2018 IEEE 14th International Conference on e-Science

Software Engineering and Digital Research Infrastructures

Carsten Thiel
CESSDA ERIC
Bergen, Norway
carsten.thiel@cessda.eu

Dieter Van Uytvanck
CLARIN ERIC
Utrecht, The Netherlands
dieter@clarin.eu

Tibor Kálmán
GWDG/DARIAH
Göttingen, Germany
tibor.kalman@gwdg.de

***Abstract*—Sustainable management of research software and the application of software engineering principles are one of the biggest challenges that the operation of digital Research Infrastructures face in the Social Sciences and Humanities community. We give an overview of our approach to apply state of the art industry standards for research software, and describe ongoing efforts towards a common set of guidelines and evaluation criteria for similar issues as EURISE Network.**

***Index Terms*—software quality, sustainability, developer guidelines, EURISE Network**

I. Introduction

Requirements and challenges arising in the operation of digital research infrastructures are to a large degree independent from the discipline they are built for, in contrast to infrastructures that include specialised physical equipment. Research software, an important pillar of such infrastructures, can be very heterogeneous, from plain scripts to highly sophisticated software suites, cf. [5]. Furthermore, as technology advances and research questions change, existing software is reused and applied to new problems, cf. [4]. This forms a fundamental challenge for infrastructure operators, whether centrally or distributed, who need to adapt and maintain these solutions.

To align technological recommendations and processes, the Social Sciences and Humanities (SSH) infrastructures CESSDA, CLARIN and DARIAH have been working together. A joint workshop was organised in Berlin in October 2017 that identified commonalities and differences in both organisational and technical areas.

This cooperation will continue under the umbrella EURISE Network [9], where research infrastructures meet research software engineers, to strengthen the combined foundations for future collaborations. Its goals are

- collaboration on technology and development requirements & solutions for research infrastructures,
- maintenance of a common set of guidelines,
- teaching and education on matters of software quality,
- support and engagement with the RSE community.

II. Technical Reference

The Technical Reference (TR) of the EURISE Network was started through a DARIAH project and has become the first common baseline of the three infrastructures [8]. It is a collection of best practices and guidelines for developers and maintainers of infrastructure components. It can be used to either gauge the quality of ongoing developments or, even better, as a starting point for new research projects, in particular when the components are planned to be attached to the infrastructures.

The TR is a reference, which lists general requirements and considerations, but it does not always specify choices. In particular, it does not define any technology stack, licences, or hosting services. These are part of an implementation of the TR in a research infrastructure or a data centre that is an explicit set of instructions tailored to the specific use case. This approach was chosen to allow an increased compatibility with existing (internal) requirements on specific choices and to reduce the effect of the 'not invented here' syndrome.

The initial work on guidelines and checklists was started by DARIAH and builds on existing work of CLARIAH [3] and DARIAH-DE. To address the existing common problems, the joint work on the TR will continue on both European as well as national levels. As a combined national SSH infrastructure, CLARIAH plans to extend their Software Quality Guidelines and DARIAH Germany is already involved with the TR.

III. Further ideas

The need to apply state of the art industry standards for software development, also to research software, is one of the core insights in recent years, [2], [6]. This includes, among others, the promotion of agile methods and the DevOps approach to infrastructure management, see [7]. Yet most research software is developed through grant based projects with limited lifespan. For an infrastructure to turn project demonstrators into sustained services, quality criteria must be applied. Standard reference criteria, which can be applied, already exist [1]. Furthermore, CESSDA has developed the Software Maturity Levels that form an abstract set of criteria with fine grained differentiation [10]. These can be used as baseline for the assessment of an individual products maturity, while an infrastructure or service provider can use it to define its requirements. Their inclusion into the common reference will strengthen its impact. Long-term maintenance of services also requires regular refactoring. If this was not taken up timely, the eventual costs to sustain a service increases dramatically.

978-1-5386-9156-4/18/$31.00 ©2018 IEEE
DOI 10.1109/eScience.2018.00071

Finally, it is a core mission to continously communicate with the community, as well as with the projects developing tools to be taken on by the infrastructure. The TR can serve as an initial reference to consider, but it is necessary to spread awareness for these topics and to even include them into curricula.

REFERENCES

[1] S. Buddenbohm, M. Matoni, S. Schmunk, C. Thiel, "Quality Assessment for the Sustainable Provision of Software Components and Digital Research Infrastructures for the Arts and Humanities" Bibliothek Forschung und Praxis, 41(2), pp. 231–241. 10.1515/bfp-2017-0024, 2017

[2] P. Doorn, P. Aerts, S. Lusher, "Research Software at the Heart of Discovery", DANS & NLeSC. Available at https://www.esciencecenter.nl/pdf/Software_Sustainability_DANS_NLeSC_2016.pdf, 2016 [Last Accessed Sept. 12, 2018]

[3] M. van Gompel, J. Noordzij, R. de Valk, A. Scharnhorst, "CLARIAH Software Quality Guidelines" version 1.1, https://github.com/CLARIAH/software-quality-guidelines, 2016 [Last Accessed Sept. 12, 2018]

[4] P. Harms, J. Grabowski, "Usability of Generic Software in e-Research Infrastructures", Journal of the Chicago Colloquium on Digital Humanities and Computer Science Volume 1 Number 3, 2011

[5] S. Hettrick "Research Software Sustainability: Report on a Knowledge Exchange Workshop". Available at https://www.esciencecenter.nl/pdf/Research_Software_Sustainability_Report_on_KE_Workshop_Feb_2016_FINAL.PDF, 2016 [Last Accessed Sept. 12, 2018]

[6] R. C. Jiménez, M. Kuzak, M. Alhamdoosh, et al. "Four simple recommendations to encourage best practices in research software" [version 1]. F1000Research, 6:876, 10.12688/f1000research.11407.1, 2016

[7] G. Kim, P. Debois, J. Willis and J. Humble "The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations" (1st ed.). IT Revolution Press, 2016

[8] C. Thiel, M. Rodzis, Y. Moranville, "EURISE Network Technical Reference", https://eurise-network.github.io/technical-reference/, 2018 [Last Accessed Sept. 12, 2018]

[9] C. Thiel, Van Uytvanck, Y. Moranville, "EURISE Network", https://eurise-network.github.io/, 2018 [Last Accessed Sept. 12, 2018]

[10] J. Shepherdson, Ø. Risnes, M. Priddy, M. Heinonen, J. Finn, W. Zenk-Möltgen, "CESSDA Software Maturity Levels (v1.0)" https://drive.google.com/file/d/0Bwk0RK5TDo6iTWxrTXFGSG5tMzg/view, 2017 [Last Accessed Sept. 12, 2018]