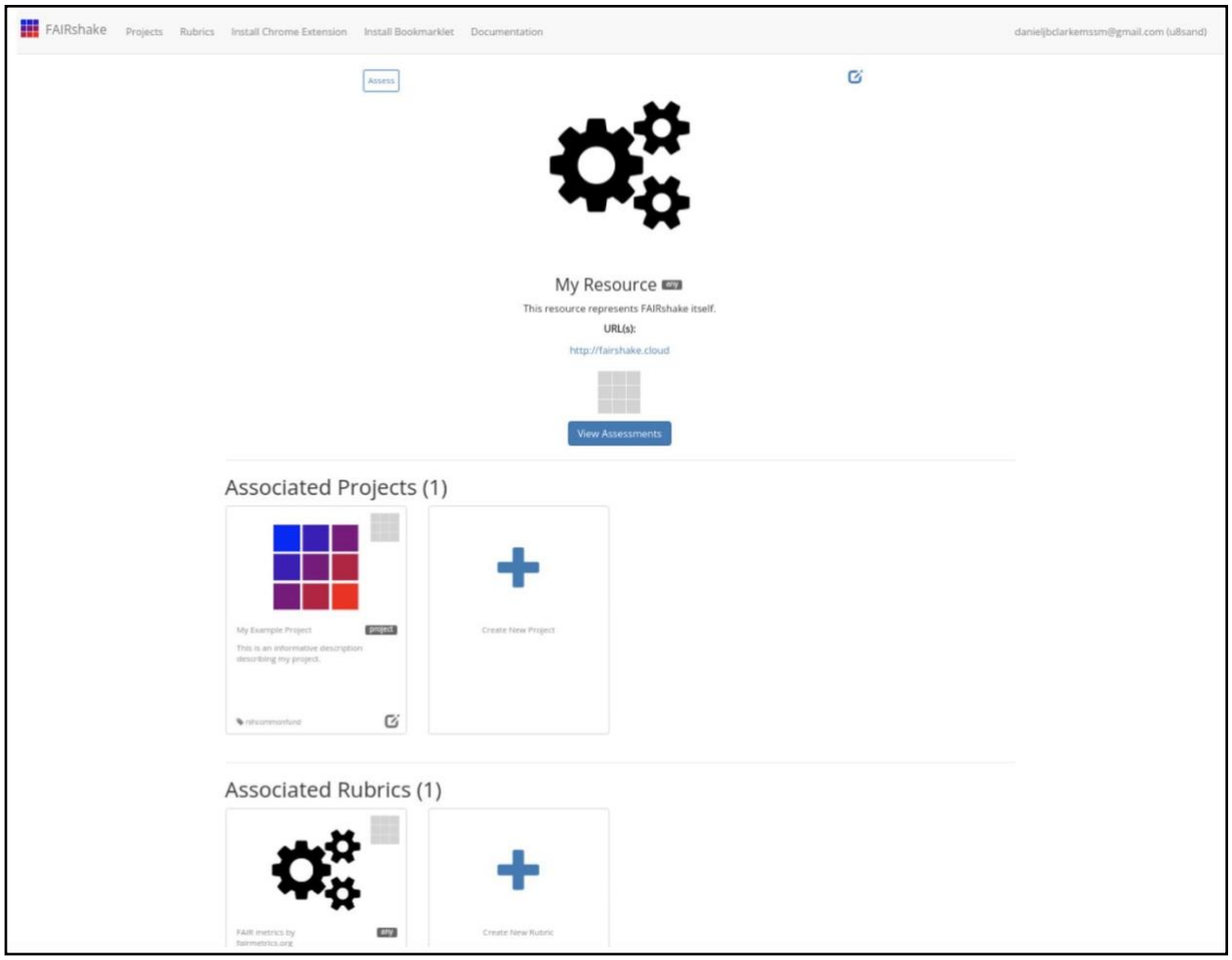


Fig 5. Digital object page

Associating a rubric with a digital object in FAIRshake

A rubric is a set of questions used to evaluate the FAIRness of a specific digital object in a particular project. To create a rubric, navigate to the “Rubrics” tab in the navigation bar. Click the “Create New Rubric” card to start. You will see an input form where you can enter information about your rubric (Fig. 6).

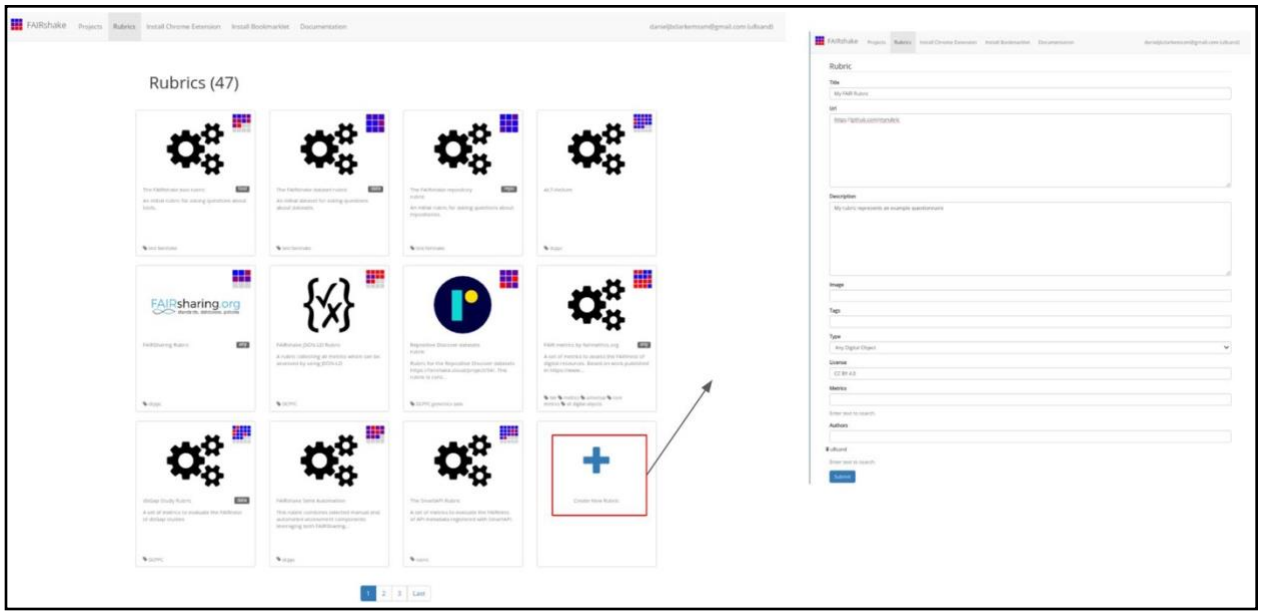


Fig 6. Input form options for creating a new rubric.

The “Metrics” field within the form is an autocomplete text search that provide options to select from existing metrics already inside the FAIRshake database. Once starting to type, previously defined metrics that may be added to the rubric are listed (Fig. 7).

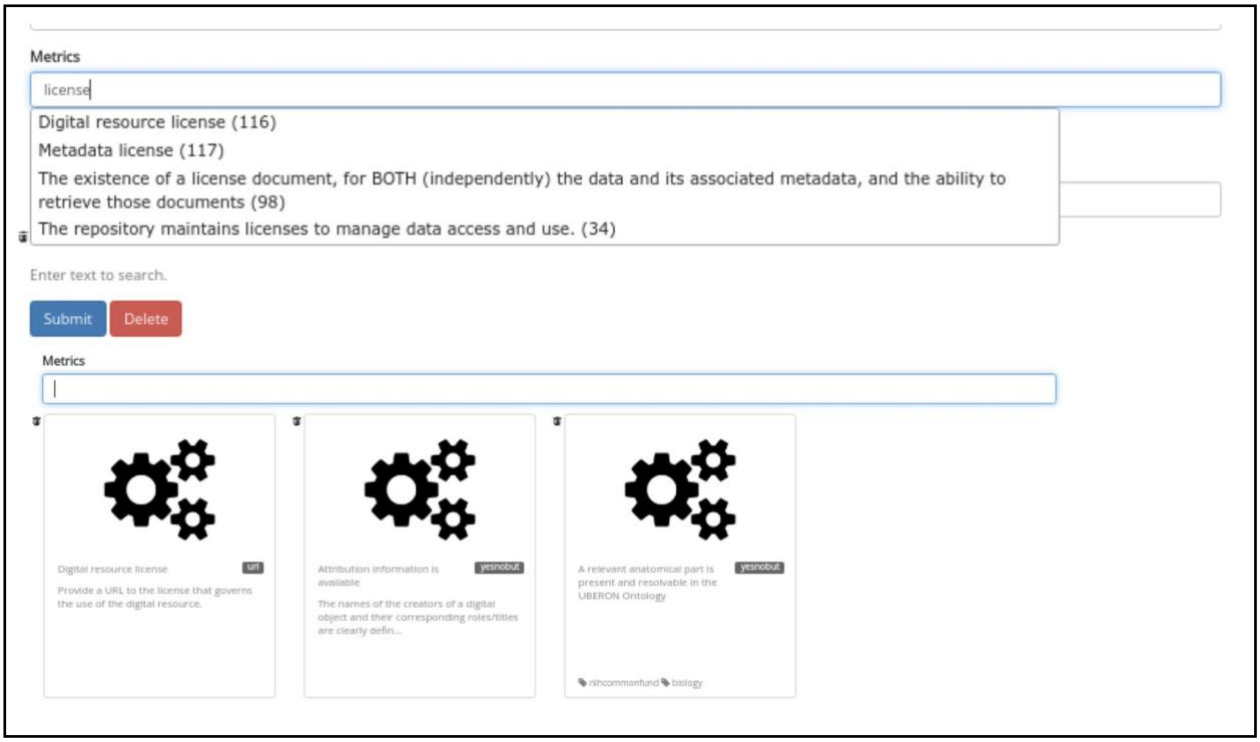


Fig 7. Autocomplete text field for available metrics.

Performing your first assessment with FAIRshake

Rubrics are developed to cover various aspects of FAIRness pertaining to a digital object in a specific domain of research. The metrics chosen for a specific rubric may represent some aspects of FAIR, but they do not need to cover all aspects. The key is to make digital objects FAIR enough to be useful in the targeted community.

To that end, and to get a better sense of the scope of the FAIR metrics that could be developed to better serve a specific community, the FAIR metrics developed by fairmetrics.org Rubric are a good starting point. This rubric is a FAIRshake entry for the universal FAIR metrics published in this paper, representing a universal set of broad criteria that should apply to most digital objects (Fig. 8).

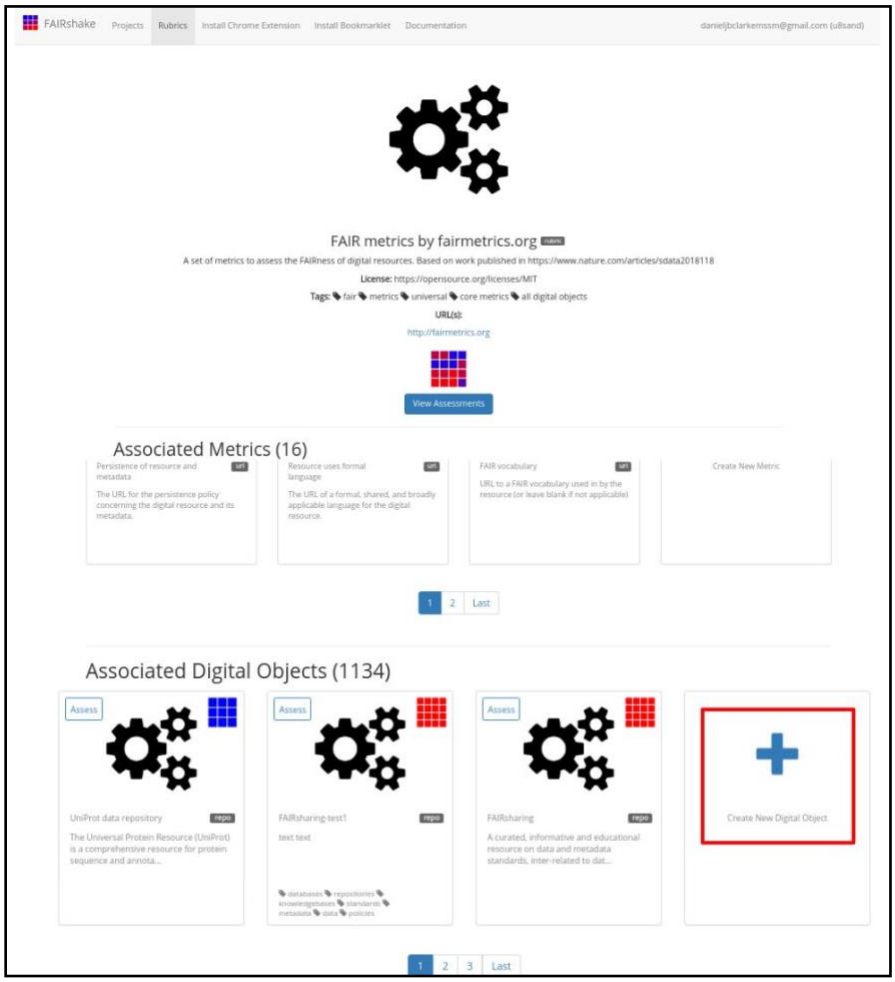


Fig. 8 FAIR metrics rubric of universal FAIR metrics.

Once you assign a rubric to all the digital objects from your project, you can try picking one of these digital objects and complete and publish a manual FAIRshake FAIR assessment of it. You will be able to delete it later.

Now let us run through a FAIR assessment example.

A FAIRshake user aims to perform a manual assessment of the LINCS Data Portal [3] using the FAIRmetrics rubric (Fig. 9).

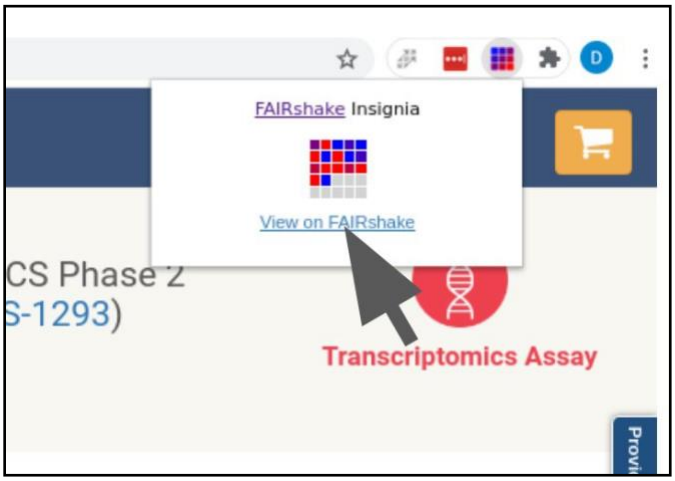


Fig. 9 The FAIRshake Google Chrome extension facilitates access to the FAIRshake website for assessments.

This brings her to FAIRshake to see the relevant information available on FAIRshake related to the page she is visiting (Fig. 10).

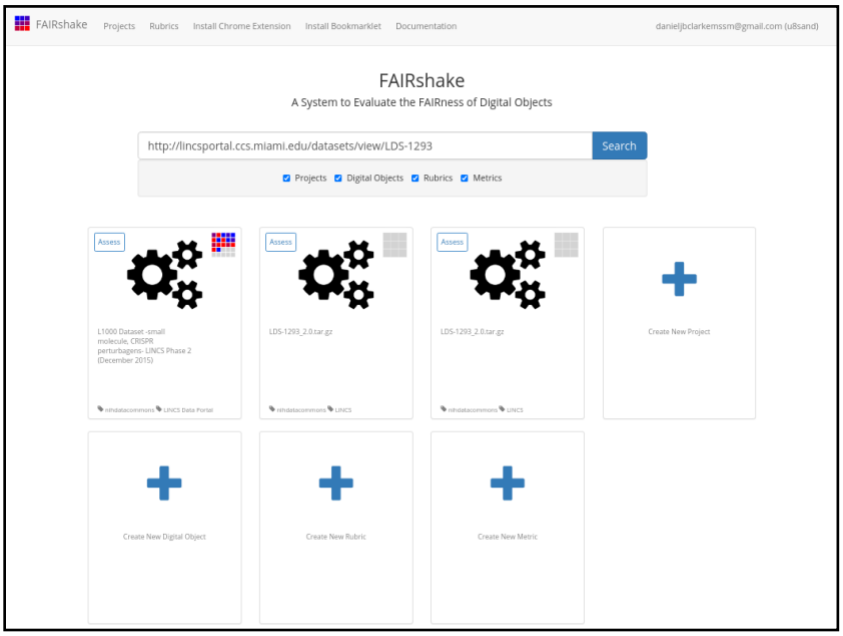


Fig. 10 Digital objects, projects, rubrics, and metrics related to the page from the LINCS Data Portal.

Alternatively, she could have found or registered this digital object directly on the FAIRshake website with the 'Create New Digital Object' button.

Clicking the assess button, she ends up at the assessment preparation page (Fig. 11).

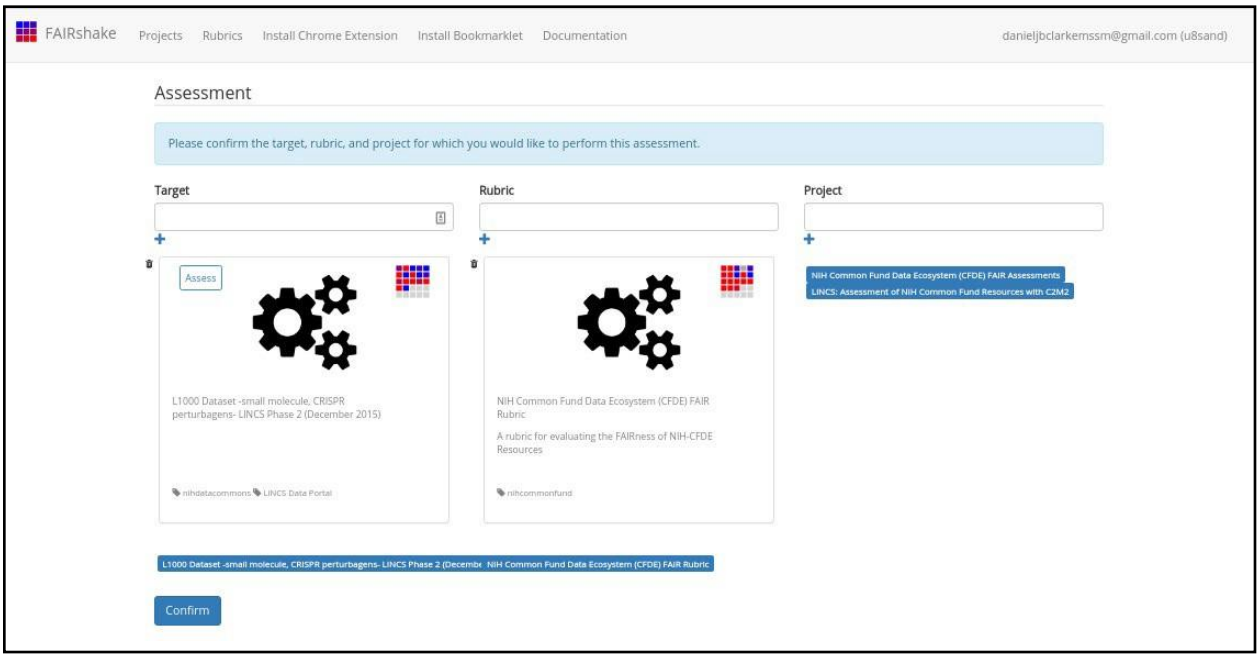


Fig. 11 FAIR Assessment Preparation Page.

The digital object and its only rubric were selected automatically, but the user ends up instead selecting the fairmetrics rubric (Fig. 12).

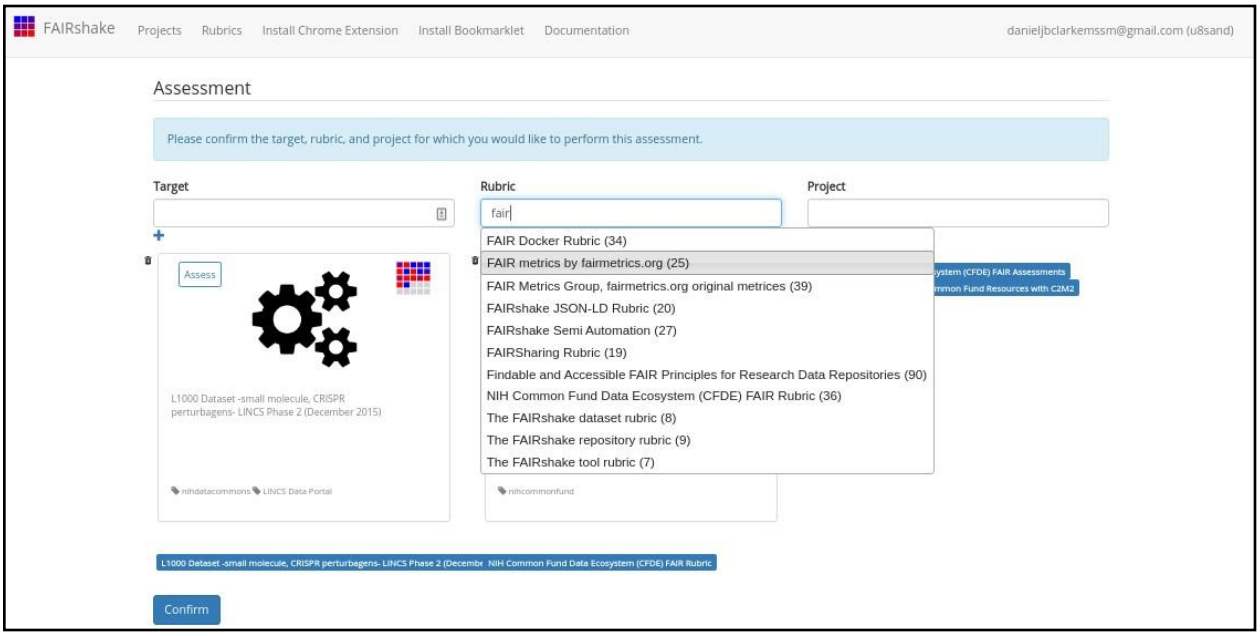


Fig. 12 Selecting the FAIR metrics rubric for the FAIR assessment.

The FAIRshake user performs this assessment as part of the FAIRshake testing project (Fig. 13). Alternatively, the FAIRshake user can create her own project. This is recommended if she expects to do a bunch of related assessments.

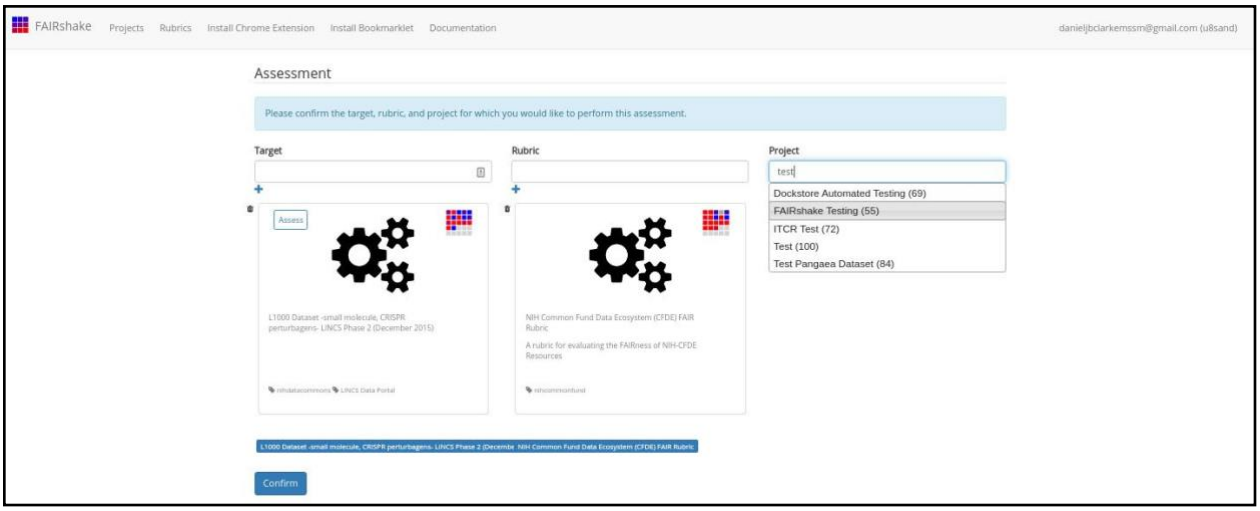


Fig. 13 Selecting the FAIRshake Testing project for the FAIR assessment.

It's also important to note that “project” here can be left blank if the assessment is not a part of any specific project.

Confirming this, the FAIRshake user begins a manual assessment (Fig. 14).

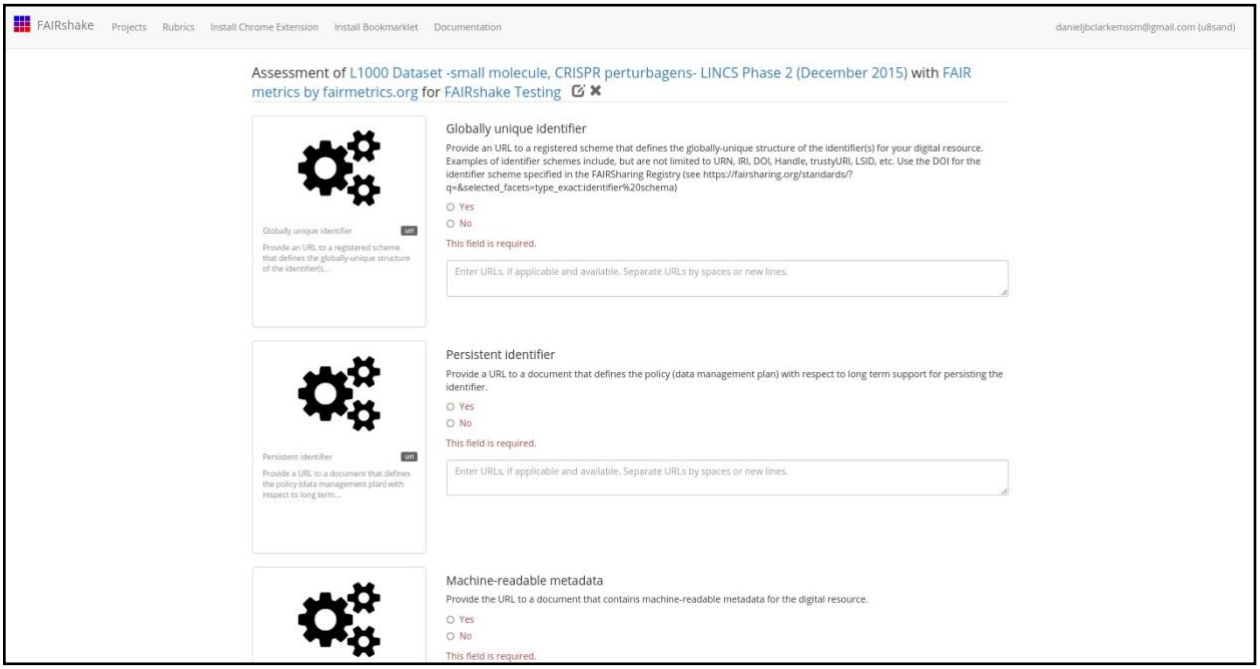


Fig. 14 Manual assessment page.

Each metric represents a concept pertinent to FAIRness which is described shortly before each prompt but potentially in more depth on the metrics' landing pages. Clicking on the metric "card" to the left of the question provides more information in a new tab (Fig. 15).

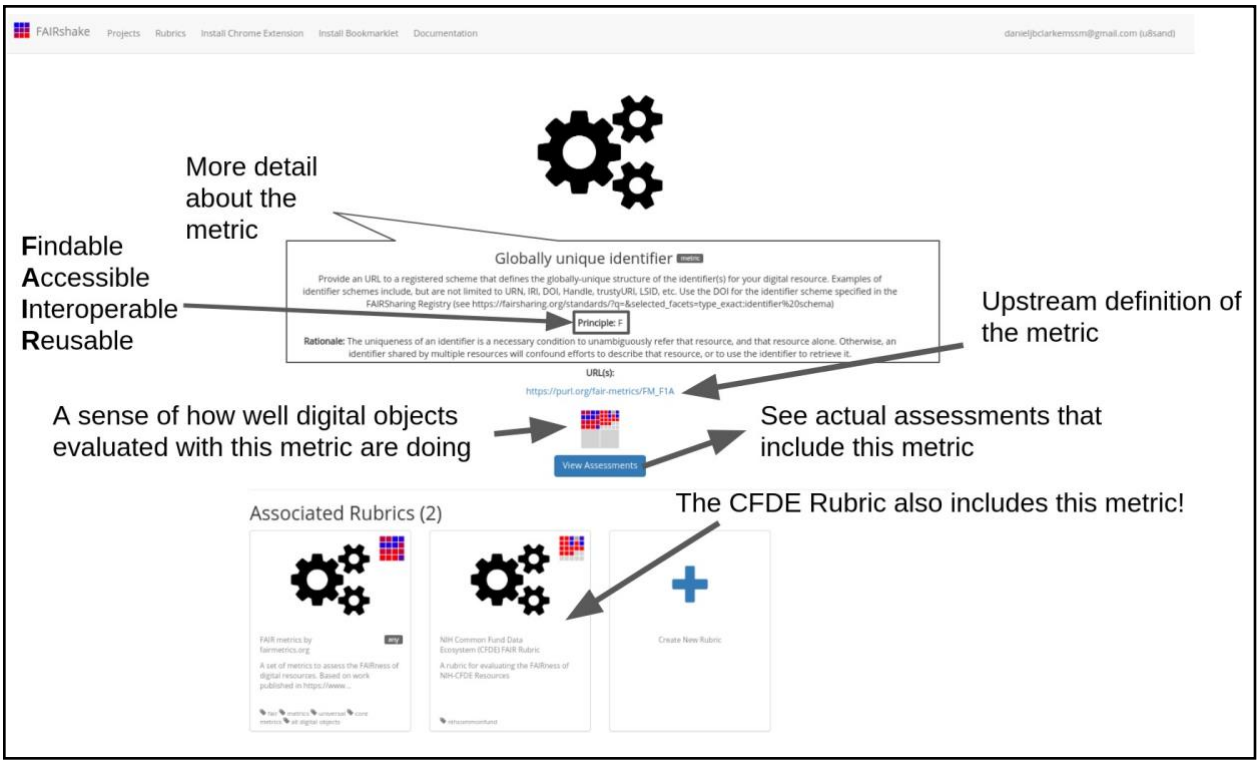
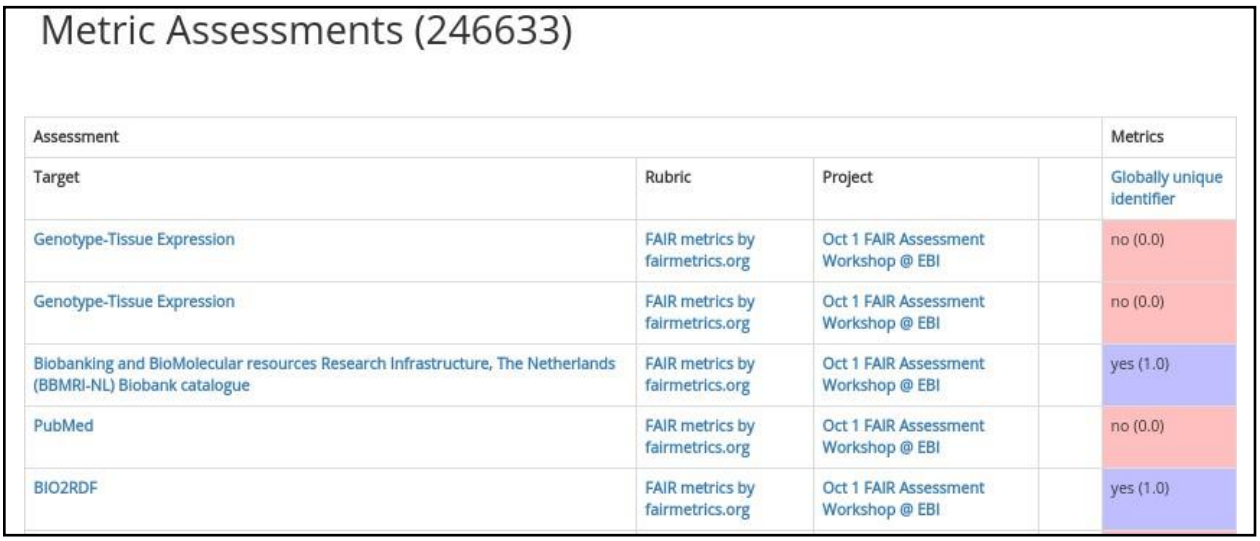


Fig. 15 More detailed information about the "globally unique identifier" metric.

Clicking 'View assessments' the user can see what other digital objects in the database received as an answer during an assessment through a tabular view (Fig. 16).



Metric Assessments (246633)

Assessment				Metrics
Target	Rubric	Project		Globally unique identifier
Genotype-Tissue Expression	FAIR metrics by fairmetrics.org	Oct 1 FAIR Assessment Workshop @ EBI		no (0.0)
Genotype-Tissue Expression	FAIR metrics by fairmetrics.org	Oct 1 FAIR Assessment Workshop @ EBI		no (0.0)
Biobanking and BioMolecular resources Research Infrastructure, The Netherlands (BBMRI-NL) Biobank catalogue	FAIR metrics by fairmetrics.org	Oct 1 FAIR Assessment Workshop @ EBI		yes (1.0)
PubMed	FAIR metrics by fairmetrics.org	Oct 1 FAIR Assessment Workshop @ EBI		no (0.0)
BIO2RDF	FAIR metrics by fairmetrics.org	Oct 1 FAIR Assessment Workshop @ EBI		yes (1.0)

Fig. 16 Tabular view of other digital object assessments for the "globally unique identifier" metric in the FAIRshake database.

Clicking on any of these links enables exploring the projects, rubrics, or digital objects that were assessed. This feature provides a more elaborate sense of why a particular score was received and in what context. We can see, for example, that the top entries refer to assessments made during an EBI workshop (Fig. 16).

Getting back to the assessment, the user can now determine whether the digital object satisfies the relevant criterion. This feature provides a standard that defines the globally unique structure of the identifier used for the resource.

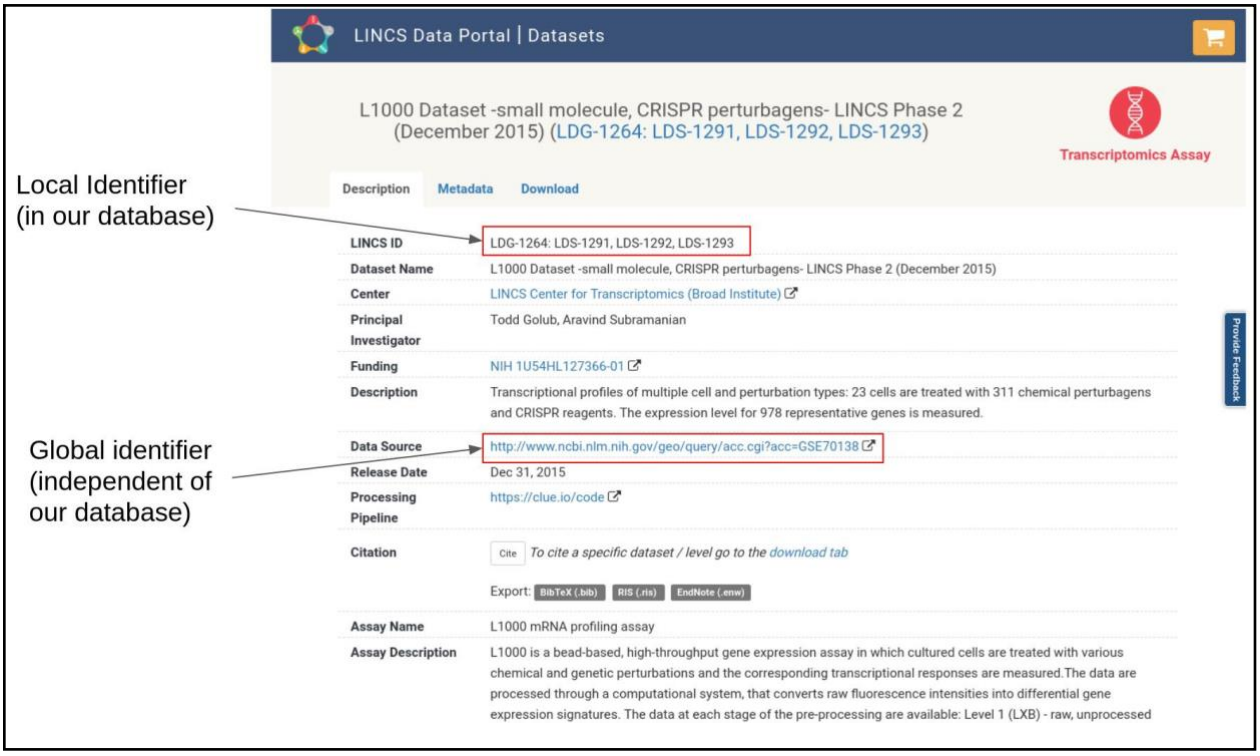


Fig. 17 LINCS Data Portal Page with various identifiers highlighted.

The user finds out quite quickly that there two identifiers for the same digital object:
(1) the data source global identifier at NCBI GEO GSE70138:
http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE70138
(3) the local LINCS Data Portal identifier: LDS-1293
http://lincsportal.ccs.miami.edu/datasets/view/LDS-1293

While these are legitimate identifiers, not all of them are used outside of the resource and may not be considered "globally-unique". The *scheme* however is shared because the URL appears in the FAIRsharing database along with a DOI and other standardized identifier schemes.

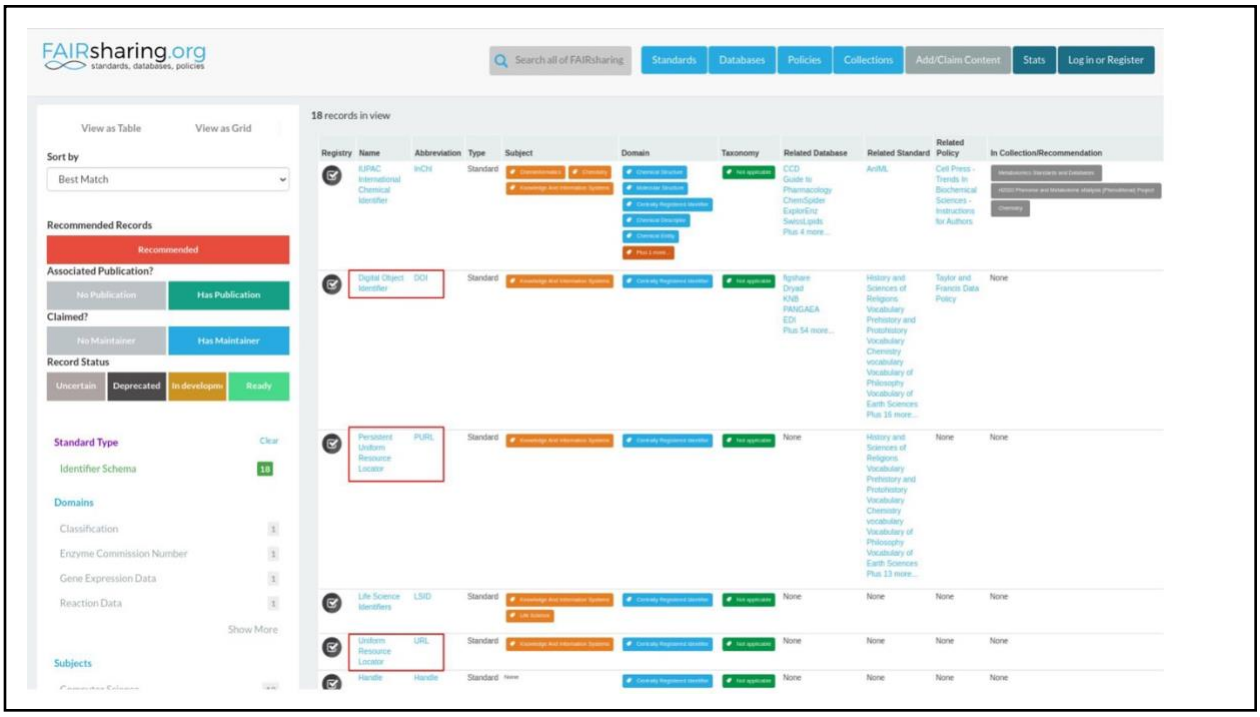


Fig. 18 Identifier schemes from FAIRsharing.org

A **URL** provides another level of standardization for identifying digital objects. However, most other identifier schemes may carry with them more information. For example, a **DOI** adds additional semantic interoperability conditions not available with URLs. Thus, in certain circumstances, a URL might be good enough as an identifier, but in other cases, a more specific standardized identifier might be more pertinent. For example, a DOI guarantees authorship information associated with the digital object, while a URL does not. Furthermore, many organizations have come together to guarantee that DOIs will not change, while URLs can be changed or removed by the owner of the resource.

Thus, the metric *is* satisfied in a broad context, though if the question was more specific, for instance -- "is there a DOI available for this digital object?" The answer might have been different. Hopefully, this example helps illuminating the need for establishing specific metrics relevant to each community. The more quantitative a metric is, the more stable and useful it will be when measured.

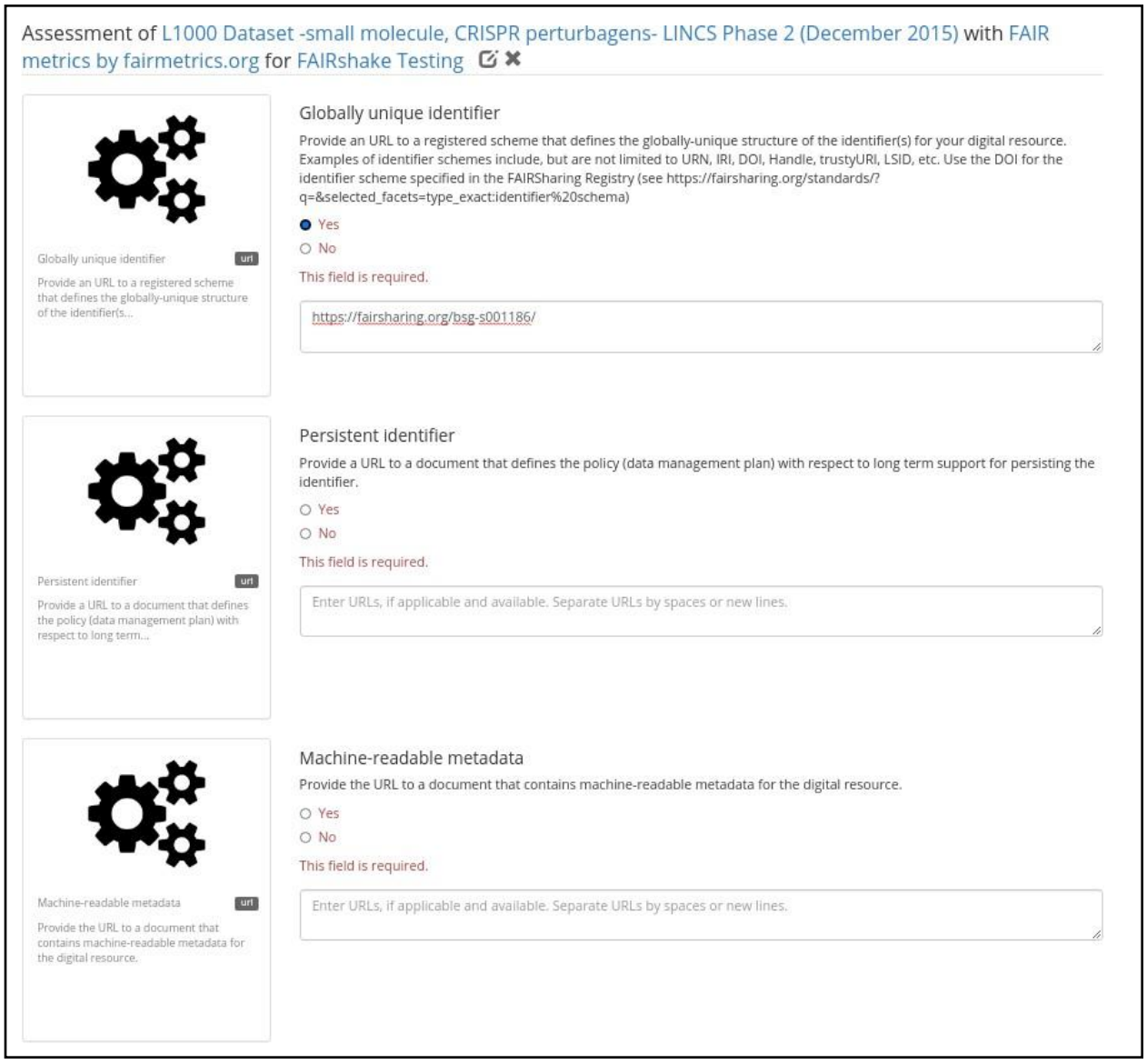


Fig. 19 FAIR assessment form with "Globally unique identifier" metric completed.

The next metric, persistent identifier, addresses persistence specifically and asks for a document describing the persistent identifier strategy. There is no obvious identifier type that guarantees this. After investigating this example, some information about the citation of the dataset can be found in the terms of use page of the LINCS Data Portal (Fig. 20):

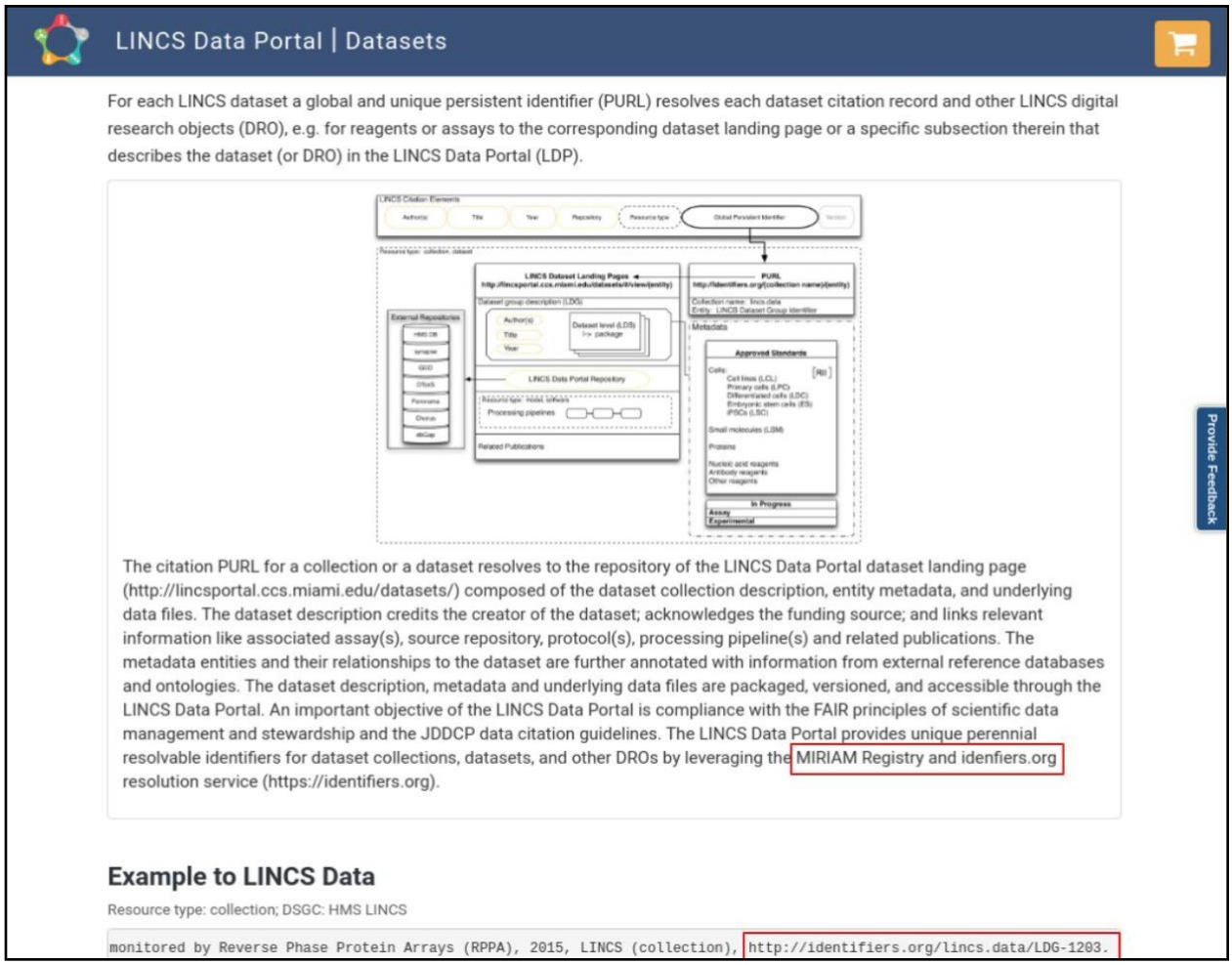


Fig. 20 Terms of use page at LINCS Data Portal with sections highlighting evidence about persistence of identifiers.

This reveals that the local identifiers are registered in identifiers.org, which is also recognized as a standard in FAIRsharing. In fact, few more identifiers can be created with this new information:

- lincs.data:LDS-1293
- http://identifiers.org/lincs.data/LDS-1293

Even if the LINCS consortium decides to change the URL structure of its data portal webpages, there is an expectation that these identifiers will be persistent and *not* change in structure. According to the terms, these are meant to be "global and unique persistent identifiers." These identifiers could likely satisfy the persistent identifier criterion citing the scheme as it is registered in identifiers.org. However, the existence of such a resolver service is not immediately obvious and available from the LINCS Data Portal landing page.

This demonstrates a scenario where even though LINCS may have persistent identifiers somewhere, they might not be discovered during the FAIR assessment. Whether we found the answer or not, we can learn something that can be improved. This is the sole purpose of performing FAIRshake FAIR assessments.

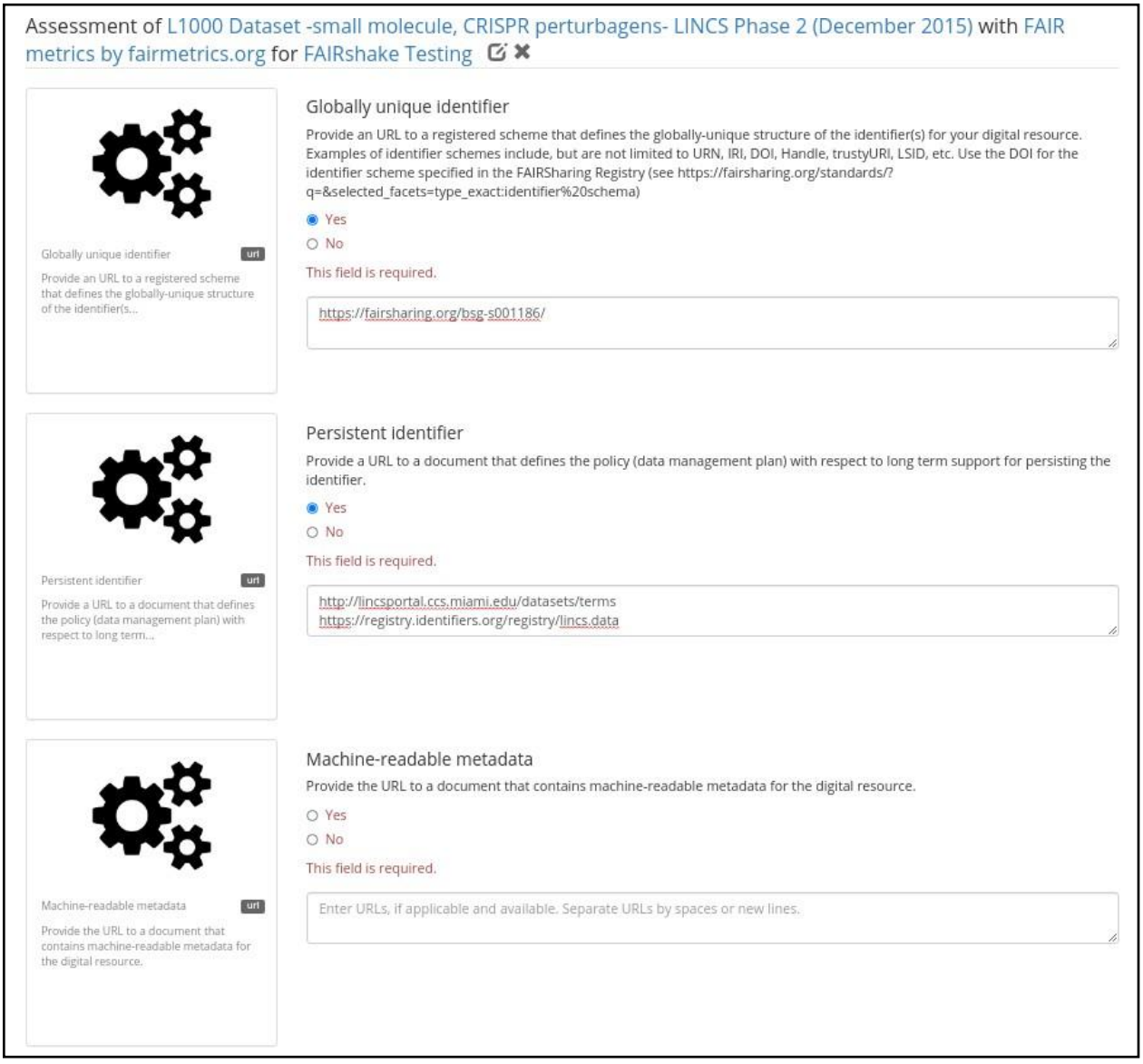


Fig. 21 FAIR assessment form with "Persistent identifier" metric completed.

Lastly, we look at machine readable metadata before discussing automated assessments.

FAIR strives to make things more Findable, Accessible, Interoperable, and Reusable, not just for humans but also for machines. With the massive amounts of data available in the public domain, many researchers conduct research by automatically locating data and operating with it without ever directly picking and choosing datasets, or analysis tools. To this end it is important that the FAIR principles are also considered from a machine perspective. For example, if a dataset that is hosted on a data

portal describes the assay used to generate the datasets as a paragraph of free text, it might be useful for a human visitor, but a software bot that visits the site will have difficulty with automatically identifying the assay-type aspect about the dataset. In this vein, machine readable metadata should ideally be available and documented. Again, it is not quite clear from the landing page of the dataset, or even from browsing the entire LINCS Data Portal site, that there *is* a public API documentation documented and registered in SmartAPI, another community resource also recognized by FAIRsharing.

Such API provides a structured way of accessing the information on the website making dataset selection and filterability more viable but nonetheless still not trivial. As such, we could say that we have machine-readable metadata, but it does not express the fully needed picture about the dataset (Fig. 22).

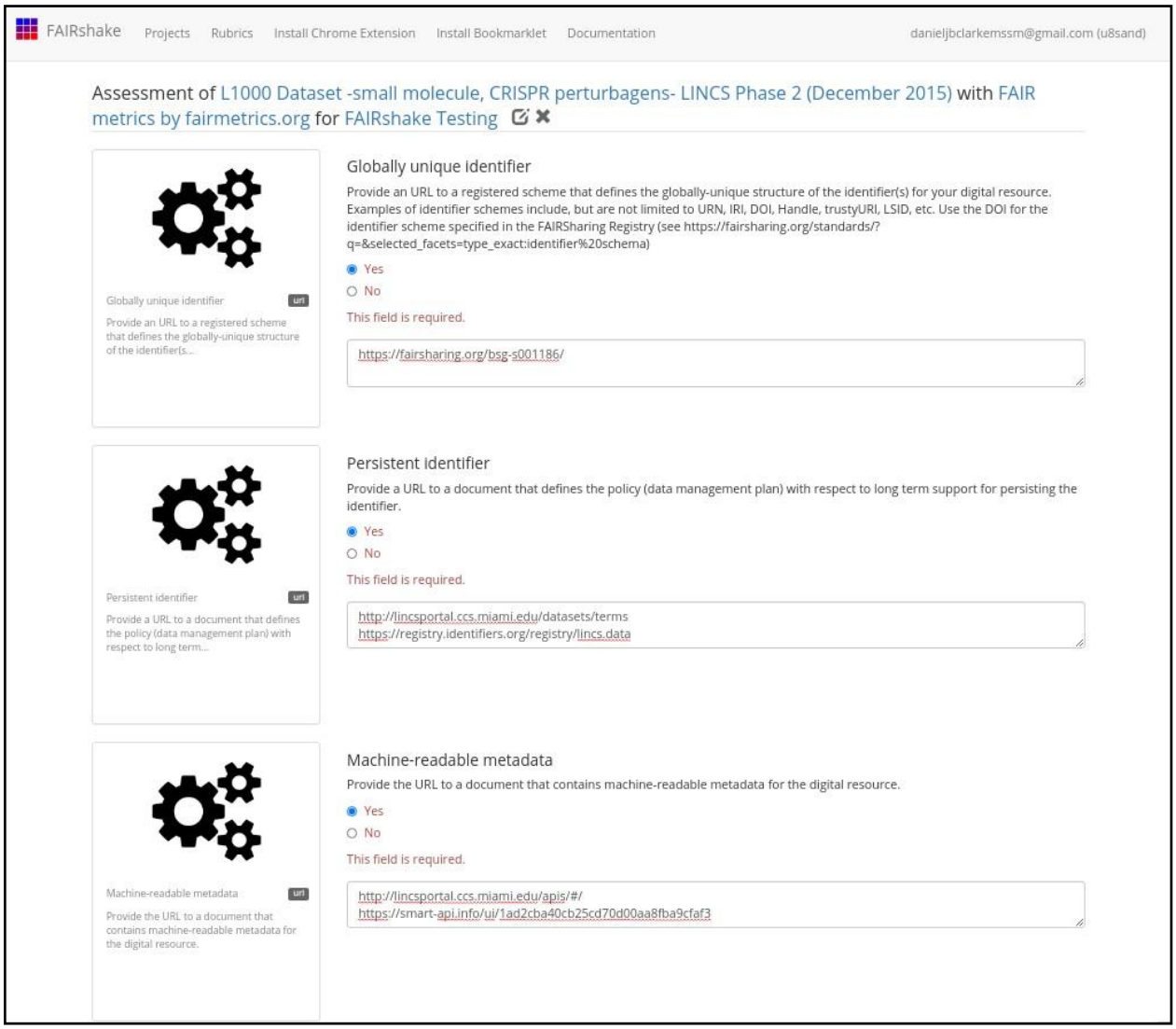


Fig. 22 FAIR assessment form with "Machine-readable metadata" metric completed.

Clearly, the FAIR metrics are just things to think about when it comes to making your digital objects FAIR. However, we likely need stricter and more concrete criterion if we are to measure FAIRness with precision. Furthermore, finding this information manually is time-consuming and would be intractable with large collections of digital objects.

This is where automated assessments and quantifiable metrics come in to help. Automated assessments and quantifiable metrics measure the moving target that is FAIRness. It is important to recognize at this point that a "good" or "bad" score produced by manual assessment with FAIRshake does little more than prompt discussions about things that can be improved towards FAIRness.

When we are done with our assessment, we can save, publish, or delete it. Once the assessment is published, the assessment cannot be modified. Only one assessment cn be applied on the same target digital object, with one rubric, in one project at one time. It is important to note that comments and URLs will only be accessible to the authors of the digital object, their assessment, or the project in which the digital object was assessed in if the assessment is not published (Fig. 23).

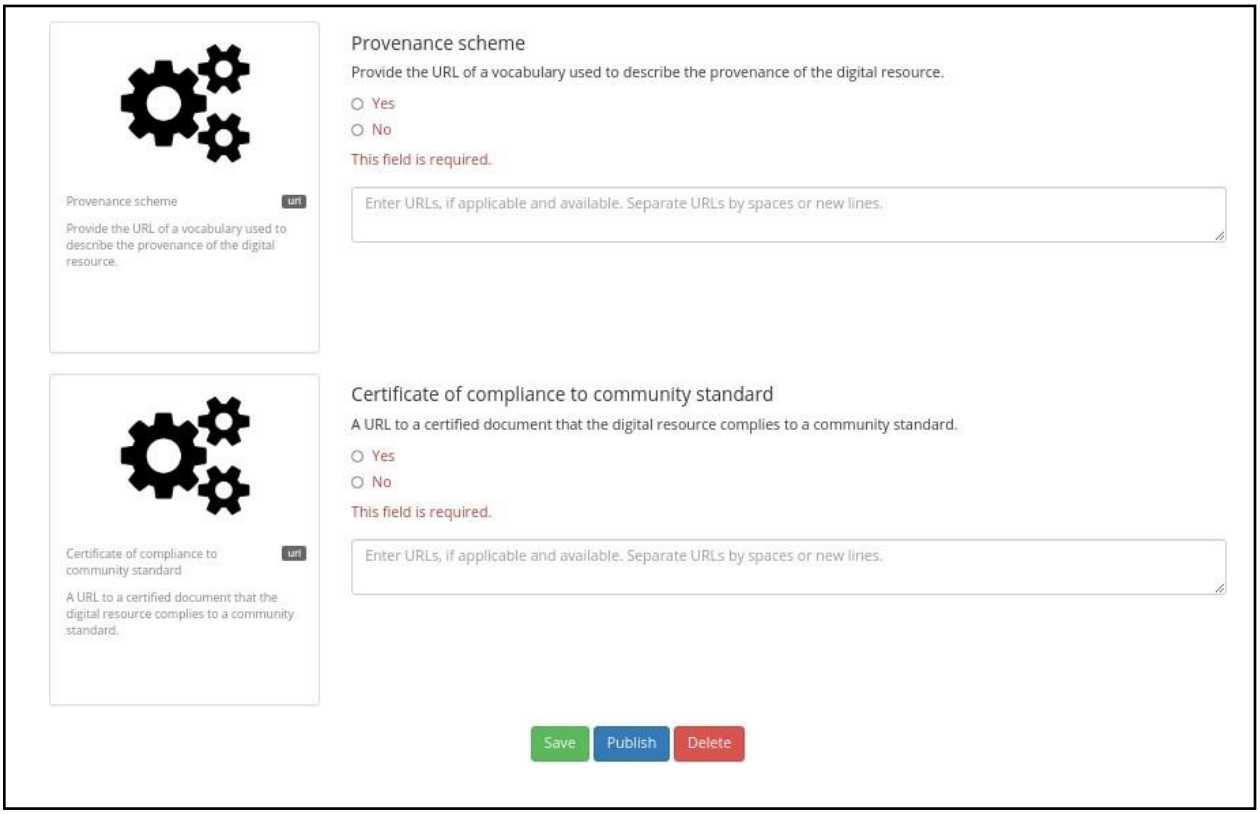


Fig. 23 Options to save, publish, or delete the FAIR assessment

If you complete and publish an assessment, your answers will become associated with the digital object that you assessed, and this information will be used for rendering the insignia and performing the analytics for that digital object (Fig. 24).

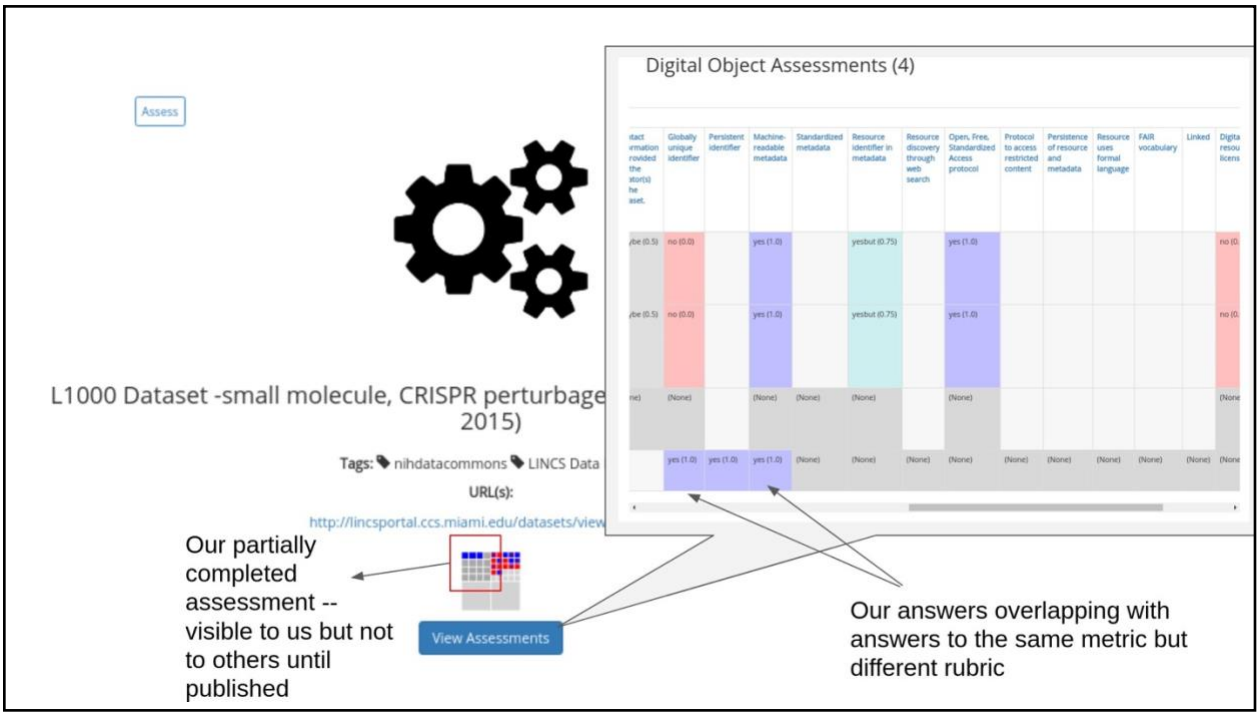


Fig. 24 Digital object assessment table denoting answers to the various metrics found across rubrics.

Though the assessments seem to agree that the digital object has machine readable metadata, it is unclear whether a globally unique identifier is present. Next, we will find out exactly why since those were reported by an automated assessment.

III. Metrics and Rubrics

What is a metric?

FAIR metrics are questions that assess whether a digital object complies with a specific aspect of FAIR. A FAIR metric is directly related to one of the FAIR guiding principles. FAIRshake adopts the concept of a FAIR metric from the FAIRmetrics effort [4]. To make FAIR metrics reusable, FAIRshake collects information about each metric and when users attempt to associate a digital resource with metrics and rubrics, existing metrics are provided as a first choice. FAIR metrics represent a human-described concept which may or may not be automated; automation of such concepts can be done independently by linking actual source code to reference the persistent identifier of that metric on FAIRshake. Without linked code, metrics are simply questions which can be answered manually. FAIRshake defines several categorical answer types to FAIR metrics when manually assessed which are ultimately quantified to a value in a range between zero and 1 $R \in 0,1$ or can take the property of undefined. Programmatically,

metric code can quantify the satisfaction of a given FAIR metric within the same continuous range. The FAIRshake toolkit provides a mechanism for contributing metric assertion code by means of RDF translation and inferencing.

What is a rubric?

The concept of a metric in FAIRshake is supplemented with that of a FAIRshake rubric. A FAIRshake rubric is a collection of FAIR metrics. An assessment of a digital resource is performed using a specific rubric by obtaining answers to all the metrics within the rubric. The use of a FAIR rubric makes it possible to establish a relevant and applicable group of metrics for many digital resources, typically under the umbrella of a specific project. Linking rubrics to digital resources by association helps users understand the context of the FAIR metrics which best fit the digital resources in their projects.

Creating a new metric

Navigate to an existing rubric page. Any rubric will contain associated metrics which are questions that assess aspects of FAIR that a digital object must comply with. Click the “Create New Metric” card to be redirected to an input form that can be populated with identifying metadata for the metric. Among the input form fields are options to change the type of question the assessor must answer, a rationale box for an explanation of why a particular choice was selected by the assessor, and a selection of which FAIR principle is being assessed.

Adding the new metric to an existing rubric

Navigate to an existing rubric page. In the top right corner of the page will be an icon that can be clicked to modify the existing rubric. Under the “Metrics” autocomplete field of the form, start typing the name of the metric that you want to be associated with the rubric. Add the metric to the rubric and click submit at the bottom of the page to save the changes to the rubric.

Creating a new rubric

To create a rubric, navigate to the “Rubrics” tab in the navigation bar. Click the “Create New Rubric” card to be presented with an input form for identifying metadata for your new rubric. The “Metrics” field within the form is an autocomplete text search that enables inputting previously defined metrics that can be added to the rubric.

IV. Visualizing and Evaluating FAIRshake Results

The FAIRshake Insignia

The FAIR insignia aggregates each metric separately to inform digital object producers where they can improve the FAIRness of the digital objects they produce and host. This is when metrics have a low percentage. Digital objects may be assessed by different rubrics which are made from different collections of metrics.

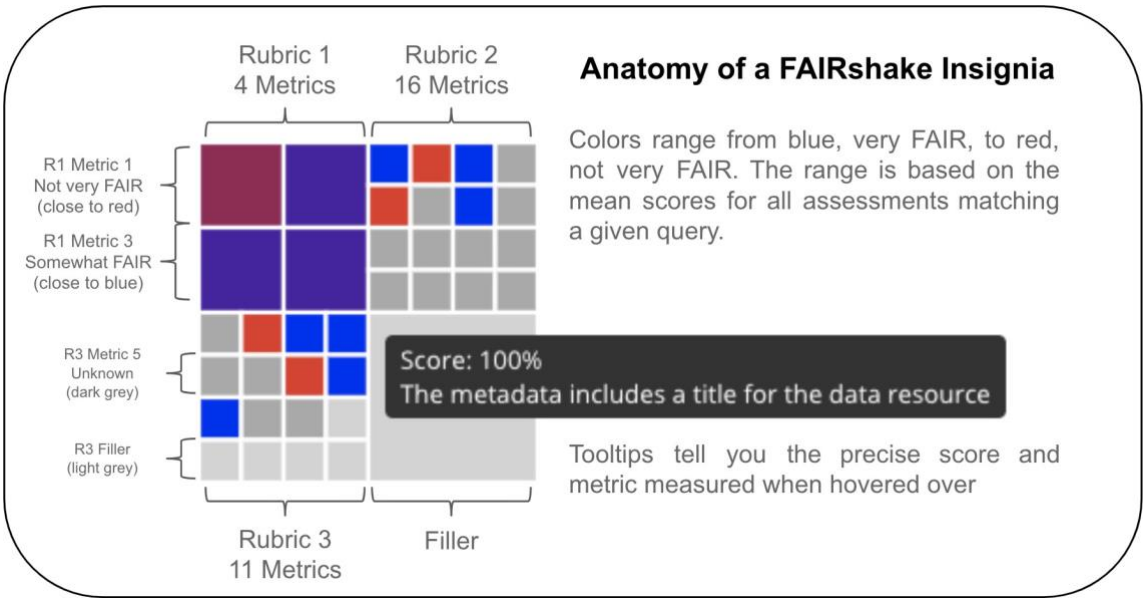


Fig. 25 FAIRshake insignia visualization for various rubrics that describe a digital object. Each of the square's colors correspond to the FAIRness of a particular metric.

The FAIRshake insignias capture a visual snapshot overview of a resource aggregated assessments. Interactive tooltips shown by hovering over a particular square reveal which metric is represented by that square. Clicking on a given box will bring you to a landing page with detailed information about the metric.

FAIRshake Analytics

Any project page with FAIR assessments will contain a Project Analytics page. This page displays informative visualizations pertaining to assessors' evaluations of the digital objects within the project. The data displayed by these visualizations include the frequencies of a particular answer (yes, no, yes/but, no/but) to a metric ranked by the number of responses, the proportion of respondents that used a specific rubric to evaluate the FAIRness of digital objects within the project, as well as the average respondent ratings for each metric in a particular rubric.

V. Advanced Topics

An automated assessment working example

With machine-readable metadata, we can assess FAIRness in an automatic fashion based on the fields available to the automated assessor script. For example, scripts that convert dataset objects hosted by NIH Common Fund Data Coordinating Center (DCCs) into a uniform metadata model, such as the cross-cut metadata model (C2M2) [5] for the Common Fund Data Ecosystem (CFDE) are available from here, and scripts to assess that unified metadata for its compliance with the CFDE Rubric are here. We produced reports over time that provide the assessments that were executed on the CFDE portal, which contains the C2M2 compliant metadata of participating DCCs. This report is summarized here. The assessment script can be executed once you have generated a C2M2 compatible metadata file. This principle applies for any assessment of data that must comply with a particular metadata format. If you have a frictionless datapackage containing your metadata, you can perform a FAIR assessment on that datapackage to identify gaps in your metadata.

Please note that you may need access to the CFDE FAIR Repo to access these scripts.

Creating an automated assessment

For assessments on completely new sets of digital objects with a completely new rubric, you need to build your own automated assessments. We will walk through how one example.

Certain standards are well-defined and designed in a way that makes it possible to computationally verify whether a digital object is complying with the standard. In an ideal world, all standards should be made in this way, such that an automated mechanisms exist for confirming compliance. However, in practice many standards are not harmonized.

Some examples of well-defined standards are TCP/IP and HTTP. The effectiveness of these standards and their adoption enables the internet to function and grow as it does. Another, more relevant standard is RDF. RDF defines a way to serialize metadata. It permits harmonization via ontologies or shape constraint languages (such as SHACL). Another standard that is not explicitly based on RDF is JSON Schema. JSON Schema builds off of JSON and allows one to use JSON to define what is a valid JSON instance of some metadata. A JSON Schema document can effectively become its own standard given that it is well described and validatable using a JSON Schema validator. In the case of assessing digital objects that comply with standards that are defined using mechanisms easily validated, automated assessments become simple. In many cases such automated assessments involve using already constructed mechanisms for asserting compliance with those standards. In the case that those standards are not well-defined, the best course of action would be to convert those digital objects to an alternative and validatable standard, or alternatively formally codify the standard. In either case, this activity is already FAIRification. We have to do this

step for automated assessments because we can’t measure compliance with a standard if we don't have a quantifiable machine-readable standard.

Case Study: Performing an Automated Assessment on DATS

One can think of an automated assessment as a unit/integration test for compliance with a standard. Ideally, this test will reveal issues with integration at the digital object provider level for the benefit of the consumer of those digital objects. Automated assessments are only possible on existing machine-readable metadata and validatable standards, such as DATS [6]. As such we will utilize DATS for our assessment. We assess compliance with DATS and go further with several additional 'optional' parts of DATS including ontological term verification and other sanity checks.

While there are several ways one can go about making an assessment, one way is to construct the rubric and metrics metadata while you construct the code to assert that metric.

```
rubric = {
  '@id': 25, # ID in FAIRshake 'name': 'NIH
  CFDE Interoperability',
  'description': 'This rubric identifies aspects of the metadata models which promote interoperable dataset querying and
filtering',
  'metrics': {},
}

def metric(schema):
  ''' A python decorator for registering a metric for the rubric. Usage: @metric({
    '@id': unique_id,
    'metric': 'metadata'
  })
  def _(asset):
    yield { 'value': 1.0, 'comment': 'Success' } '''
  global rubric
  def wrapper(func):
    rubric['metrics'][schema['@id']] = dict(schema, func=func)
```

```
  setattr(wrapper, '__name__', schema['name']) return
  wrapper

def assess(rubric, doc):
  ''' How to use use this rubric for assessing a document. Usage: assess(rubric, {
  "your": "metadata" })
  '''
  assessment = {  '@type':
    'Assessment', 'target': doc,
    'rubric': rubric['@id'],
    'answers': []
  }
  # print(assessment)
  for metric in rubric['metrics'].values():
    # print('Checking {}...'.format(metric['name'])) for
    answer in metric['func'](doc):
      # print(' => {}'.format(answer))
      assessment['answers'].append({
        'metric': { k: v for k, v in metric.items() if k != 'func' }, 'answer':
        answer,
      })
  return assessment
```

With these functions, all we have left to do is to define the metrics and their metadata, then the assess function can operate on a given document. Let's write a metric for assessing DATS:

```
@metric({
  '@id': 107, # ID in FAIRshake 'name':
  'DATS',
  'description': 'The metadata properly conforms with the DATS metadata specification', 'principle':
  'Findable',
})
def _(doc):
  from jsonschema import Draft4Validator
  errors = list(Draft4Validator({'$ref': 'http://w3id.org/dats/schema/dataset_schema.json'}).iter_errors(doc)) yield {
    'value': max(1 - (len(errors) / 100), 0),
    'comment': 'DATS JSON-Schema Validation results in {} error(s)\n{}'.format( len(errors) if errors
      else 'no',
      '\n'.join(map(str, errors))
    ).strip(),
  }

# ... additional metrics ...
```

With this added metric, which uses JSONSchema to validate the conformance of the metadata document to the DATS metadata model, an assessment would now produce answers for this

specific metric. We have normalized the answers between 0 and 1. Hence, 1 is for full conformance or a 0 for 100 or less validation errors. It's important to note that this is not the complete picture, perhaps you have a field for a landing page, but that website is down. This can be assessed too.

```
@metric({
  '@id': 16, # ID in FAIRshake 'name':
  'Landing Page',
  'description': 'A landing page exists and is accessible', 'principle':
  'Findable',
})
def _(doc): landingPages =
  set(
    node['access']['landingPage'] for node
    in jsonld_frame(doc, { '@type':
    'DatasetDistribution', 'access': {
        'landingPage': {},
      }
    })['@graph']
    if node['access'] and node['access']['landingPage']
  )
  if landingPages:
    for landingPage in landingPages:
      if requests.get(landingPage).status_code < 400:
        yield { 'value':
          1,
          'comment': 'Landing page found {} and seems to be accessible'.format(landingPage)
        }
      else:
        yield {
          'value': 0.75,
          'comment': 'Landing page found {} but seems to report a problem'.format(landingPage)
              }
          else:
    yield { 'value':
      0,
      'comment': 'Could not identify any landing pages'
    }
```

Above we have an example which uses JSON-LD framing to find landing pages. For each of those landing pages, we attempt to load the page and expect to receive back a reasonable HTTP status code. This is a value less than 400, specifically, 200-299 for success, or 300-399 for redirects. This could be improved further to be more stringent. In other words, to ensure we can find the title of our document on the landing page or something along those lines. However, even this basic loose criterion is not always satisfied. Ultimately this can become a command line application that we run in parallel on lots of DATS metadata. You can refer to the scripts here for examples on how you can accomplish this. It is also possible to resolve additional metadata in the process of the assessment

through forward chaining or other methods. An example of an assessment like that is also provided in that repository: data_citation_assessment.py which uses a URL to negotiate and resolve microdata according to this Data citation paper's guidelines.

Embedding the FAIRshake insignia in my website

The FAIRshake insignia can be embedded within any website. For example, we added FAIR insignias to datasets hosted on the SigCom LINCS data portal (Fig. 26).

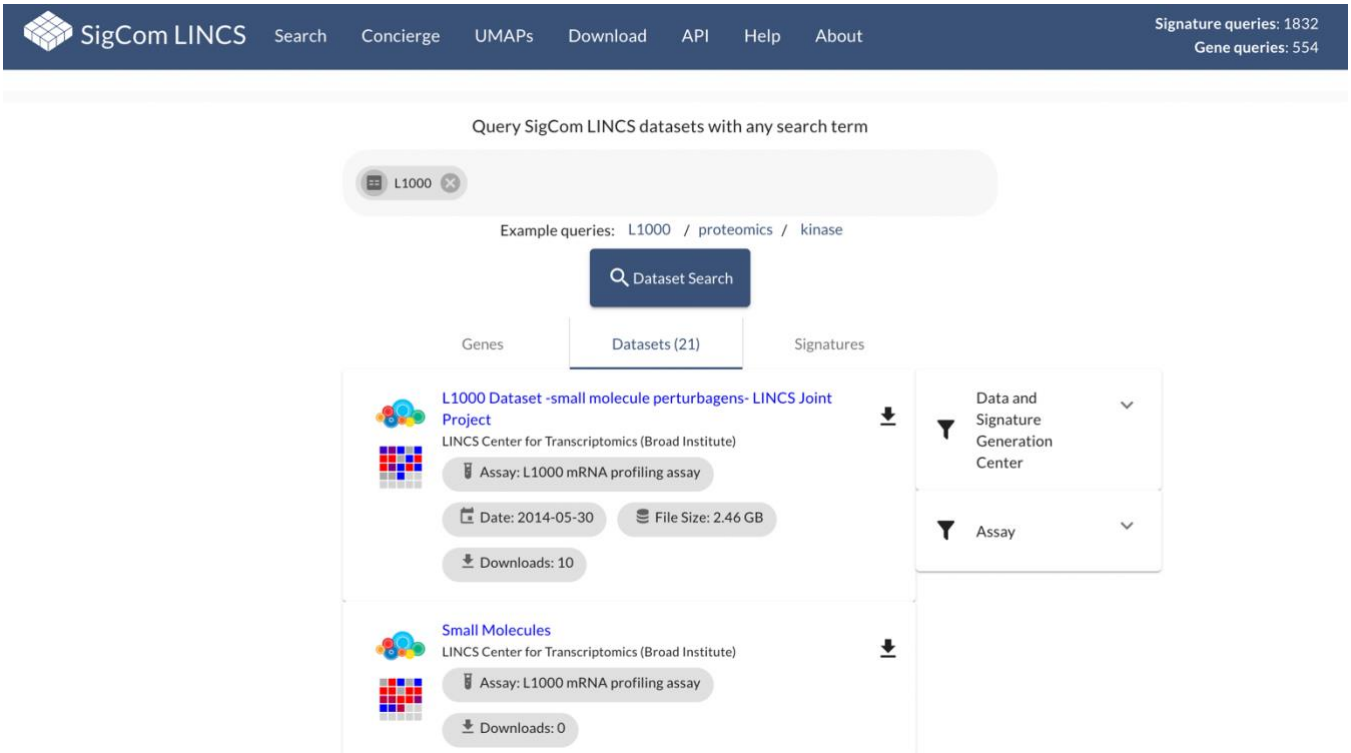


Fig. 26 Screenshot from the metadata search engine of SigCom LINCS displays an insignia based on the FAIR assessment of the datasets hosted by the portal.

To display the insignia, FAIRshake can process either a globally unique identifiers.org resolvable CURI or a fully resolvable URL, corresponding to the digital object registered in FAIRshake. For example, we demonstrate how to create an insignia from a project within FAIRshake. The project is https://fairshake.cloud/project/87/ so the ID is 87.

Using RequireJS

Demo: http://jsfiddle.net/tybx32gu/17/

```
<body>
  <div id="insignia" style="width: 40px; height: 40px;"></div>
  <script>
    require([
      'https://fairshake.cloud/v2/static/scripts/insignia.js'
```

```
    ], function(insignia) {
      insignia.build_svg_from_score(
        document.getElementById('insignia'), { project:
          87,
          url: 'https://your_fully_resolvable_id',
        }
      )
    })
  </script>
</body>
```

Using npm

NPM Package: https://github.com/MaayanLab/FAIRshakeInsignia

```
import { build_svg_from_score } from 'fairshakeinsignia'

build_svg_from_score(document.getElementById('insignia'), { project:
  87,
  url: 'https://your_fully_resolvable_id',
})
```

VI. REFERENCES

1. Wilkinson MD, Dumontier M, Aalbersberg IJ, Appleton G, Axton M, Baak A, Blomberg N, Boiten JW, da Silva Santos LB, Bourne PE, Bouwman J, Brookes AJ, Clark T, Crosas M, Dillo I, Dumon O, Edmunds S, Evelo CT, Finkers R, Gonzalez-Beltran A, Gray AJ, Groth P, Goble C, Grethe JS, Heringa J, 't Hoen PA, Hooft R, Kuhn T, Kok R, Kok J, Lusher SJ, Martone ME, Mons A, Packer AL, Persson B, Rocca-Serra P, Roos M, van Schaik R, Sansone SA, Schultes E, Sengstag T, Slater T, Strawn G, Swertz MA, Thompson M, van der Lei J, van Mulligen E, Velterop J, Waagmeester A, Wittenburg P, Wolstencroft K, Zhao J, Mons B. The FAIR Guiding Principles for scientific data management and stewardship. Sci Data. 2016 Mar 15;3:160018. doi: 10.1038/sdata.2016.18. Erratum in: Sci Data. 2019 Mar 19;6(1):6. PMID: 26978244; PMCID: PMC4792175.
2. Clarke DJB, Wang L, Jones A, Wojciechowicz ML, Torre D, Jagodnik KM, Jenkins SL, McQuilton P, Flamholz Z, Silverstein MC, Schilder BM, Robasky K, Castillo C, Idaszak R, Ahalt SC, Williams J, Schurer S, Cooper DJ, de Miranda Azevedo R, Klenk JA, Haendel MA, Nedzel J, Avillach P, Shimoyama ME, Harris RM, Gamble M, Poten R, Charbonneau AL, Larkin J, Brown CT, Bonazzi VR, Dumontier MJ, Sansone SA, Ma'ayan A. FAIRshake: Toolkit to Evaluate the FAIRness of Research Digital Resources. Cell Syst. 2019 Nov 27;9(5):417-421. doi: 10.1016/j.cels.2019.09.011. Epub 2019 Oct 30. PMID: 31677972; PMCID: PMC7316196.
3. Koleti A, Terryn R, Stathias V, Chung C, Cooper DJ, Turner JP, Vidovic D, Forlin M, Kelley TT, D'Urso A, Allen BK, Torre D, Jagodnik KM, Wang L, Jenkins SL, Mader C, Niu W, Fazel M, Mahi N, Pilarczyk M, Clark N, Shamsaei B, Meller J, Vasiliauskas J, Reichard J, Medvedovic M, Ma'ayan A, Pillai A, Schürer SC. Data Portal for the Library of Integrated Network-based Cellular Signatures (LINCS) program: integrated access to diverse large-scale cellular perturbation response data. Nucleic Acids Res. 2018 Jan 4;46(D1):D558-D566. doi: 10.1093/nar/gkx1063. PMID: 29140462; PMCID: PMC5753343.
4. Wilkinson MD, Sansone SA, Schultes E, Doorn P, Bonino da Silva Santos LO, Dumontier M. A design framework and exemplar metrics for FAIRness. Sci Data. 2018 Jun 26;5:180118. doi: 10.1038/sdata.2018.118. PMID: 29944145; PMCID: PMC6018520.
5. Amanda L Charbonneau, Arthur Brady, C. Titus Brown, Susanna-Assunta Sansone, Avi Ma'ayan, Rick Wagner, Robert Carter, Rayna M Harris, Alicia Gingrich, Marisa C.W. Lim, James B Munro, Daniel J.B. Clarke, Heather H Creasy, Philippe Rocca-Serra, Minji Jeon, R. Lee Liming, Robert E. Schuler, Cia Romano, Kyle Chard, Michelle Giglio, Suvarna Nadendla, Theresa K Hodges, Meisha Mandal, Saranya Canchi, Alex Waldrop, Owen White. Making Common Fund data more findable: Catalyzing a Data Ecosystem bioRxiv 2021.11.05.467504; doi: https://doi.org/10.1101/2021.11.05.467504
6. Sansone SA, Gonzalez-Beltran A, Rocca-Serra P, Alter G, Grethe JS, Xu H, Fore IM, Lyle J, Gururaj AE, Chen X, Kim HE, Zong N, Li Y, Liu R, Ozyurt IB, Ohno-Machado L. DATS, the data tag suite to enable discoverability of datasets. Sci Data. 2017 Jun 6;4:170059. doi: 10.1038/sdata.2017.59. PMID: 28585923; PMCID: PMC5460592.