FAIR Software?
How can we make easier to find, access, deposit and reuse software?

Slides: https://doi.org/10.6084/m9.figshare.5620690

22nd November 2017, Dealing With Data 2017, Edinburgh
Neil Chue Hong (@npch), Software Sustainability Institute
ORCID: 0000-0002-8876-7606 | N.ChueHong@software.ac.uk
Findable
Accessible
Interoperable
Reusable


Slides: https://doi.org/10.6084/m9.figshare.5620690
“FAIRness is a prerequisite for proper data management and data stewardship”

What does this mean when applied to software?
Research Software Workflow

Develop → Share → Preserve

Developed and versioned using code repository
Published via code repository or website
Deposited in digital repository with paper / for preservation
Findable

assigned unique identifier

described with rich metadata

indexed in searchable resource
First: Use a code repository

- Makes software management *much* easier
- Will help make software FAIR (as you’ll see)

https://education.github.com/discount_requests/new
Identifying software

- Identifying software is hard
- Giving software an identifier is easy
Giving software an identifier

Making Your Code Citable
© 10 minute read

Digital Object Identifiers (DOI) are the backbone of the academic reference and metrics system. If you’re a researcher writing software, this guide will show you how to make the work you share on GitHub citable by archiving one of your GitHub repositories and assigning a DOI with the data archiving tool Zenodo.

ProTip: This tutorial is aimed at researchers who want to cite GitHub repositories in academic literature. Provided you’ve already set up a GitHub repository, this tutorial can be completed without installing any special software. If you haven’t yet created a project on GitHub, start first by uploading your work to a repository.

42,000 software packages with DOIs (2013-2017)
Rich metadata is the issue

• Many programmers don’t like to describe their software
• Not always clear what should be described to make things findable
• However:
  ▪ A lot of metadata can be automatically generated
  ▪ Start small and improve
• A README file is the human-readable starting point
CodeMeta: A Rosetta Stone for Software Metadata

Crosswalk for GitHub API

<table>
<thead>
<tr>
<th>Property</th>
<th>GitHub</th>
</tr>
</thead>
<tbody>
<tr>
<td>codeRepository</td>
<td>html_url</td>
</tr>
<tr>
<td>programmingLanguage</td>
<td>languages_url</td>
</tr>
<tr>
<td>downloadUrl</td>
<td>archive_url</td>
</tr>
<tr>
<td>author</td>
<td>login</td>
</tr>
<tr>
<td>dateCreated</td>
<td>created_at</td>
</tr>
<tr>
<td>dateModified</td>
<td>updated_at</td>
</tr>
<tr>
<td>license</td>
<td>license</td>
</tr>
<tr>
<td>description</td>
<td>description</td>
</tr>
<tr>
<td>identifier</td>
<td>id</td>
</tr>
<tr>
<td>name</td>
<td>full_name</td>
</tr>
<tr>
<td>issueTracker</td>
<td>issues_url</td>
</tr>
</tbody>
</table>

Crosswalk for Zenodo metadata

Zenodo.org is a data archive based at CERN which is popularly used to archive and provide DOIs to academic software from GitHub, as described in the official GitHub guide to Making your code citable.

<table>
<thead>
<tr>
<th>Property</th>
<th>Zenodo</th>
</tr>
</thead>
<tbody>
<tr>
<td>codeRepository</td>
<td>relatedLink</td>
</tr>
<tr>
<td>applicationCategory</td>
<td>communities</td>
</tr>
<tr>
<td>author</td>
<td>creators</td>
</tr>
<tr>
<td>datePublished</td>
<td>date_published</td>
</tr>
<tr>
<td>funder</td>
<td>contributors.Funder</td>
</tr>
<tr>
<td>keywords</td>
<td>keywords</td>
</tr>
<tr>
<td>license</td>
<td>license</td>
</tr>
<tr>
<td>description</td>
<td>description/notes</td>
</tr>
<tr>
<td>identifier</td>
<td>id</td>
</tr>
<tr>
<td>name</td>
<td>name</td>
</tr>
</tbody>
</table>

Accessible

retrievable using standard, open, free protocol

metadata accessible even when software is no longer available
Software is easy to make accessible

- By using code repositories (for development) and digital repositories (for preservation) software is accessible
- By using a digital repository which issues DOIs, metadata is guaranteed to be kept available
- But... is this metadata useful?
  - It is for citation, which improves reuse and credit
    - Software Citation Principles. PeerJ Computer Science 2:e86. DOI: 10.7717/peerj-cs.86
  - ORCIDS make it easier to tie together
Interoperable

formal, accessible, shared
language for knowledge
representation

qualified references to other
metadata
CodeMeta built on Schema.org and JSON-LD

```json
{
   "@context": [
      "https://doi.org/10.5063/schema/codemeta-2.0",
      "http://schema.org"
   ],
   "@id": "https://doi.org/10.5281/zenodo.1048320",
   "@type": "SoftwareSourceCode",
   "identifier": "codemeta",
   "description": "The 'Codemeta' Project defines a 'JSON-LD' format for describing software metadata, as detailed at <https://codemeta.github.io>. This package provides utilities to generate, parse, and modify 'codemeta.jsonld' files automatically for R packages, as well as tools and examples for working with 'codemeta' 'JSON-LD' more generally.",
   "name": "codemeta: Generate CodeMeta Metadata for R Packages",
   "issueTracker": "https://github.com/codemeta/codemeta/issues",
   "license": "https://spdx.org/licenses/MIT",
   "version": "0.1.2",
   "programmingLanguage": {
      "@type": "ComputerLanguage",
      "name": "R",
      "version": "3.4.2",
      "url": "https://r-project.org"
   },
   "runtimePlatform": "R version 3.4.2 (2017-09-28)",
   "provider": {
      "@id": "https://cran.r-project.org",
      "@type": "Organization",
      "name": "Central R Archive Network (CRAN)"
   },
   "author": [
      {
         "@type": "Person",
         "givenName": "Carl",
         "familyName": "Boettiger",
         "email": "cboettig@gmail.com",
         "@id": "http://orcid.org/0000-0002-1642-628X"
      }
   ],
   "codeRepository": "https://github.com/ropensci/codemeta",
   "isPartOf": "https://ropensci.org",
   "keywords": ["metadata", "codemeta", "ropensci", "citation", "credit", "linked-data"],
   "relatedLink": "https://codemeta.github.io/codemeta",
   "contIntegration": "https://travis-ci.org/ropensci/codemeta",
   "developmentStatus": "active",
   "releaseNotes": "https://github.com/ropensci/codemeta/blob/master/NEWS.md",
   "readme": "https://github.com/ropensci/codemeta/blob/master/README.md",
   "fileSize": "662.402KB"
}
```
Reusable

clear and accessible license

detailed provenance

meet community standards
Clarity is quite complicated

• Software Package Data Exchange (SPDX) increasingly used for licensing information
  ▪ But all you need to know is that you should choose a commercial license or one from this list:
    • https://opensource.org/licenses

• Provenance is simple (from code repository) and difficult (from underlying research)
  ▪ Jupyter Notebooks can make things easier
  ▪ Principle of “recording your working” – see next talk!
Community standards depend on the community!

- CLARIAH (Arts and Humanities): https://github.com/CLARIAH/software-quality-guidelines
- ELIXIR (Life Sciences): https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5490478/
Research Software Workflow

describe

develop → share → preserve

Developed and versioned using code repository
Published via code repository or website
Deposited in digital repository with paper / for preservation

Software Sustainability Institute
Making Software FAIR

• Get an ORCID for yourself
• Develop your software in a code repository
• Choose a clear license for your software
• Keep essential metadata and README in your code repository up to date
  ▪ Codemeta.json file in future
• Archive major versions, to get a DOI
• Cite your software in your papers, and put your preferred citation in your README
Interested in doing more?

- FORCE11 Software Citation Implementation Working Group
  - [https://www.force11.org/group/software-citation-implementation-working-group](https://www.force11.org/group/software-citation-implementation-working-group)
- RDA Software Source Code Interest Group
  - [https://www.rd-alliance.org/groups/software-source-code-ig](https://www.rd-alliance.org/groups/software-source-code-ig)
- SWORD v3 development
  - [http://swordapp.org/](http://swordapp.org/)
- Software Sustainability Institute
  - [https://www.software.ac.uk/](https://www.software.ac.uk/)
Find out more about the SSI

• Community Engagement (Lead: Shoaib Sufi)
  ▪ Fellowship Programme
  ▪ Events and Workshops

• Consultancy (Lead: Steve Crouch)
  ▪ Open Call for Projects / Collaborations
  ▪ Software Evaluation

• Policy and Publicity (Lead: Simon Hettrick)
  ▪ Case Studies / Policy Campaigns
  ▪ Software and Research Blog

• Training (Lead: Aleksandra Nenadic)
  ▪ Software Carpentry / Data Carpentry (300+ students/year)
  ▪ Guides and Top Tips

• Journal of Open Research Software (Editor: Neil Chue Hong)

• Collaboration between universities of Edinburgh, Manchester, Oxford and Southampton
  Supported by EPSRC Grant EP/H043160/1 + EPSRC/ESRC/BBSRC grant EP/N006410/1

Software Sustainability Institute