Data Stewards and Data Management Principles

(LTAT.02.014)

Priit Adler
Research Fellow, University of Tartu
Elixir EE, BIIT
Recap

Data steward
Open data
FAIR
RDM
FAIR principle

Findability, Accessibility, Interoperability, and Reuse of digital assets

https://www.go-fair.org/fair-principles/
FAIR principles

Findability, Accessibility, Interoperability, and Reuse of digital assets

To be Findable:
F1. (meta)data are assigned a globally unique and persistent identifier
F2. data are described with rich metadata (defined by R1 below)
F3. metadata clearly and explicitly include the identifier of the data it describes
F4. (meta)data are registered or indexed in a searchable resource

https://www.go-fair.org/fair-principles/
The first step in (re)using data is to find them. Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services.
Discoverability of data

Metadata Types for Search
- Content Description
- Provenance
- License
- Revenue
- Credentials
- Quality / Metrics
- Release Schedule
- Data Format
- Data Access

Metadata Types for Use
- URI Design Principles
- Machine Access to Data
- Format Specification
How to find appropriate standard metadata for datasets or samples?

There are multiple standards for different types of data, ranging from **generic** dataset descriptions (e.g. DCAT, Dublin core, (bio)schema.org) to **specific data types** (e.g. MIABIS for biosamples). Therefore, *how to find standard metadata*, and *how to find an appropriate repository for depositing your data* become relevant questions.

Decide at the beginning of the project what are the **recommended repositories** for your data types.

- Note that you can use several repositories if you have different data types.
- Distinguish between generic (e.g. Zenodo) and data type (technique) specific repositories (e.g. EBI repositories).

https://rdmkit.elixir-europe.org/metadata_management.html
How to find appropriate standard metadata for datasets or samples?

If you don’t know yet what repository you will use, look for what is the recommended minimal information (i.e. “Minimum Information ...your topic”, e.g. MIAME or MINSEQE or MIAPPE) required for your type of data in your community, or other metadata, at the following resources:

- Research Data Alliance (RDA): Metadata Dictionary: Standards
- FAIRsharing.org at “Standards” and “Collections”
- The Digital Curation Centre (DCC): List of Metadata Standards

https://rdmkit.elixir-europe.org/metadata_management.html
Data on the Web Best Practices

Best Practice 1: Provide metadata
Best Practice 2: Provide descriptive metadata
Best Practice 3: Provide structural metadata
Best Practice 4: Provide data license information
Best Practice 5: Provide data provenance information
Best Practice 6: Provide data quality information
Best Practice 7: Provide a version indicator
Best Practice 8: Provide version history
Best Practice 9: Use persistent URIs as identifiers of datasets
Best Practice 10: Use persistent URIs as identifiers within datasets
Best Practice 11: Assign URIs to dataset versions and series
Best Practice 12: Use machine-readable standardized data formats
Best Practice 13: Use locale-neutral data representations
Best Practice 14: Provide data in multiple formats
Best Practice 15: Reuse vocabularies, preferably standardized ones
Best Practice 16: Choose the right formalization level
Best Practice 17: Provide bulk download
Best Practice 18: Provide Subsets for Large Datasets
Best Practice 19: Use content negotiation for serving data available in multiple formats
Best Practice 20: Provide real-time access
Best Practice 21: Provide data up to date
Best Practice 22: Provide an explanation for data that is not available
Best Practice 23: Make data available through an API
Best Practice 24: Use Web Standards as the foundation of APIs
Best Practice 25: Provide complete documentation for your API
Best Practice 26: Avoid Breaking Changes to Your API
Best Practice 27: Preserve identifiers
Best Practice 28: Assess dataset coverage
Best Practice 29: Gather feedback from data consumers
Best Practice 30: Make feedback available
Best Practice 31: Enrich data by generating new data
Best Practice 32: Provide Complementary Presentations
Best Practice 33: Provide Feedback to the Original Publisher
Best Practice 34: Follow Licensing Terms
Best Practice 35: Cite the Original Publication

https://www.w3.org/TR/2017/REC-dwbp-20170131
Metadata

Best Practice 2: Provide descriptive metadata

Provide metadata that describes the overall features of datasets and distributions.

Why

Explicitly providing dataset descriptive information allows user agents to automatically discover datasets available on the Web and it allows humans to understand the nature of the dataset and its distributions.

Intended Outcome

Humans will be able to interpret the nature of the dataset and its distributions, and software agents will be able to automatically discover datasets and distributions.

Possible Approach to Implementation

Descriptive metadata can include the following overall features of a dataset:

- The title and a description of the dataset.
- The keywords describing the dataset.
- The date of publication of the dataset.
- The entity responsible (publisher) for making the dataset available.
- The contact point for the dataset.
- The spatial coverage of the dataset.
- The temporal period that the dataset covers.
- The date of last modification of the dataset.
- The themes/categories covered by a dataset.

Descriptive metadata can include the following overall features of a distribution:

- The title and a description of the distribution.
- The date of publication of the distribution.
- The media type of the distribution.

The metadata-representable version of the descriptive metadata can be provided using the vocabulary recommended by W3C to describe datasets, i.e., the Data Catalog Vocabulary (VOCAB-DCAT). This provides a framework in which datasets can be described as abstract entities.
The RDA Metadata Standards Directory Working Group is supported by individuals and organizations involved in the development, implementation, and use of metadata for scientific data. The overriding goal is to develop a collaborative, open directory of metadata standards applicable to scientific data can help address infrastructure challenges.

https://rd-alliance.github.io/metadata-directory/
ENA virus pathogen reporting standard checklist

Minimum information about a virus pathogen. A checklist for reporting metadata of virus pathogen samples associated with genomic data. This minimum metadata standard was developed by the COMPARE platform for submission of virus surveillance and outbreak data (such as Ebola) as well as virus isolate information.

https://www.ebi.ac.uk/ena/browser/view/ERC000033
Data provenance

Best Practice 5: Provide data provenance information

Provide complete information about the origins of the data and any changes you have made.

Why

Provenance is one means by which consumers of a dataset judge its quality. Understanding its origin and history helps one determine whether to trust the data and provides important interpretive context.

Intended Outcome

Humans will know the origin and history of the dataset and software agents will be able to automatically process provenance information.

Possible Approach to Implementation

The machine-readable version of the data provenance can be provided using an ontology recommended to describe provenance information, such as W3C’s Provenance Ontology [PROV-O].

EXAMPLE 5

Machine-readable

The example below shows the machine-readable metadata for the bus stops dataset with the inclusion of the provenance metadata. The properties `dct:creator`, `dct:publisher` and `dct:issued` are used to give information about the origin of the dataset. The property `prov:actedOnBehalfOf` is used to designate that Adrian acted on behalf of the Transport Agency of MyCity.

```xml
:stops-2015-05-05
  a dct:Dataset, prov:Entity;
  dct:title "Bus stops of MyCity";
  dct:keyword "transport", "mobility", "bus";
  dct:issued "2015-05-05"^^xsd:date;
  dct:contactPoint <http://data.mycity.example.com/transport/contact>;
  dct:spatial <http://sws.geonames.org/3399415>;
  dct:publisher :transport-agency-mycity;
  dct:accrualPeriodicity <http://purl.org/linked-data/sdmx/2009/code#Freq-A>
    <http://id.loc.gov/vocabulary/iso639-1/en>;
  dct:creator :adrian
```

https://www.w3.org/TR/2017/REC-dwbp-20170131/#DataProvenance
https://www.w3.org/TR/prov-o/
Data provenance

Data provenance is metadata that is paired with records that details the origin, changes to, and details supporting the confidence or validity of data. Data provenance is important for tracking down errors within data and attributing them to sources. Additionally, data provenance can be useful in reporting and auditing for business and research processes.

https://blog.diffbot.com/knowledge-graph-glossary/data-provenance/
Provide version indicator

Best Practice 7: Provide a version indicator

Assign and Indicate a version number or date for each dataset.

Why

Version information makes a revision of a dataset uniquely identifiable. Uniqueness can be used by data consumers to determine whether and how data has changed over time and to determine specifically which version of a dataset they are working with. Good data versioning enables consumers to understand if a newer version of a dataset is available. Explicit versioning allows for repeatability in research, enables comparisons, and prevents confusion. Using unique version numbers that follow a standardized approach can also set consumer expectations about how the versions differ.

Intended Outcome

Humans and software agents will easily be able to determine which version of a dataset they are working with.

Possible Approach to Implementation

The best method for providing versioning information will vary according to the context; however, there are some basic guidelines that can be followed, for example:

- Include a unique version number or date as part of the metadata for the dataset.
- Use a consistent numbering scheme with a meaningful approach to incrementing digits, such as [SchemaVer].
- If the data is made available through an API, the URI used to request the latest version of the data should not change as the versions change, but it should be possible to request a specific version through the API.
- Use Memento [RFC7089], or components thereof, to express temporal versioning of a dataset and to access the version that was operational at a given datetime. The Memento protocol aligns closely with the approach for assigning URIs to versions that is used for W3C specifications, described below.

The Web Ontology Language [OWL2-QUICK-REFERENCE] and the Provenance, Authoring and versioning Ontology [PAV] provide a number of annotation properties for version information.
g:GOSt performs functional enrichment analysis, also known as over-representation analysis (ORA) or gene set enrichment analysis, on input gene list. It maps genes to known functional information sources and detects statistically significantly enriched terms. We regularly retrieve data from Ensembl database and fungi, plants or metazoa specific versions of Ensembl Genomes, and parasite specific data from WormBase ParaSite.

https://www.ebi.ac.uk/ena/browser/view/ERC000033
Persistent URIs *(uniform resource identifier)*

---

**Best Practice 9: Use persistent URIs as identifiers of datasets**

Identify each dataset by a carefully chosen, persistent URI.

**Why**

Adopting a common identification system enables basic data identification and comparison processes by any stakeholder in a reliable way. They are an essential pre-condition for proper data management and reuse.

Developers may build URIs into their code and so it is important that those URIs persist and that they dereference to the same resource over time without the need for human intervention.

**Intended Outcome**

Datasets or information about datasets will be discoverable and citable through time, regardless of the status, availability or format of the data.

**Possible Approach to Implementation**

To be persistent, URIs must be designed as such. A lot has been written on this topic, see, for example, the European Commission’s Study on Persistent URIs [PURI] which in turn links to many other resources.

Where a data publisher is unable or unwilling to manage a URI space directly for persistence, an alternative approach is to use a redirection service such as Permanent Identifiers for the Web or pur.org. These provide persistent URIs that can be redirected as required so that the eventual location can be ephemeral. The software behind such services is freely available so that it can be installed and managed locally if required.

Digital Object Identifiers (DOIs) offer a similar alternative. These identifiers are defined independently of any Web technology but can be appended to a ‘URI stub.’ DOIs are an important part of the digital infrastructure for research data and libraries.

**EXAMPLE 9**

The URI http://data.mycity.example.com/transport/dataset/bus/stops has several features that support persistence:

- All names are subject to change over time but in choosing a domain name, it is reasonable for City to expect that the City will continue to exist and that it will continue to have a government. Therefore, while cases like Yugoslavia prove that even country names change and...
Standard identification mechanism

DOI

A DOI, or Digital Object Identifier, is a string of numbers, letters and symbols used to permanently identify an article or document and link to it on the web. A DOI will help your reader easily locate a document from your citation. Think of it like a Social Security number for the article you’re citing — it will always refer to that article, and only that one. While a web address (URL) might change, the DOI will never change.

ORCID

Open Researcher and Contributor ID

ROR

Research Organisation Registry

https://ror.org/
https://info.orcid.org/what-is-orcid/
https://library.uic.edu/help/article/1966/what-is-a-doi-and-how-do-i-use-them-in-citations
Machine-readable standardised data formats

Best Practice 12: Use machine-readable standardized data formats

Make data available in a machine-readable, standardized data format that is well suited to its intended or potential use.

Why

As data becomes more ubiquitous, and datasets become larger and more complex, processing by computers becomes ever more crucial. Posting data in a format that is not machine-readable places severe limitations on the continuing usefulness of the data. Data becomes useful when it has been processed and transformed into information. Note that there is an important distinction between formats that can be read and edited by humans using a computer and formats that are machine-readable. The latter term implies that the data is readily extracted, transformed and processed by a computer.

Using non-standard data formats is costly and inefficient, and the data may lose meaning as it is transformed. By contrast, standardized data formats enable interoperability as well as future uses, such as remixing or visualisation, many of which cannot be anticipated when the data is first published. It is also important to note that most machine-readable standardized formats are also locale-neutral.

Intended Outcome

Machines will easily be able to read and process data published on the Web and humans will be able to use computational tools typically available in the relevant domain to work with the data.

Possible Approach to Implementation

Make data available in a machine-readable standardized data format that is easily parseable including but not limited to CSV, XML, HDF5, JSON and RDF serialization syntaxes like RDF/XML, JSON-LD, or Turtle.

EXAMPLE 12

Adrian knows that tabular data is commonly used on the Web and he decides to use CSV as the data format for one of the distributions of the bus stops dataset. To facilitate data processing, he uses the Model for Tabular Data and Metadata on the Web for publishing the CSV distribution (stops-2015-05-05.csv). The example below presents a fragment of the CSV distribution which complies with the structural metadata defined in Example 4.

Identifier,Name,Description,Latitude,Longitude,ZONE,URL
645 Castle Avenue, Sunset Drive, 33.513652, -88.755739, 29 http://data.mycity.example.com/stopdata/data.mycity.example.com/stopdata/645-castle-avenue-sunset-drive-33.513652,-88.755739,29/
Data format (netCDF)

```
variables:
  float lat(lat) ;
  lat::long_name = "Latitude" ;
  lat::units = "degree_north" ;
  float lon(lon) ;
  lon::long_name = "Longitude" ;
  lon::units = "degree_east" ;
  int time(time) ;
  time::long_name = "Time" ;
  time::units = "days since 1895-01-01" ;
  time::calendar = "gregorian" ;
  float rainfall(rainfall) ;
  rainfall::long_name = "Precipitation" ;
  rainfall::units = "mm yr^-1" ;
  rainfall::missing_value = -99999 ;

// global attributes:
	:time = "Historical Climate Scenarios" ;
	:Conventions = "CF-1.0" ;

data
  lat = 48.75, 48.25, 47.75 ;
  lon = -124.25, -123.75, -123.25, -122.75 ;
  time = 364, 730 ;
  rainfall =
    761, 3265, 2184, 1812, 1405, 688, 365, 266, 328, 455, 524, 877, 1019, 714, 865, 697, 927, 926, 1452, 626, 275, 221, 196, 223 ;
```

NetCDF Library Architecture

```
ncdump	C apps
ncgen	F90 apps
nccopy	perl apps
ncgen	F77 apps
ncdisp	old perl API
ncdisp	C++ apps
ncdisp	python, ... apps

libdispatch (C API)
libsrc (classic)
libsrc4 (netCDF-4)
libdap2
netCDF classic
netCDF-4
HDF5
HDF4 SD
oc (DAP2 C library)
libcurl
remote data (netCDF, GRIB, ...)

User
Unidata
3rd party
```

```
temperature
precipitation
latitude
longitude

x
reference_time
y
t
```
JSON: JavaScript Object Notation

```json
{
  "firstName": "John",
  "lastName": "Smith",
  "isAlive": true,
  "age": 27,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021-3100"
  },
  "phoneNumbers": [
    { "type": "home", "number": "212 555-1234" },
    { "type": "office", "number": "646 555-4567" }
  ],
  "children": [],
  "spouse": null
}
```
Best Practice 15: Reuse vocabularies, preferably standardized ones

Use terms from shared vocabularies, preferably standardized ones, to encode data and metadata.

Why

Use of vocabularies already in use by others captures and facilitates consensus in communities. It increases interoperability and reduces redundancies, thereby encouraging reuse of your own data. In particular, the use of shared vocabularies for metadata (especially structural, provenance, quality and versioning metadata) helps to compare and automatic processing of both data and metadata. In addition, referring to codes and terms from standards helps to avoid ambiguity and clashes between similar elements or values.

Intended Outcome

Interoperability and consensus among data publishers and consumers will be enhanced.

Possible Approach to Implementation

The Vocabularies section of the W3C Best Practices for Publishing Linked Data [LD-BP] provides guidance on the discovery, evaluation and selection of existing vocabularies.

Organizations such as the Open Geospatial Consortium (OGC), ISO, W3C, WMO, libraries and research data services, etc. provide lists of codes, terminologies and Linked Data vocabularies that can be used by everyone. A key point is to make sure the dataset, or its documentation, provides enough (human- and machine-readable) context so that data consumers can retrieve and exploit the standardized meaning of the values. In the context of the Web, using unambiguous, Web-based identifiers (URIs) for standardized vocabulary resources is an efficient way to do this, noting that the same URI may have multilingual labels attached for greater cross-border interoperability. The European Union's multilingual thesaurus, Eurovoc, provides a prime example.

EXAMPLE 15

1. The DCAT vocabulary expresses metadata concerning datasets [VOCAB-DCAT] and re-uses elements from several pre-existing vocabularies: Dublin Core, FOAF, SKOS and vCard. Reusing Dublin Core properties like dcat:title instead of creating new ones (say, dcat:title) enables DCAT-based metadata to be consumed by any application that can read and manipulate Dublin Core statements.

More in the digital culture section: the data model for Europeana (EDM) also makes extensive uses of...
Summary

To be Findable:
F1. (meta)data are assigned a globally unique and persistent identifier
F2. data are described with rich metadata (defined by R1)
F3. metadata clearly and explicitly include the identifier of the data it describes
F4. (meta)data are registered or indexed in a searchable resource