EUROVIS
Leipzig 2023

Making Vis FAIR – Experiences from Computational Biology

Daniel Wiegreffe

- In Biology, many experiments are conducted multiple times
- Software is often used to analyze the experiment data
    - The software thereby also falls under the FAIR principles
- This results in certain requirements:
    - Software must remain available
    - Software must function stably
    - Software versioning in case of changes is very important

- This problem also applies to the visualization that is used:
  - Changes of visual representation with the same data are critical
  - Availability of all dependencies of the system must be ensured
  - Often the visualization systems are integrated into the analysis programs

But how can we deal with it?

Lessons from biology:

- Mandate disclosure of source code
    - The software remains available
    - No long-term solution
        - Compiler problem
        - External dependencies

- Usage and maintenance of package management systems like bioconda
    - Useful, but you have to rely on the care of the maintainers

Lessons from biology:

- Archiving all versions of the application
    - One still depends on the permanent availability.

- Usage of containerization
    - executable, versioned state of a program can be conserved with external dependencies
    - operates system independent
    - It is more complex, because in addition to the software the container has to be created.

- Software in visualization should be made available as source code and executable application

- The versions of the applications should be archived and permanently available
  - Zenodo, OSF, etc.

- Journals and conferences should make these requirements mandatory to ensure that this is implemented

Thank you for listening!

Daniel Wiegreffe
Leipzig University