RI.
SE

JOHAN LINÅKER

How to Identify and Avoid Cracks and Bumps in your Digital Infrastructure?

-> By Considering the Health of Open Source in your Intake Process

Open Source Software Health

- An Open Source Software project's capability to stay viable and maintained over time without interruption or weakening

Photo by Jan Piatkowski | https://unsplash.com/photos/eopz9bkwROs

Open Source Software Health

- Productivity: There is an active development of the project
- Robustness: The development is open and spread out on several (independent) individuals
- Openness: Users of the project can influence and contribute to the development of the project

Photo by Markus Frieauff | https://unsplash.com/photos/IJ0KiXl4uys

Photo by Modestas Urbonas | https://unsplash.com/photos/vj_9l20fzj0

Open Source Software and our Digital Infrastructure

- Open Source Software makes up a vitale building block in our digital infrastructure
- Needs maintenance as with physical infrastructure to stay secure and robust

RI.
SE

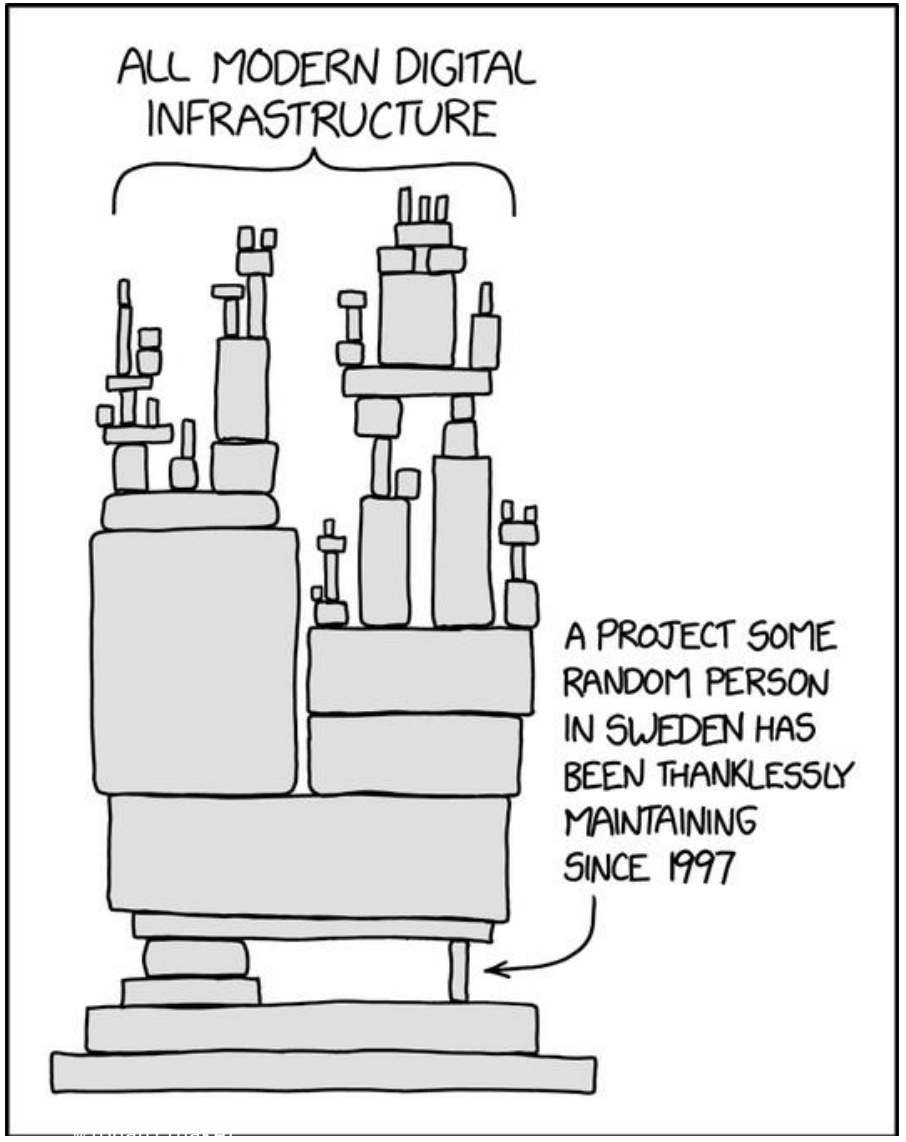


Photo by @tsjost and XKCD | https://twitter.com/tsjost/status/1295662118113619968

Open Source Software and our Digital Infrastructure

- Open Source Software makes up a vitale building block in our digital infrastructure
- Needs maintenance as with physical infrastructure to stay secure and robust

RI.
SE

Linus' law

- ”Given enough eyeballs, all bugs are shallow”
- Requires that enough eyeballs actually reaches the codebase
- Free-riding, for both good and bad

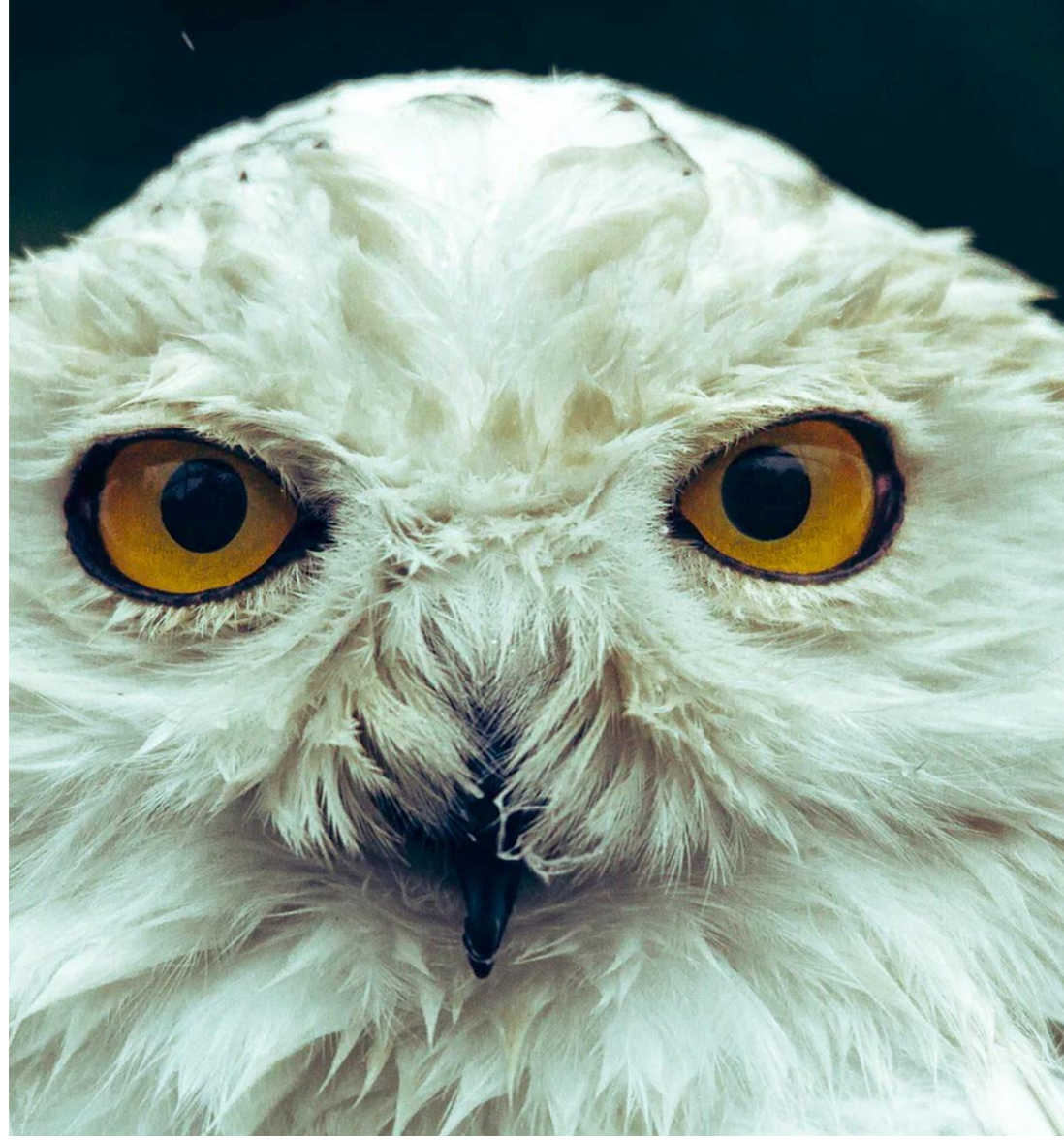


Photo by Frida Bredesen | https://unsplash.com/photos/c1fFv08N7qE

The Tragedy of the commons

- Commonly exemplified through Hardin's open pastures (Hardin, 1968)
- May be considered as a Common Pool Resource (CPR)
- A resource system that is non-exclusive, and subtractable (Ostrom, 1990)

Photo by Martin Schmidli | https://unsplash.com/photos/qg8ihPPQkKo

Brain-time as a Common Pool Resource

- “Brain-time” and maintenance effort is subtractable
- Maintainers are humans, not robots
  - Burnout, changed family or working conditions
- Companies must adapt to stay competitive
  - Refactorization, new products, changed business model

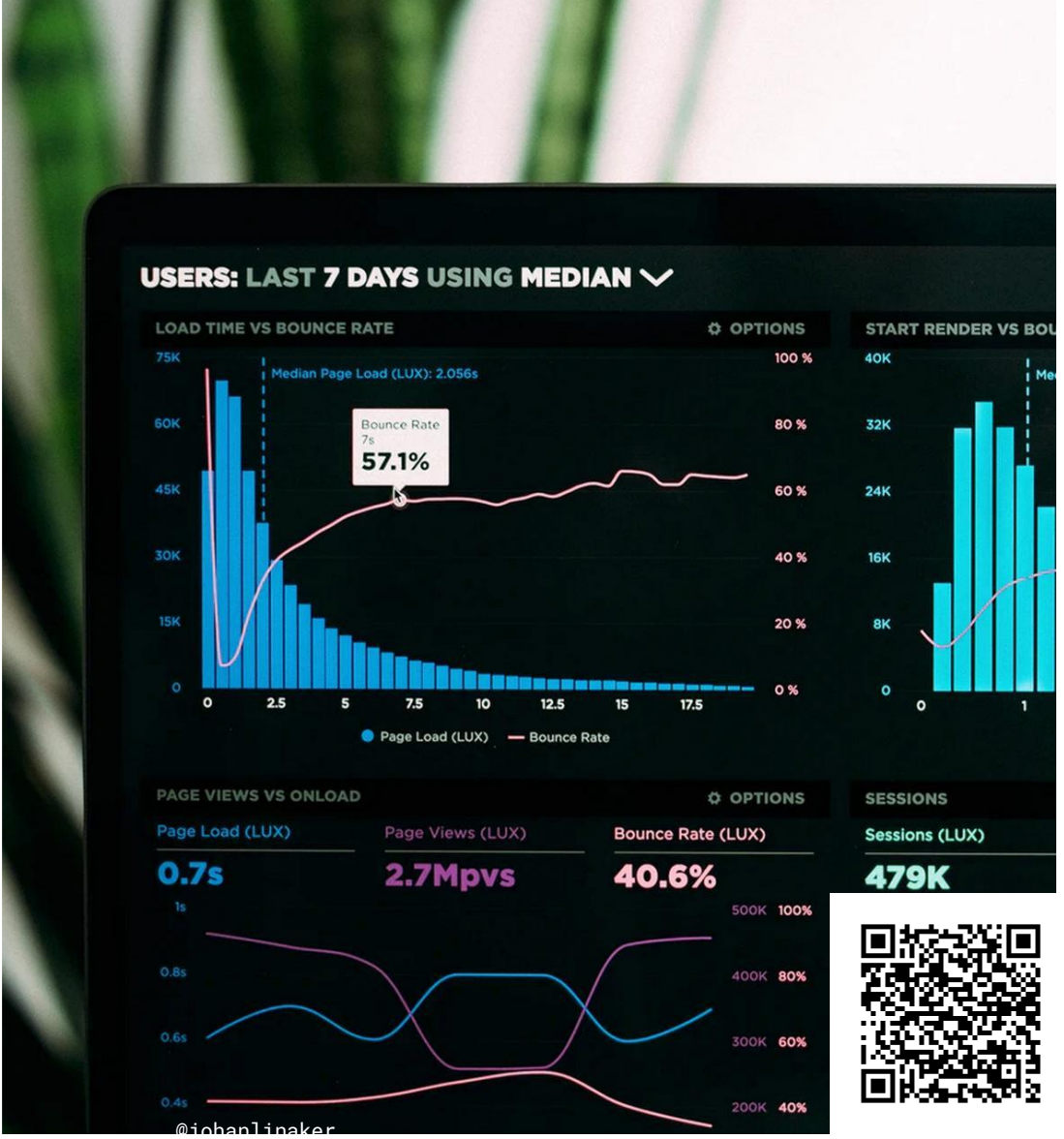
Photo by NeONBRAND | https://unsplash.com/photos/KYxXMTpTzek

How can we find the cracks and bumps before they appear?

How can we avoid them?

How can we mitigate them?

RI.
SE



By Considering the Health of Open Source in your Intake Process

- First part of a longer design science research project*
- Goals:
  - Enable health analysis at intake and acquisition of OSS, and ongoing consumption
  - Enable sourcing decisions and proactive health improving measures

* https://bit.ly/3AM5NR8

RI.SE

Photo by Luke Chesser | https://unsplash.com/photos/JKUTrJ4vK00

Photo by Jaredd Craig | https://unsplash.com/photos/HH4WBGNyltc

What can we find in literature?

- 146 studies
- 107 characteristics (+associated metrics
- Divided over 15 themes
- Supplementary material: https://doi.org/10.6084/m9.figshare.20137175
- Paper: https://www.ri.se/sites/default/files/2022-09/opensym2022-6%20%281%29.pdf

RI.SE

Framework structure

- Level of abstraction
  - Network-level
    Characteristics related to the Overarching software ecosystem or network that the OSS project is part
  - Project-level
- Socio-technical dimension
  - Actors
    Human and Community-related characteristics
  - Software
    Technical and project-related characteristics
  - Orchestration
    Governance-related characteristics

RI.SE

Actor-related characteristics (1/4)

- Communication (4)
  - *how productive an OSS project is in planning and discussing the evolution and development of its technical and non-technical deliverables*
  - E.g., response time & quality, social activity, visibility
- Culture (6)
  - *how able a community is to facilitate a positive and inclusive collaboration and dialogue among existing and potential actors*
  - E.g., conflicts, openness, sentiment, recognition

Photo by Ryoji Iwata | https://unsplash.com/photos/IBaVuZsJJTo

Actor-related characteristics (2/4)

- Diversity (5)
  - how able a community is to accommodate and attract a diverse community of actors, while enabling existing and new use cases of the OSS project
  - E.g., application, demographic, & organization diversity, technical knowledge
- Finance (2)
  - *how financially viable actors are in an OSS community in terms of being able to dedicate their time and resources to the long-term maintenance of the OSS project*
  - E.g., financial stability & support

Photo by Ryoji Iwata | https://unsplash.com/photos/IBaVuZsJJTo

Actor-related characteristics (3/4)

- Popularity (5)
  - *how popular and well-adopted an OSS project is among existing and potential end-users and contributors*
  - E.g., competing projects, external interest, adoption
- Stability (12)
  - *how capable the OSS project is in terms of preserving a critical population of actors with the capability to maintain the OSS project long-term*
  - E.g., predicted evolution, knowledge concentration, retention

Photo by Ryoji Iwata | https://unsplash.com/photos/IBaVuZsJJTo

Actor-related characteristics (4/4)

- Technical activity (5)
  - *how productive an OSS project is in evolving and developing its technical and non-technical deliverables*
  - E.g., development activity (community, maintainers, overall)

Photo by Ryoji Iwata | https://unsplash.com/photos/IBaVuZsJJTo

Software-related characteristics (1/4)

- Development process (7)
  - *how capable a community is in terms of its development process to maintain the OSS project to a high quality long-term*
  - E.g., contribution process, onboarding, quality assurance
- Documentation (6)
  - *how capable a community is to develop, persist, and disseminate knowledge among current and future actors engaged in the project*
  - E.g., completeness, complexity, currentness

RI. SE

Photo by Markus Spiske| https://unsplash.com/photos/1LLh8k2_YFk

Software-related characteristics (2/4)

- General characteristics (7)
  - *how attractive an OSS project is based on its general technical features*
  - E.g., application domain, project complexity, type of technologies
- License (4)
  - *how license choices and related practices may affect the popularity and attractiveness of an OSS project, both for commercial actors and individuals*
  - E.g., flexibility, implications, jargon, management

RI. SE

Photo by Markus Spiske| https://unsplash.com/photos/1LLh8k2_YFk

Software-related characteristics (3/4)

- Scaffolding (10)
  - *how robust and accessible the development and communication infrastructure used in the OSS project is in terms of enabling a collaborative and high quality maintenance of the project*
  - E.g., build environment, infrastructure availability and accessibility
- Security (10)
  - *how robust an OSS project is in terms of mitigating and managing vulnerabilities and security-related aspects in the current and future maintenance of the project*
  - E.g., vulnerability persistence & presence, dependency, management, security practices

RI. SE

Photo by Markus Spiske| https://unsplash.com/photos/1LLh8k2_YFk

Software-related characteristics (4/4)

- Technical quality (10)
  - *how robust an OSS project is in terms of its technical quality, considering both a user and developer perspective*
  - E.g., source code quality & complexity, maintainability, product quality

RI. SE

Photo by Markus Spiske| https://unsplash.com/photos/1LLh8k2_YFk

Orchestration-related characteristics

- Orchestration (9)
  - *how mature and open the orchestration is in the OSS project or its overarching ecosystem in terms of enabling an open and inclusive collaboration and long-term maintenance of the OSS project*
  - E.g., community and ecosystem structure, governance, processes

Photo by Sivani B | https://unsplash.com/photos/bczrpU9n8f4

Limitations and threats to validity

- Not a systematic overview of literature
- Limited coverage of the network-level
- Did we really characterize the health of OSS projects?

Extant work

- Community Health Analytics for Open Source Software (CHAOSS)
  - Framework with metrics for health analysis and assessments
- Open Software Security Foundation (OpenSSF)
  - Industry foundation focused on raising security of critical OSS
- SustainOSS
  - Community focused on sustainability and health topics

RI.
SE

Adapting the intake process

- Pre-trial on a large international software-producing organization
- Process initiated, owned and managed by enterprise architects
- Objective:
  - Lower risk of OSS used and considered in the intake process
- Goals:
  - Decentralized, self-managed process
  - Enable but don’t overburden developers
  - Enable follow-up and actionable insights

RI.
SE

Design approach

- Questionnaire developed through iterations based on CHAOSS metrics
- Main concerns and risks, as well as types of OSS projects identified through group discussions and interviews
- Observed developers as they walked through the questionnaire
- Went from 2h to <15 min evaluation
- Considered to raise awareness and decrease overall risk in the intake process from both engineers and process owners

RI.SE

Things to consider (1/3)

- Interview and map up main concerns from internal stakeholders
- Consider types of projects used and need for tailoring
- Lightweight questionnaire/checklist designed for developers
- Needs simple answers (Yes/No) or clear categories (1-5, 6-10…)

Photo by Patrick Perkins | https://unsplash.com/photos/ETRPjvb0KM0

Things to consider (2/3)

- Must be easy to find and process the data needed
- Remove thresholds and be inclusive
- Automate where possible
- Add screening functionality in intake and build pipelines to flag projects of concern
- Provide support for yellow and red flags

Photo by Patrick Perkins | https://unsplash.com/photos/ETRPjvb0KM0

Things to consider (3/3)

- There’s no one model or number to measure the health of OSS projects
- Different characteristics can help guide you in painting the picture and how to view it

Photo by RhondaK Native Florida Folk Artist | https://unsplash.com/photos/_Yc7OtfFn-0

Sourcing and acquisition

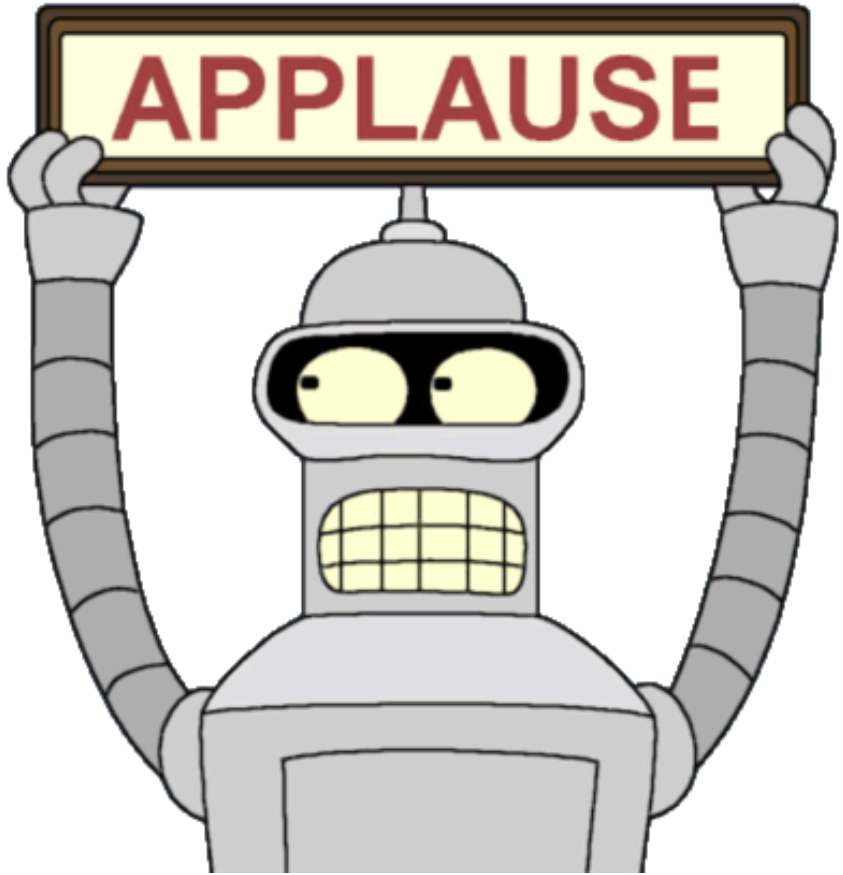
- Pre-trial at large Swedish national agency
- Workshop format with internal stakeholders
- Goal was to evaluate health of to OSS e-archival solutions
- Questionnaire developed through iterations based on CHAOSS metrics
- Enable comparison between open and closed alternatives in an acquisition
- Evaluation needs to be thorough and detailed

RI.
SE

Future work

- Validate, prioritize and further characterize characteristics through an interview survey with a general and a case-specific sample (ongoing)
- Gather metrics and data sources from practitioners through observations, and contrast to what we've found in literature
- Investigate applicability and possibility to automate and quantify characteristics with industry partner (ongoing)

Photo by Tomasz Frankowski | https://unsplash.com/photos/kBUfvkbFIoE


APPLAUSE