From Data to Action: Building Healthy and Sustainable Open Source Projects

Dawn Foster, CHAOSS

This article provides advice and resources for proactively using metrics to improve open source project health and sustainability before a crisis occurs to make software more sustainable and reliable for everyone.

Open source software has become ubiquitous and can be found in almost every codebase,[1] proprietary and open source alike, but sustaining open source projects and communities over the long term can be a challenge. Project leaders, maintainers, and contributors don't always have the time or experience to focus on sustainability. Using metrics is one way to help open source projects more quickly identify potential issues and areas where they can improve to make their projects more sustainable over the long term. Within the open source CHAOSS[a] project, metrics definitions and software exist to help people collect metrics for their open source projects, which has been described in more detail in previous articles. Goggins et al.[2] described how CHAOSS plays an integral role in the automation of key measures to make the state of open source readily observable using a CHAOSS tool called Augur[2] (see Figure 1). Gonzalez-Barahona et al.[3] took a slightly different approach by describing how people fitting several personas might use CHAOSS's GrimoireLab tools for data analysis of open source software.[3] Both of these articles are consistent with the approach that the CHAOSS project has taken in the past to provide tools and metrics to help gather data—but stopping short of providing advice about how to take action on the data and make improvements within open source projects. However, over the past couple of

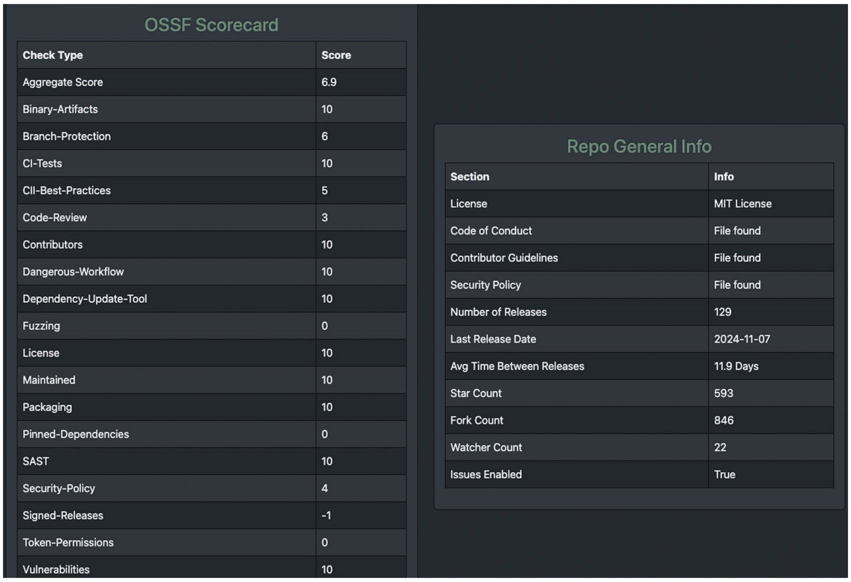
Digital Object Identifier 10.1109/MC.2025.3530556
Date of current version: 29 May 2025

[a] CHAOSS (Community Health Analytics for Open Source Software) is an open source project under The Linux Foundation https://chaoss.community/.

EDITOR DIRK RIEHLE
Friedrich Alexander-University of Erlangen Nürnberg;
dirk.riehle@fau.de

FROM THE EDITOR

Welcome back to the "Open Source" column! This month, Dawn Foster takes a look at open source community metrics. Anyone interested in setting up their open source project for community collaboration rather than commercial exploitation is well advised to dig into the metrics that work by Dawn and her colleagues at the CHAOSS project unearthed. Happy collaborating, everyone, and be healthy and happy!—*Dirk Riehle*



OSSF Scorecard

Check Type	Score
Aggregate Score	6.9
Binary-Artifacts	10
Branch-Protection	6
CI-Tests	10
CII-Best-Practices	5
Code-Review	3
Contributors	10
Dangerous-Workflow	10
Dependency-Update-Tool	10
Fuzzing	0
License	10
Maintained	10
Packaging	10
Pinned-Dependencies	0
SAST	10
Security-Policy	4
Signed-Releases	-1
Token-Permissions	0
Vulnerabilities	10

Repo General Info

Section	Info
License	MIT License
Code of Conduct	File found
Contributor Guidelines	File found
Security Policy	File found
Number of Releases	129
Last Release Date	2024-11-07
Avg Time Between Releases	11.9 Days
Star Count	593
Fork Count	846
Watcher Count	22
Issues Enabled	True

FIGURE 1. OSSF Scorecard security assessment and general information for a repository. OSSF: Open Source Security Foundation. (Source: Image generated using Augur.[e])

years, providing advice has gradually started to change at the CHAOSS project (see Figure 2).

The CHAOSS project has learned that not everyone has the experience or skills required to know how to interpret metrics and use those learnings to make improvements within an open source project and community. This is why the CHAOSS project began working on a series of MIT-licensed Practitioner Guides.[b] The goal of these guides and this article is to help practitioners, who may not be experts in data analysis or open source, understand how to interpret the data about an open source project and develop insights that can help to improve the health of that project. Open Source Program Offices, project leads, community managers, maintainers, and anyone who wants to better understand project health and take action on what can be learned from metrics will benefit from this article and the CHAOSS practitioner guides.

Measuring project health is complex with a complex array of aspects to consider.[4] One of the best places to start isn't actually with the metrics but by spending some time understanding the overall goals for a project in question[c] and talking to the people who participate in and maintain that project.[5] One reason that the CHAOSS project has avoided providing specific advice in the past is because there is no one-size-fits-all approach to using metrics to measure open source project health. Every open source project is a little different, and metrics should always be interpreted with the needs of that project and its context taken into account (Goggins et al.[2]). This is why it's important to look at trends in the data over time and think about whether other factors might be influencing those trends (for example, conferences, release timing, and vacation season). However, it is still possible to provide advice about certain topics that are common across open source projects, like contributor sustainability, responsiveness, organizational participation, and security.

CONTRIBUTOR SUSTAINABILITY

Many open source projects struggle to find enough people to sustain them.[6] If there are too few contributors and maintainers to sustain a project, the risk that the project will fail increases,[7] which creates a variety of challenges for the users and other projects that depend on that project. With respect to open source project sustainability, the relationship between contributors and maintainers is important to understand, and the Contributor Sustainability Practitioner Guide[d] helps in this regard. For example, bringing on new contributors increases the maintainer load because those maintainers will need to provide feedback on and merge contributions from

[b] https://chaoss.community/about-chaoss-practitioner-guides/.

[c] https://chaoss.community/practitioner-guide-introduction/.

[d] https://chaoss.community/practitioner-guide-contributor-sustainability/.

[e] https://github.com/chaoss/augur.

those new contributions. Promoting existing established contributors into maintainer roles to handle that increased load is key because projects require enough maintainers to handle incoming requests.[8] By focusing on recruiting and retaining contributors and subsequently promoting those contributors to maintainers, projects can help proactively prevent sustainability crises later. In this regard, there are several CHAOSS metrics that can help to understand the contributor, and related maintainer, sustainability of a project.

The CHAOSS project has learned that not everyone has the experience or skills required to know how to interpret metrics and use those learnings.

By starting with the Contributor Absence Factor metric, the risk to the project if one or more key contributors/maintainers decide to leave can be assessed while also better understanding which people are making the most contributions. The Contributors metric looks broadly at who contributes to a project to help understand how many contributors are active along with how many have increasing or decreasing activity over time. Because there are so many ways to contribute to an open source project, the Types of Contributions metric can help build a more holistic understanding of where and how people are contributing.

If it has been determined, via these metrics, that a project would benefit from improvements to contributor sustainability, there are a number of actions that can be taken. A good place to start is by looking for ways to reduce maintainer load through better contribution documentation. Projects may also benefit from taking a phased approach to recruiting new maintainers and reducing the scope that they will be responsible for (for example, a subproject or a portion of the codebase) and creating reviewer roles to help people build the skills they need as a maintainer while still allowing someone more experienced to oversee contributions before merging them. Maintainers can also use mentoring[9] and/or shadowing to more quickly teach people how to engage in maintainer work and help them learn to perform tasks that they'll need to do to become a maintainer. One reason to look at the Types of Contributions metric is that it can help to identify opportunities to promote people into maintainer roles to be responsible for activities that take up time from maintainers but that might be more effectively done by someone with more specialized expertise (for example, community management, marketing, and technical writing). Finally, having a written succession plan can also provide better sustainability if something happens to one or more of the existing maintainers.

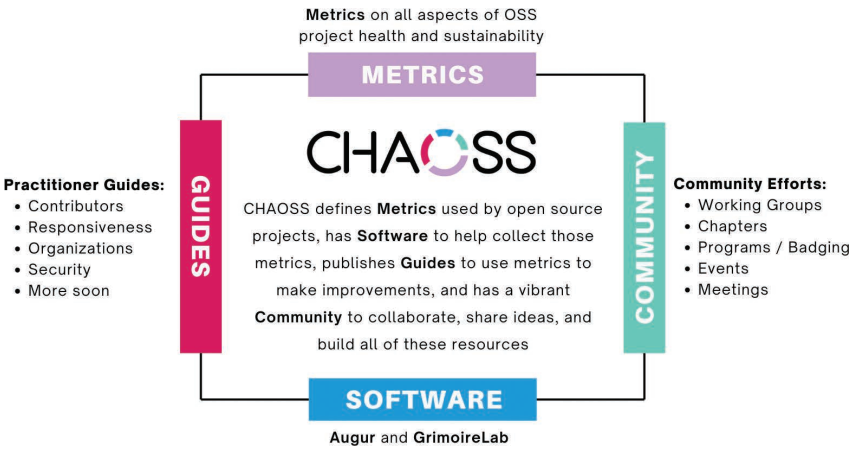

Metrics on all aspects of OSS project health and sustainability
METRICS
CHAOSS
CHAOSS defines Metrics used by open source projects, has Software to help collect those metrics, publishes Guides to use metrics to make improvements, and has a vibrant Community to collaborate, share ideas, and build all of these resources
GUIDES
Practitioner Guides:
• Contributors
• Responsiveness
• Organizations
• Security
• More soon
COMMUNITY
Community Efforts:
• Working Groups
• Chapters
• Programs / Badging
• Events
• Meetings
SOFTWARE
Augur and GrimoireLab

FIGURE 2. The CHAOSS community produces metrics, software, and guides to improve project health and sustainability.

RESPONSIVENESS

Responsiveness metrics[f] are an important part of assessing project health[8] since responsiveness is one of the most important factors in attracting newcomers[10] and retaining existing contributors to a project. New and existing contributors can become discouraged when they don't receive a timely and appropriate response to their contribution but can be encouraged when they get a quick and helpful resolution to their contribution. When projects are responsive, it can make people want to contribute more or continue contributing. Timely, thoughtful, and kind responses to contributors indicate that their work is appreciated.

By looking at Time to First Response, Time to Close, and Change Request Closure Ratio metrics together, a project can get a sense of whether contributors are getting a timely response and whether maintainers are keeping up with contributions by closing change requests (for example, pull requests/merge requests). For example, large numbers of open change requests can indicate that maintainers aren't particularly attentive to the project.[11] It can be tempting to put pressure on existing maintainers to respond more quickly,

[f] https://chaoss.community/practitioner-guide-responsiveness/.

but this rarely solves the long-term problem. It might result in short-term gains but can result in maintainer burnout if the underlying problems that are causing the lack of responsiveness are not resolved.

Like Contributor Sustainability, it can help to promote more contributors into maintainership roles so that more people can help respond, particularly into roles that free up time from code maintainers (for example, community management and documentation maintenance). Projects can also set clear expectations about when someone can expect a response, including delayed responses during busy times or holiday breaks. Using issue and pull request templates can further help people make better contributions the first time to reduce the reviewer load later.

ORGANIZATIONAL PARTICIPATION

Organizations can have a significant impact on the health and sustainability of an open source project,[g] especially when they come together under foundations to collaborate with other organizations.[12] On the one hand, organizations can help sustain open source projects by employing people to work on the open source projects that they use or by contributing other resources to those projects.[6] However, if all or most of the contributions are from the employees at a single organization, what happens when that organization is no longer willing or able to continue contributing at that same level?

From a metrics standpoint, a good starting point is looking at the Elephant Factor metric to determine how the work is distributed among multiple organizations along with the Organizational Diversity metric to look at which organizations are making contributions. Finally, it's also important to think about Organizational Influence metrics to understand which organizations have employees in leadership or other decision-making positions.

There are several CHAOSS metrics that can help to understand the contributor, and related maintainer, sustainability of a project.

If a need to improve organizational diversity has been identified, how to accomplish this depends on whether or not some of the contributors work for the dominant organization. If an organization is dominant, a good first step is to improve transparency and make sure that open source project work is being done in the open. It can also help to use professional connections to other organizations that are using the project and discuss ways for them to contribute. If another organization is dominant, make sure that contributions from others are welcome since, unfortunately, some organizationally dominated projects aren't particularly welcoming to contributions from outside of the leading organization. If contributions are welcome, other companies can dedicate time from employees to work within the project to provide more organizational diversity and act as a catalyst to show other organizations that their employees are welcome.

SECURITY

Open source software packages can be found in almost all software, so the security[h] of open source projects can have wide-reaching implications for other projects, their users, and the broader software ecosystem. Security is only as strong as its weakest link, so the security of any software component is only as good as the security of its dependencies.[13]

Security is a complex topic, but there are a few key metrics that can be used as a starting point. First, the OpenSSF Best Practices Badging criteria create a good engineering foundation that incorporates basic security practices. Second, using outdated dependencies results in projects that are four times as likely to have security issues,[14] so using the Libyears metric can help to understand if dependencies are kept up to date. Third, the Release Frequency metric helps gauge whether security fixes and other updates are incorporated in a release in a timely manner so that users can benefit from those security updates.

To improve the security of an open source project, securing the code repository and creating a detailed security policy document, often in a SECURITY.md file, is a solid place to start. Using automated tools (for example, Dependabot) can help keep dependencies as up to date as possible. On an ongoing basis, projects are likely to find or receive reports about security vulnerabilities that will need to be fixed, so those security fixes should be clearly documented and released in a timely fashion.

Finally, using some of the OpenSSF tools and resources can help find areas within a project where security practices could be improved. The OpenSSF Scorecard (see Figure 1) can help identify areas to improve, and working through the OpenSSF Best Practices Badge criteria is a good way to continue to make security improvements for open source projects.

The CHAOSS Practitioner Guides provided the inspiration for this article because contributor

[g] https://chaoss.community/practitioner-guide-organizational-participation/.

[h] https://chaoss.community/practitioner-guide-security/.

sustainability, responsiveness, organizational participation, and security are all key topics as open source projects work to improve sustainability. The guides are MIT licensed and can be used as-is, or they can be forked from the CHAOSS Data Science Working Group repository[i] and modified to meet other needs.

Building sustainable open source projects over the long term can be a challenge. Project leaders, maintainers, and contributors are busy people who don't always have the time to focus on growing a community along with maintaining their software. Using metrics is one way to help identify potential issues and areas where a project can be improved to make it more sustainable over the long term. Metrics are best used if they aren't used once and never again. By monitoring the data over time, projects can understand trends that might indicate areas for improvement as well as see if those improvements are having the desired effect.

Being proactive about improving sustainability before it becomes a crisis can help make open source software more sustainable and reliable for everyone, but this requires work. The CHAOSS project is addressing these issues now with metrics, software, guides, and community collaboration, but ongoing work is needed from all of us to maintain and build on these resources while also using these resources to make open source projects more sustainable over time. C

ACKNOWLEDGMENT

This work was funded by the Alfred P. Sloan Foundation (Grant 10384).

[i] https://github.com/chaoss/wg-data-science/tree/main/practitioner-guides.

REFERENCES

1. "Open source security and risk analysis report," Black Duck, Burlington, MA, USA, 2024. [Online]. Available: https://www.blackduck.com/resources/analyst-reports/open-source-security-risk-analysis.html
2. S. P. Goggins, M. Germonprez, and K. Lumbard, "Making open source project health transparent," *Computer*, vol. 54, no. 8, pp. 104–111, Aug. 2021, doi: 10.1109/MC.2021.3084015.
3. J. M. Gonzalez-Barahona, D. Izquierdo-Cortazar, and G. Robles, "Software development metrics with a purpose," *Computer*, vol. 55, no. 4, pp. 66–73, Apr. 2022, doi: 10.1109/MC.2022.3145680.
4. J. Linåker, E. Papatheocharous, and T. Olsson, "How to characterize the health of an Open Source Software project? A snowball literature review of an emerging practice," in *Proc. 18th Int. Symp. Open Collaboration*, 2022, pp. 1–12, doi: 10.1145/3555051.3555067.
5. A. Casari, J. Ferraioli, and J. Lovato, "Beyond the repository: Best practices for open source ecosystems researchers," *Queue*, vol. 21, no. 2, pp. 14–34, 2023, doi: 10.1145/3595879.
6. N. Eghbal, *Roads and Bridges: The Unseen Labor Behind Our Digital Infrastructure*. New York, NY, USA: Ford Foundation, 2016.
7. G. Avelino, E. Constantinou, M. T. Valente, and A. Serebrenik, "On the abandonment and survival of open source projects: An empirical investigation," in *Proc. ACM/IEEE Int. Symp. Empirical Softw. Eng. Meas. (ESEM)*, Piscataway, NJ, USA: IEEE Press, 2019, pp. 1–12, doi: 10.1109/ESEM.2019.8870181.
8. N. Eghbal, *Working in Public: The Making and Maintenance of Open Source Software*. San Francisco, CA, USA: Stripe Press, 2020.
9. F. Fagerholm, A. S. Guinea, J. Münch, and J. Borenstein, "The role of mentoring and project characteristics for onboarding in open source software projects," in *Proc. 8th ACM/IEEE Int. Symp. Empirical Softw. Eng. Meas.*, 2014, pp. 1–10, doi: 10.1145/2652524.2652540.
10. F. Fronchetti, I. Wiese, G. Pinto, and I. Steinmacher, "What attracts newcomers to onboard on OSS projects? TL;DR: Popularity," in *Proc. IFIP Int. Conf. Open Source Syst.*, 2019, pp. 91–103.
11. L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb, "Social coding in GitHub: Transparency and collaboration in an open software repository," in *Proc. ACM 2012 Conf. Comput. Supported Cooperative Work*, 2012, pp. 1277–1286.
12. D. Riehle, "The innovations of open source," *Computer*, vol. 52, no. 4, pp. 59–63, Apr. 2019, doi: 10.1109/MC.2019.2898163.
13. N. Imtiaz, A. Khanom, and L. Williams, "Open or sneaky? Fast or slow? Light or heavy?: Investigating security releases of open source packages," *IEEE Trans. Softw. Eng.*, vol. 49, no. 4, pp. 1540–1560, Apr. 2023, doi: 10.1109/TSE.2022.3181010.
14. J. Cox, E. Bouwers, M. Van Eekelen, and J. Visser, "Measuring dependency freshness in software systems," in *Proc. IEEE/ACM 37th IEEE Int. Conf. Softw. Eng.*, vol. 2, Piscataway, NJ, USA: IEEE Press, 2015, pp. 109–118, doi: 10.1109/ICSE.2015.140.

DAWN FOSTER is the director of data science at CHAOSS, London, U.K. Contact her at dawn@dawnfoster.com.