Painting the Picture of Software Impact with the Research Software Directory

Jurriaan H. Spaaks, Tom Klaver, Stefan Verhoeven, Jason Maassen, Tom Bakker,
Atze van der Ploeg, Ben van Werkhoven, Willem van Hage, Rob V. van Nieuwpoort
Netherlands eScience Center
Amsterdam, The Netherlands
{j.spaaks, t.klaver, s.verhoeven, j.maassen, t.bakker}@esciencecenter.nl
{a.vanderploeg, b.vanwerkhoven, w.vanhage, r.vannieuwpoort}@esciencecenter.nl

***Abstract*—In this lightning talk we will describe the Research Software Directory; a content management system that is tailored to research software with the goal of enabling a qualitative assessment of software impact and improving software findability.**

***Index Terms*—software impact, findability, research software engineering**

I. Introduction

The Netherlands eScience Center is the Dutch national center of excellence for the development and application of research software. As such, our employees contribute to scientific progress by writing software. Getting credited for such contributions is still not commonplace in many scientific domains [1]–[3]. Generally speaking, getting credited is not typically within the control of whomever is being cited. However, as an organization, we still need to show that we have a positive impact on science through the projects that we do. Therefore we started exploring alternative ways of demonstrating the impact of software in the aptly named IMPACT project [4], [5].

Building on previous work such as OSSMETER[1] and CROSSMINER[2], the IMPACT project collected a non-exhaustive list of software impact metrics. The list included metrics such as the number of downloads of a given software package, the number of bug reports, the number of persistent identifiers (e.g. DOIs) associated with a given software package, the number of registered users, and so on.

When we tried applying selected metrics however, we experienced a somewhat surprising problem. We found that it proved difficult to correctly outline the relevant collection of documents (source code, artifacts, documentation, etc.) to which you would like to apply software impact metrics. For example, even though most of our code is developed on just one platform (GitHub[3]), that platform is not necessarily the main outlet for users of the software, as Python code is typically installed via PyPI[4], R code via CRAN[5], Java via Maven[6], and JavaScript code via npm[7]. During the IMPACT project, we found that the link between the source code on GitHub and the corresponding item on such package management websites was often implicit, obstructing the automated collection of, for example, download statistics.

Secondly, we found that many software impact metrics are flawed in some way: for example, it is easy to get excited about publishing a package on, say, `npmjs.com` and watch it accumulate maybe 100 downloads within the first week or so, until you realize that only a few of these downloads represent humans interested in your code and the majority is triggered by mirrors and bots. As a result, even when numbers are available, interpreting them is difficult.

A third problem was that although we could potentially identify many metrics, but we did not know how to combine them into one index that would neatly summarize the software's impact.

Given these difficulties, we concluded that we needed to take a different route, and focus on providing a software impact assessment that is more qualitative in nature.

For this, we developed a software stack, collectively known as the Research Software Directory[8] [6]: think of it as a content management system that is tailored to software.

At the time of writing, the Research Software Directory combines data collection scripts that scrape sources like GitHub, Zotero[9] (our organization-wide reference manager), Zenodo[10] (which provides most of the persistent identifiers we use), CITATION.cff files for machine readable citation data[11] [7], our organization's blog on Medium[12], project descriptions from our corporate website, and more. Best of all, it requires only little manual input from our engineers, which they provide through a web form.

For each software package that we develop, we create a so-called 'product page' on the Research Software Directory. An example is shown in Figure I. Each product page includes a short description of the software and a *Mentions* section, which

[1]http://www.ossmeter.org/
[2]https://www.crossminer.org/
[3]https://github.com/
[4]https://pypi.org/
[5]https://cran.r-project.org/
[6]https://mvnrepository.com/repos/central
[7]https://www.npmjs.com/
[8]https://research-software.nl
[9]https://www.zotero.org
[10]https://zenodo.org
[11]https://github.com/citation-file-format/citation-file-format
[12]https://blog.esciencecenter.nl

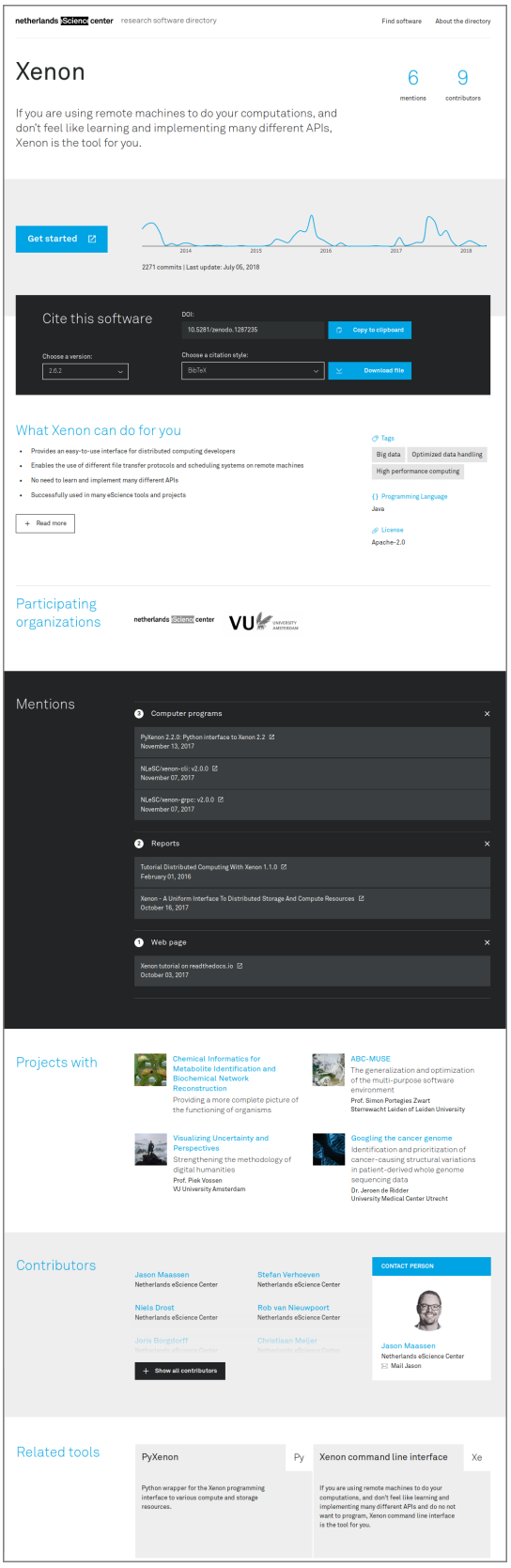

Fig. 1. An example product page for the *Xenon* software package.

we use to characterize the context in which the software exists. This context may include links to scientific papers, blog posts, demos, videos, tutorials, notebooks, etc., anything that helps visitors decide if the software could be useful for them. In addition, information is provided on which research projects use the software, which related tools exists, who the developers are, development activity, and, importantly, how the software should be cited.

By collecting all documents related to a software package in one place (i.e. the product page), an image starts to emerge of the impact of the software. The type of impact may be very different for different software packages. For example, one may have many scientific papers in one specific niche, while another may be featured in mainstream media such as tweets, blog posts, newspaper articles and so on, while yet another may have neither, but is instead being used as a dependency in many scientific projects. With the Research Software Directory, it is quite easy to distinguish between these three examples of impactful software, even without being able to put a number on it.

Besides enabling a qualitative assessment of software impact, the Research Software Directory improves the findability of software packages. This is partly because it provides metadata that helps search engines understand what the software is about. More importantly however, it provides humans with a clear view of the scientific and social context that each software package is used in. Together with the text snippets describing the goal of each software package, this information helps people to find the software that they need.

In this lightning talk, we will give a brief overview of the Research Software Directory, the features it currently offers, and our plans for further development of this tool with several partners from Dutch Research Institutes.

References

[1] Nick Barnes, David Jones, Peter Norvig, Cameron Neylon, Rufus Pollock, Joseph Jackson, Victoria Stodden, Peter Suber, "Science Code Manifesto", 2013, url: http://sciencecodemanifesto.org/, (visited July 2018).

[2] Olivier Philippe, Neil Chue Hong, Simon Hettrick, "Preliminary analysis of a survey of UK Research Software Engineers", in Proceedings of the fourth Workshop on Sustainable Software for Science: Practice and Experience (WSSSPE4), 2016

[3] Olivier Philippe, Martin Hammitzsch, Stephan Janosch, Anelda van der Walt, Ben van Werkhoven, Simon Hettrick, Daniel S. Katz, Katrin Leinweber, Sandra Gesing, and Stephan Druskat, "softwaresaved/international-survey: Public release for 2017 results", 2018, doi: 10.5281/zenodo.1194669

[4] P. Aerts, W. van Hage, D. Landman, P. Klint, J. Maassen, R. van Nieuwpoort, A. van der Ploeg, and J. J. Vinju, "A System for Impact Analysis of Academic Software Output", 2017

[5] W. R. van Hage, J. Maassen, and R. van Nieuwpoort, "Software impact measurement at the Netherlands eScience Center," in Proceedings of the fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4), 2016.

[6] J. H. Spaaks, J. Maassen, T. Klaver, S. Verhoeven, W. van Hage, L. Ridder, L. Kulik, T. Bakker, V. van Hees, L. Bogaardt, A. Mendrik, B. van Es, J. Attema, E. Rangelova, and R. van Nieuwpoort, "Research Software Directory". Zenodo, 05-Jul-2018. doi: 10.5281/zenodo.1154130

[7] S. Druskat, R Haines, and J. Baker, "Citation File Format (CFF) - Specifications". Zenodo, 07-May-2018. doi: 10.5281/zenodo.1003149