From a single-use script to a reusable Python package: assisting researchers in creating FAIR software

van Boheemen, Jelte
j.vanboheemen@uu.nl
Utrecht University, Netherlands, The

Baarda, Tijmen Christiaan
t.c.baarda@uu.nl
Utrecht University, Netherlands, The

Recently, it has become more and more common that individual researchers in the humanities create pieces of software to support their research. Research councils today often require all output to be available in open access, which implies that output in the form of software should be available as open source software. However, for various reasons, publishing source code itself is generally not enough to allow others to replicate the research results or to reuse it for their own purposes. In this poster presentation, we want to show that with the right professional but low-key technical support, researchers who write their own software can be enabled to make it useful for others as much as for themselves.

To promote the long-term value of research software, the international FAIR for Research Software working group (FAIR4RS) recently published a set of principles for making software FAIR, which stands for Findable, Accessible, Interoperable and Reusable (Barker et al: 2022). Making software FAIR according to these principles entails a number of formal requirements. Some of these, such as assigning a unique identifier and publishing it with a proper open source license, are more or less trivial to implement. Other principles, such as the requirement of providing good documentation and interoperability using commonly-used data formats and programming interfaces, is less obvious to implement for those who do not write software on a daily basis and in collaboration with others.

This poster will present a detailed step-by-step plan for collaboration with researchers to make their software FAIR, along with a number of examples from the Research Software Lab, part of the Centre for Digital Humanities at Utrecht University where we assisted researchers to make their smaller and larger pieces of software useful for a larger audience, adhering to the FAIR principles. Many researchers are used to writing computer programs but may not be very well introduced to common software engineering procedures. They can adapt their software with relatively little additional efforts from both the side of the researcher and the side of the technical support staff. Bleeker et al. (2022) have shown that while the importance of code literacy in the field of digital humanities is clear, many scholars believe their own is lacking. We believe that the proposed collaboration between researchers and technical support can contribute to the overall level of code literacy in the humanities.

At the Research Software Lab, we collaborate with researchers by providing advice, technical support and writing software. During the last decade, the Research Software Lab has mainly been developing software at the request of researchers, often in the form of web applications. In the recent years, it has become more and more common for researchers in the humanities to write their own software. We actively encourage the researchers in their efforts, but we also assist them not only in writing the software, but also to preserve it for future use. In our poster, we give a detailed step-by-step plan showing how we assist and stimulate researchers in this process.

In this poster, we identify three types of research software we encounter in our Lab: standalone research scripts, reusable libraries, and (web-)applications. Research scripts are often published and citable, but lack documentation and licenses which makes them reusable. On the other hand, large and complex applications often contain pieces of code that may be of use in other projects as well, and it is difficult to cite specific parts of the application. In the Lab, we help researchers to transform their research scripts as well as parts of existing web applications into reusable libraries. Thanks to the infrastructure that the Lab provides, the pieces of software we develop or maintain are preserved in a sustainable way and can be picked up by other researchers even after the original research project is discontinued. We offer insight into this process for both existing and new projects. We are convinced that FAIR research software is a reachable goal for any researcher who cares about the sustainability of their software, provided that the research institution provides the right technical support infrastructure.

Bibliography

Barker, Michelle / Chue Hong, Neil P. / Katz, Daniel S. / Lamprecht, Anna-Lena / Martinez-Ortiz, Carlos / Psomopoulos, Fotis / Harrow, Jennifer / Jael Castro, Leyla / Gruenpeter, Morane / Martinez, Paula Andrea / Honeyman, Tom (2022): “Introducing the FAIR Principles for research software”, in *Scientific Data* 9. https://doi.org/10.1038/s41597-022-01710-x [01.05.2023]

Bleeker, Elli / Koolen, Marijn / Beelen, Kaspar / Melgar, Liliana / van Zundert, Joris / Chambers, Sally (2022): “A Game of Persistence, Self-doubt, and Curiosity: Surveying Code Literacy in Digital Humanities”, in *DH Benelux Journal* 4: 1-28. https://doi.org/10.17613/03ds-9973 [01.05.2023]