



Contribution ID: 99

Type: **Tutorial or Skill-Up**

Enhancing Research Software Sustainability through Modular Open-Source Software Templates

Thursday, March 7, 2024 9:00 AM (1h 30m)

Research software development is crucial for scientific advancements, yet the sustainability and maintainability of such software pose significant challenges. In this tutorial, we present a comprehensive demonstration on leveraging software templates to establish best-practice implementations for research software, aiming to enhance its longevity and usability.

Our approach is grounded in the utilization of Cookiecutter, augmented with a fork-based modular Git strategy, and rigorously unit-tested methodologies. By harnessing the power of Cookiecutter, we streamline the creation process of research software, providing a standardized and efficient foundation. The fork-based modular Git approach enables flexibility in managing variations, facilitating collaborative development while maintaining version control and traceability.

Central to our methodology is the incorporation of unit testing, ensuring code integrity and reliability of the templates. Moreover, we employ Cruft, a tool tailored to combat the proliferation of boilerplate code, often referred to as the “boilerplate-monster.” By systematically managing and removing redundant code, Cruft significantly enhances the maintainability and comprehensibility of research software. This proactive approach mitigates the accumulation of technical debt and facilitates long-term maintenance.

The open-source templates are available at <https://codebase.helmholtz.cloud/hcdc/software-templates/>. In the first 30 minutes of the tutorial, participants will gain insights into the structured organization of these software templates, enabling them to understand the framework’s architecture and application to their own software products. The subsequent 30 minutes will be dedicated to a hands-on tutorial, allowing participants to engage directly with the templates, guiding them through the process of implementing and customizing them for their specific research software projects.

Maintaining research software presents distinct challenges compared to traditional software development. The diverse skill sets of researchers, time constraints, lack of standardized practices, and evolving requirements contribute to the complexity. Consequently, software often becomes obsolete, challenging to maintain, and prone to errors.

Through our tutorial, we address these challenges by advocating for the adoption of software templates. These templates encapsulate best practices, enforce coding standards, and promote consistent structures, significantly reducing the cognitive load on developers. By providing a well-defined starting point, researchers can focus more on advancing their scientific endeavors rather than grappling with software complexities.

Furthermore, the utilization of software templates fosters collaboration and knowledge sharing within research communities. It encourages the reuse of proven solutions, accelerates the onboarding process for new contributors, and facilitates better documentation practices. Ultimately, this approach leads to a more sustainable ecosystem for research software, fostering its evolution and ensuring its relevance over time.

In summary, our tutorial offers a practical and comprehensive guide to creating and utilizing software templates for research software development. By harnessing Cookiecutter with Git-based modularity, unit testing, and the power of Cruft, we aim to empower researchers in building robust, maintainable, and sustainable software, thereby advancing scientific progress in an efficient and impactful manner.

Slot length

Workshop (1h)

Primary author: SOMMER, Philipp Sebastian (Helmholtz-Zentrum Hereon)

Co-authors: SASS, Björn Lukas (Helmholtz-Zentrum Hereon); BENNINGHOFF, Markus (Helmholtz-Zentrum Hereon)

Presenter: SOMMER, Philipp Sebastian (Helmholtz-Zentrum Hereon)

Track Classification: Research Software (legacy): Open Source