Prof. Dr. A-S. Smith
am Lehrstuhl für Theoretische Physik I
Department für Physik
Friedrich-Alexander-Universität
Erlangen-Nürnberg

Advanced Python for Research Projects

Exercise sheet 6: Collaboration and FAIR principles

Problem 6.1 *FAIR principles - Findability*

We want to carry out final steps to achieve true FAIR publication standards. For this, we need to make our code/data uniquely addressible using an open, standard protocol like http(s)-links using the DOI-standard. A commonly used provider for digital object identifiers for publicly available data is ZENODO. We will use ZENODO to upload a release of our code and have it associated with a DOI that can then permanently point to that version.

a) Create an Account on ZENODO (`https://zenodo.org`). We recommend using your university email which you would use for scientific publications. You can also create an ORCID (`https://orcid.org/`), which is a unique identifier for scientists to avoid name confusion in publications and use that to log in to ZENODO.

b) To create the ZENODO archive, first, download the release zip from the git provider. Then start a new archive on ZENODO and upload the downloaded release zip in the file selection dialog.

c) To turn the mere code into a documented archive, add all necessary meta-information for your Project to be well-documented, including contributors, Usage-Information, a link to the git repository, etc. We also recommend re-stating most of the information presented in your `README` file here in the archive description text on ZENODO. You may want to specify the version identifier and the commit hash of the release to be very thorough.

d) Preview your inputs and check correctness. Click on the button to assign a DOI to your archive, as we do not yet have one. Then publish your archive and retrieve the DOI for your release. Be careful, you cannot modify the files in an archive once it has been published, so the files need to be in order before publication. Metadata can still be corrected after you hit publish without triggering a new version of the project being created with a new DOI.

e) Concerning FAIR principles: Why is it helpful that git retains author information for commits and releases?

f) Why do code comments and documentation constitute *metadata* for our project?

g) How does adding our archive to a repository like ZENODO satisfy the requirement of indexing and searchability?

h) Why should we include the reference to the git project and the commit? Which part of the FAIR principles is satisfied by this?

Problem 6.2 *FAIR principles - Accessibility*

The project has now been made publicly available with a DOI to be able to permanently access its (Meta)data.

a) Discuss, which steps for achieving accessibility have already been taken and which steps may still need to be taken at this point. You should consider each individual sub-aspect of the A-category of FAIR principles.

Problem 6.3 *FAIR principles - Interoperability*

We have now prepared code in a version controlled repository with a public identifier on ZENODO, which allows access to its release state and metadata.

a) Which parts of the steps taken up to this point open up our project to the aspects of Interoperability?

b) Which steps may still be necessary?

c) How does the documentation using a standard like the numpy/scipy notation with well-defined sections help in regards to both accessibility and interoperability?

d) Is the choice of Python as a language for our project also an aspect contributing to interoperability?

Problem 6.4 *FAIR principles - Reusability*

Our code includes detailed documentation and we have used the pdoc3-module to generate documentation that can be accessed separate from our code.

a) How does this level of documentation contribute to Reusability?

b) Why is it essential to include a `LICENSE` statement in our repository?

c) Why are the choice of python as a programming language and git as a standard version control system also essential for the aspect of reusability?