

The background features a grid of various colored doors and windows in shades of white, grey, brown, and yellow. A blue orbital path with a yellow arrowhead curves across the scene from the top left towards the right. A semi-transparent orange rectangular box is positioned in the lower-left area, containing text.

**RI
SE**

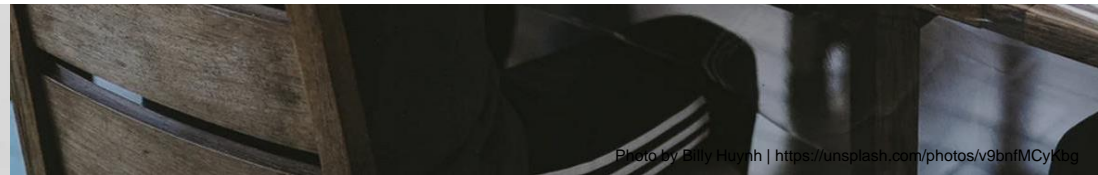
JOHAN LINÅKER

Open Source Strategy and Risk Management

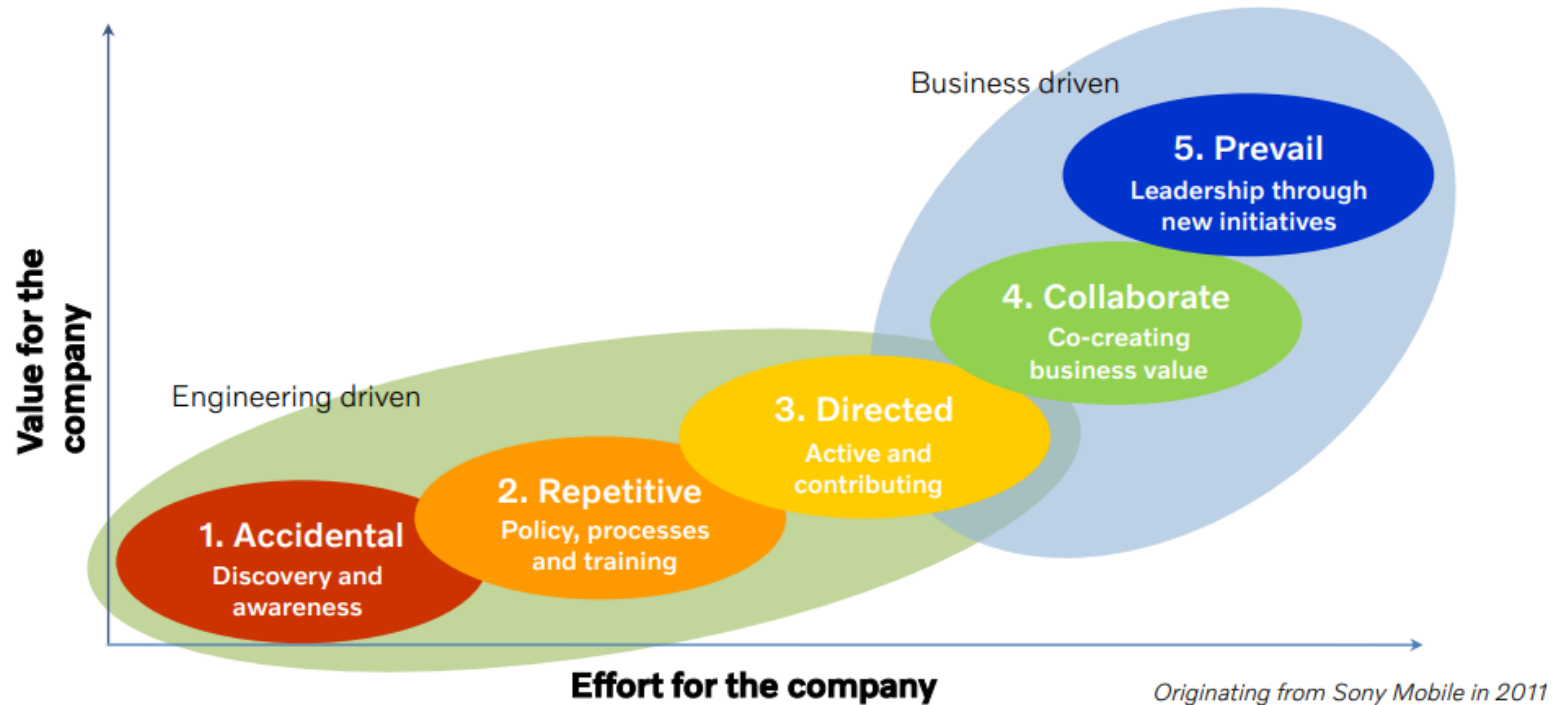
A birds-eye view



Liberaly Licensed, Collaboratively Developed Software



Maturing from consumption to leadership



Open Source is about Opening up

- Focus on openness
 - Whomever can contribute
 - Influence through merit
 - Self-appointment of tasks
 - Transparency in work
- Often implying new ways of working
- Requires organizational and cultural change



Open Source Program Offices (OSPOs)

- Center of competency and support
- Drives organizational readiness and maturity forward on open source
- Designs and executes an organization's overarching open source strategy
- Provides voice of reason and objectivity on the benefits, risks, and costs of open source and how to balance between
- Supports use, development, and collaboration on open source

<https://linaker.se>

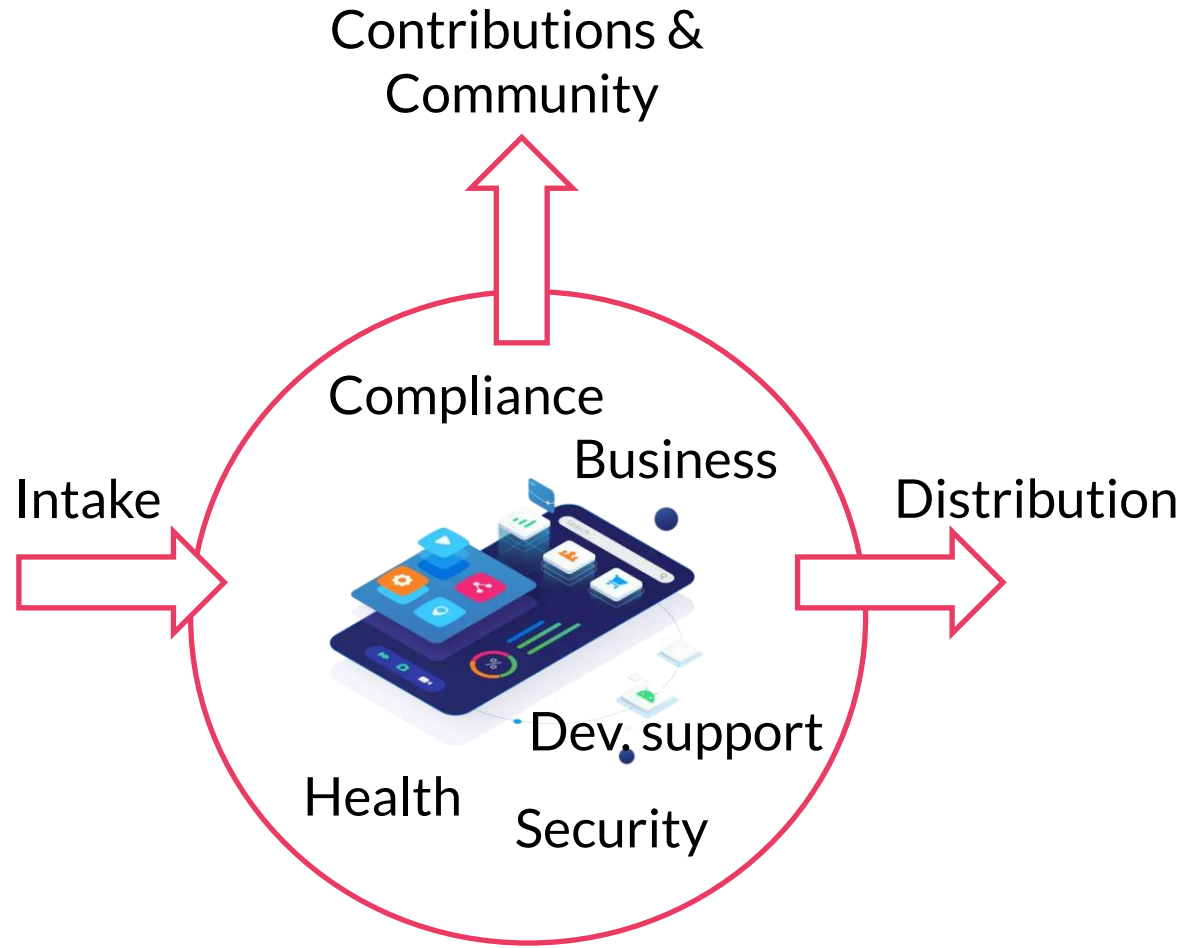


Photo by Quino AI | <https://unsplash.com/photos/dhGFLj3r10Q>

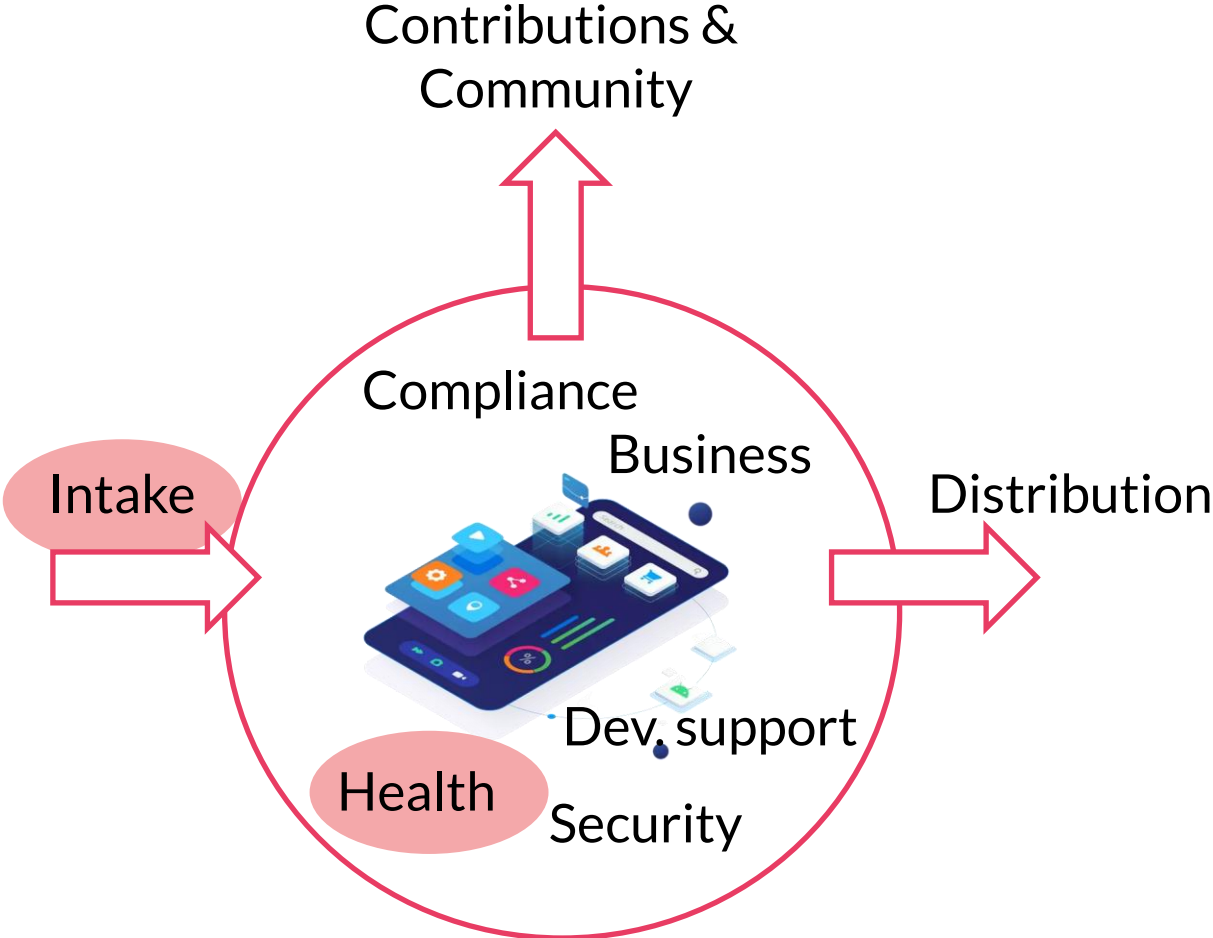
A balancing act between Value and Risk



Open Source Management



Health analysis of OSS intake



Paper: <https://www.linaker.se/blog/health-check-ups-on-open-source-software-projects/>

Open Source Software Health

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

<https://linaker.se>



Photo by Jan Piatkowski | <https://unsplash.com/photos/eopz9bkwROS>



- An MD asks questions and uses tools at disposal to examine the patient, identify symptoms, arrive at a diagnosis, and prescribe a treatment.



- A developer asks questions and uses tools at disposal to examine the OSS project, identify symptoms, arrive at a sourcing decision, and potential actions for community engagement.

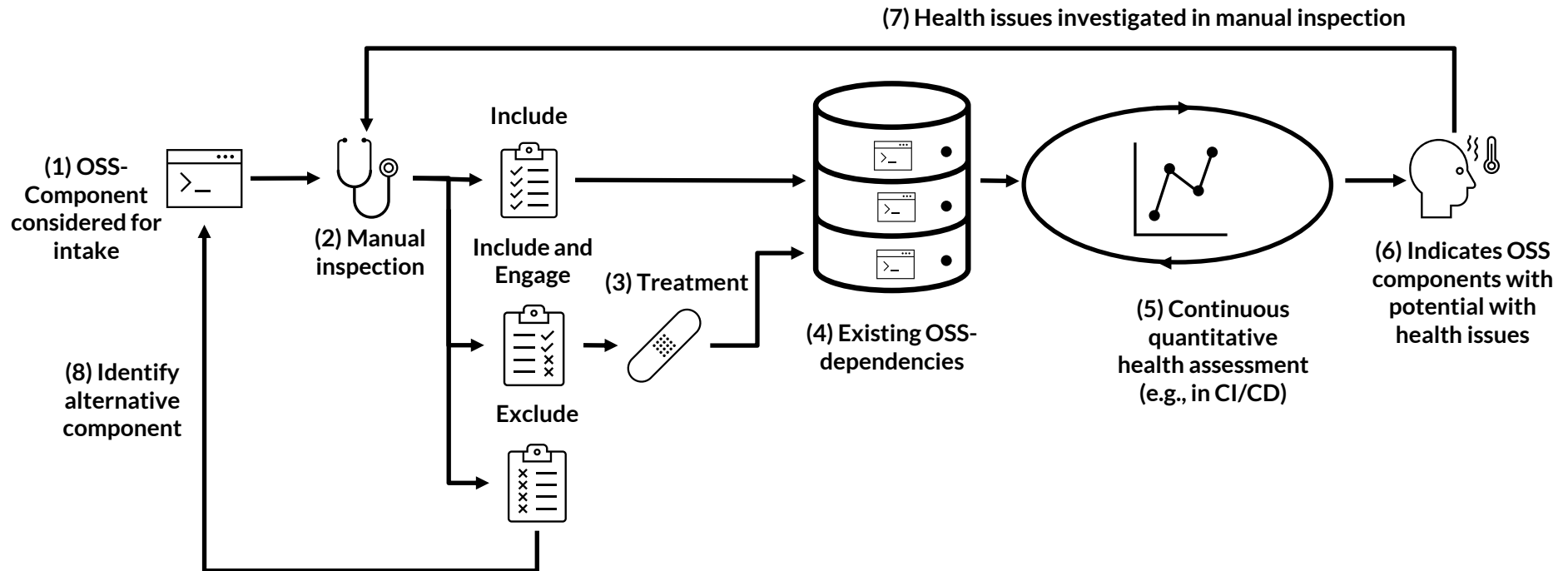
Open Source Software Health

- Health is not a number, not binary
- 100+ health aspects to consider across project and ecosystem levels, and
 - Community productivity, and stability
 - Orchestration
 - Production process and outputs
- Important not to compare apples and oranges. Need to consider (e.g.,)
 - Life-cycle stage,
 - Project Complexity
 - Governance concentration,
 - Strategic Importance

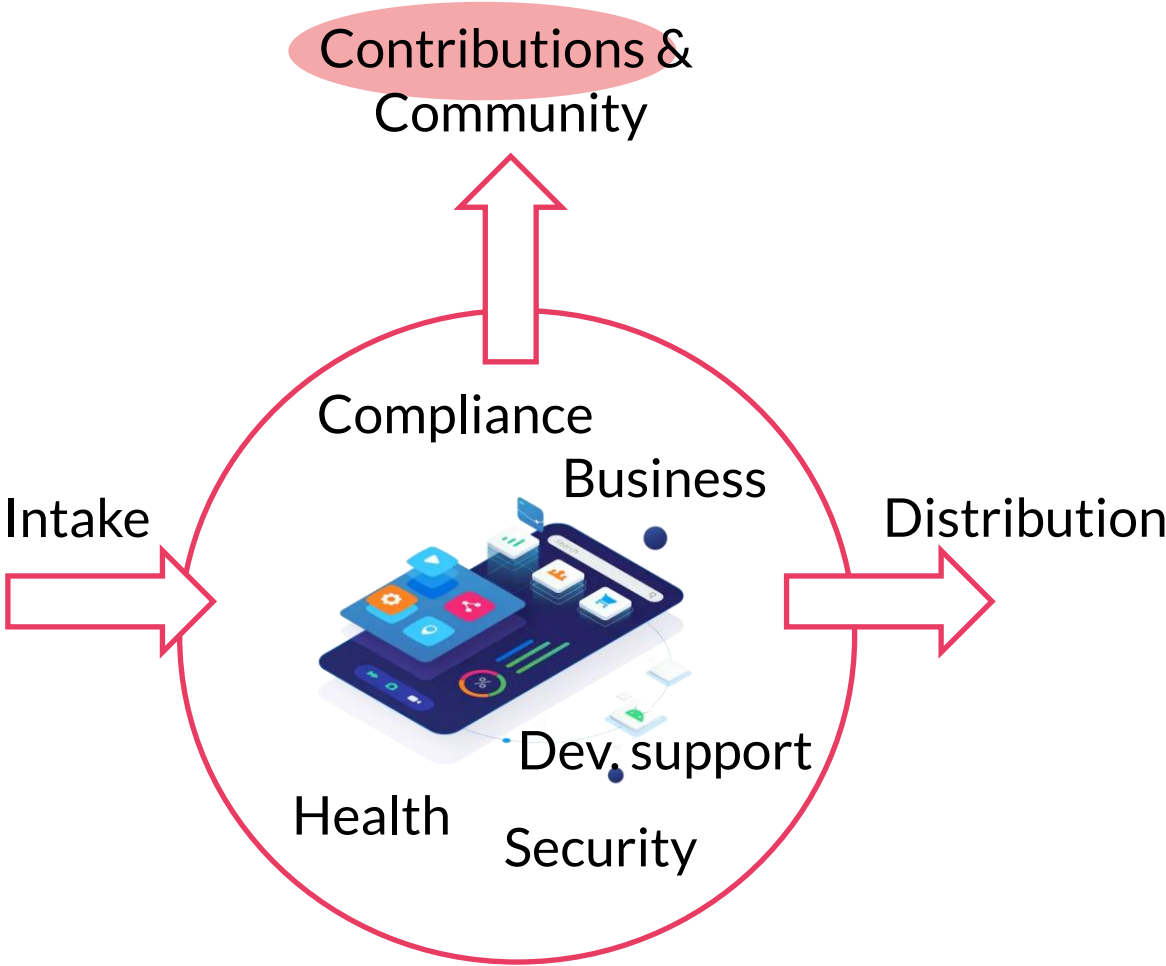
<https://linaker.se>



Semi-automating the health-check process



Contribution strategies



Paper: <https://doi.org/10.1007/s10664-020-09855-2>

Why need a contribution strategy?

- Answers if, why, when, and how something should be open sourced
- Help to execute the overarching open source strategy
- Promotes and support contributions in line with business objectives
- Highlights potential costs and risks upfront
- Enables the balancing between objectives and complexities



Enable critical assessments

- Support individuals posting contribution request to make their case
- Support decision makers to make informed decisions
- Reduce doubts and make contribution process more efficient
- Enable a proactive planning on contributions



Provide clarity on process

- Lower barrier and remove uncertainties
- Decision options
 - Reject, conditional accept, or accept
- Contribution options
 - Existing community, new community in specific external entity, or new community “in the wild”
- Needs to be weighed against potential objectives and complexities



**Weights will
differ based on
objectives and
context**



Contribution Objectives

- Reputation-centric objectives
 - Prove skill and influence
 - Be a good open source citizen
 - Improve employer branding
 - Increase transparency
- Supplier-centric objectives
 - Create price pressure
 - Outsource infrastructure operation

<https://linaker.se>



Photo by Alexander Grey | <https://unsplash.com/photos/dW6dFB0HUu4>



Contribution Objectives

- Strategy-centric objectives
 - Enable data sharing, and gathering
 - Standardize a solution
 - Build a software ecosystem
 - Improve partner collaboration
- Engineering-centric objectives
 - Open up innovation process
 - Extend development resources

Contribution Complexities

- Control-centric complexities
 - Impact on value proposition
 - Impact on internal operations
- IPR-centric complexities
 - Differentiating functionality
 - Commoditization
 - Sensitive IPRs
 - Substitutes
 - License compliance

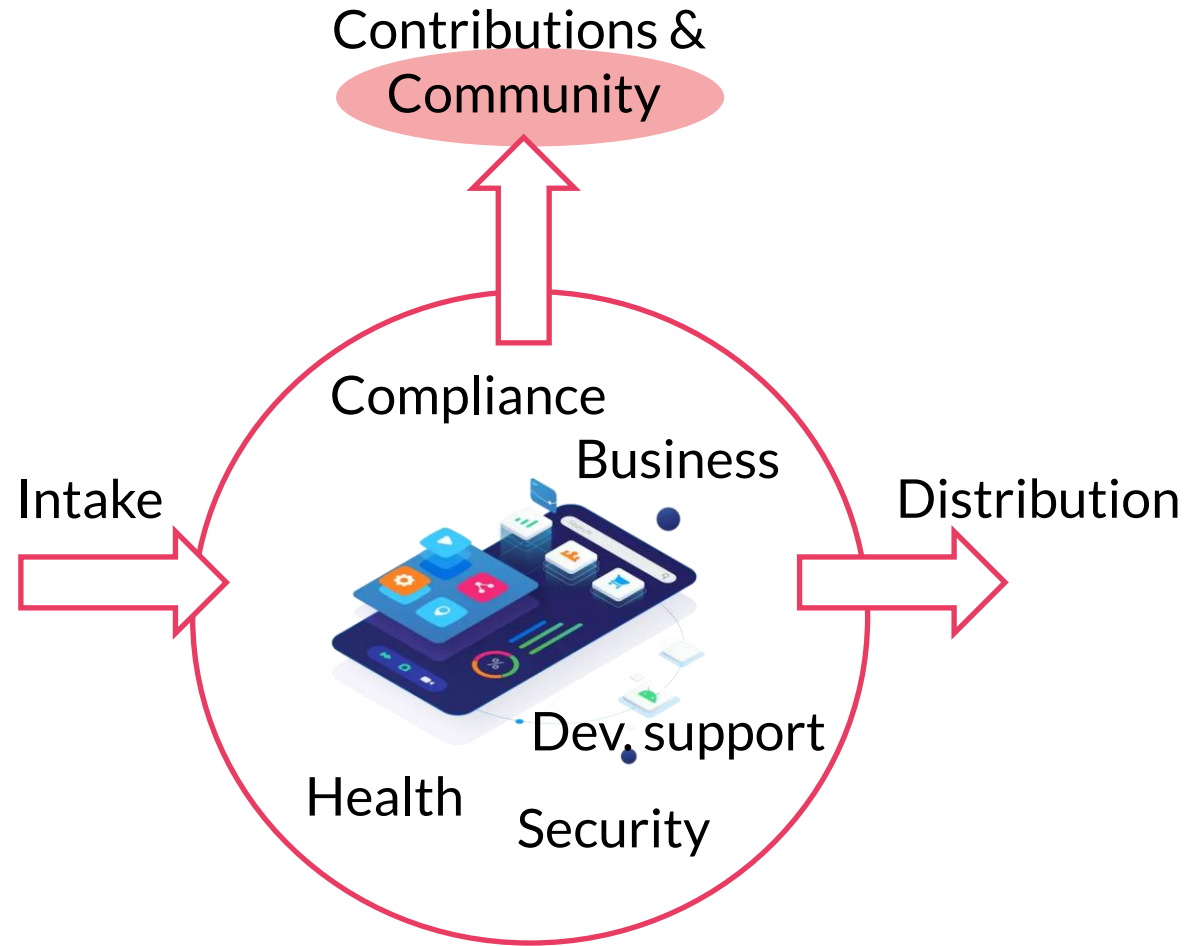




Contribution Complexities

- Exposure-centric complexities
 - Ethical use
 - Security threats
- Cost-centric complexities
 - Budget and resource constraints
 - Modularity and architecture
 - Code alignment
- Community-centric complexities
 - External interest
 - Influence in community
 - Community health

Community strategies



Paper: <https://arxiv.org/pdf/2208.03302>

Why need a community strategy?

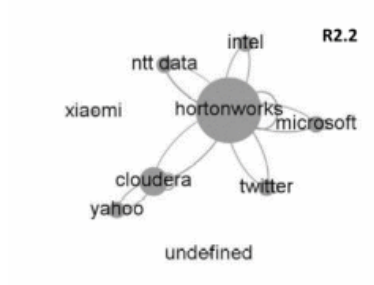
- Answers if, why, when, and how you should engage in a community
- Sets the agenda and goals for any engagement
- Help to execute the overarching open source strategy
- Promotes and support community engagement in line with business objectives and identified risks



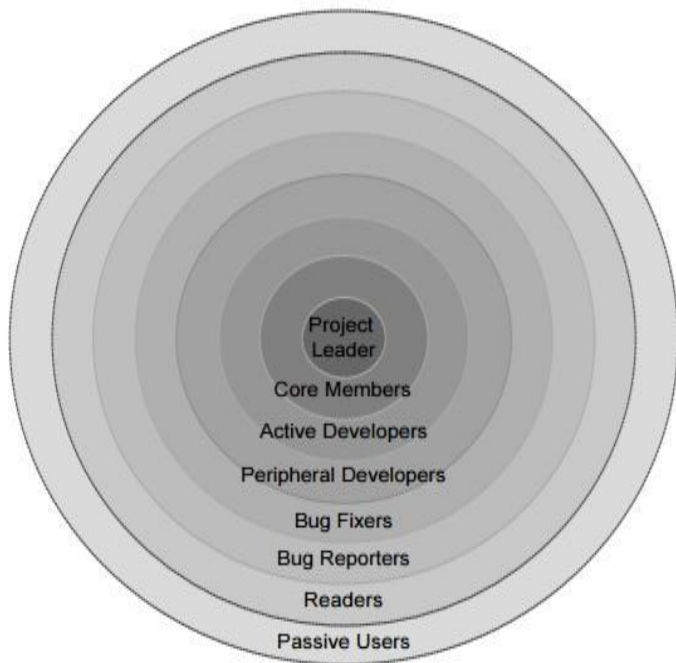
**Use your weights
to prioritize your
engagements**



Understand the community



Understand the governance



@<https://linaker.se>



Photo by Leon | <https://unsplash.com/photos/Oalh2MojUuk>

Adapt relationship with community

- Symbiotic
 - Win-win for both firm and community
 - Contributing to influence projects according to internal agenda and improve health to mitigate security risks
- Commensalistic
 - Gain for firm, community indifferent
 - Use project and doing lighter contributions. Project in line with internal agenda and healthy with others already supporting it.
- Parasitic
 - Firm free-riding on community.
 - Using as is not giving anything back. Worst case expecting free work for nothing in return. Looked down on from communities.

<https://linaker.se>



Engage accordingly

- Development of new functionality and bug fixes
- Requirements identification, analysis, and prioritization
- Testing and quality assurance
- Documentation, marketing and community management
- Financial and infrastructural support
- ...



