



SHARING & REUSE CONFERENCE

OPEN.SHARE.LINK

MAHA SHAIKH

Senior lecturer of Digital Innovation at King's College London

**How the Public/Private Sector Organizations
Help Sustain Open Source Communities**

Follow discussions online  @EU_isa2
#SRCONF19

Why should any organization worry
about sustaining an open source
community?

"Here's a technology. How powerful is the community that's using this technology? How stable is that community? Do I want to invest my business in it?" (Red Hat CEO, James Whitehurst, 2013).

Need to *invest* in open source communities

Questions that need an answer:

- Which community would be best to invest in?
- How do we judge the worth of a community?
- How do we ensure the health of a community – i.e. the sustainable, long-term existence of membership and high quality software production?

What is a healthy community?

A viable, organizational-friendly community that consistently produces high-quality, innovative products and services

Evaluating a Community's Product and Services

Characteristics	Detailed perspective
Product features	- Fit with corporate needs
Software quality	- Community size as proxy for testing (many developers 'eyeballing' the code) - Number of downloads by users (indirect measure of software approval) - Number and variety of email threads interrogating issues with the code
Documentation availability	- Availability of wiki HOW TO pages - Accessible FAQ pages - Level of detail of documentation - Speed of updating documentation
Corporate-friendly license type	- Reciprocity level demanded by the license - Possibility to dual license the software
Track record of community-based support	- Average velocity of Q&A - Average time for issue resolution
3 rd party support costs	- Costs of contracting with 3rd parties for support services (only available for popular open source products)
Internal support costs	- Personnel cost of providing support services - Costs of maintaining multiple versions and reintegrating repeatedly

Evaluating a Community's Viability

Characteristics	Detailed perspective
Vibrancy of the developer base	- Number of active contributors - Growth of active contributors - Renewal of the core contributor group - Turnover of participants
Growth of the code base	- Lines of code - Number of subsystems
Attention paid to software quality improvements	- Number of bug reports - Number of upgrade patches made available - Number of testers (members eyeballing code) - Number of responses to questions

Evaluating a Community's Friendliness to Corporations

Characteristics	Detailed perspective
Community's orientation towards corporate engagements	- Evidence that past decisions were based on ideological grounds rather than pragmatic ones - Number of core developers who dropped out in protest against corporate involvement in the community - Stated attitudes of community leaders towards corporate involvement - The degree of community leaders' ideological influence over others (e.g., their ability to overcome resistance if they are supportive of corporations)
Accessible communication processes	- Access to developer mailing lists - Consistency with which the community maintains online FAQs and transparent documentation through wikis, blogs, and community sites - Direct access to specific developers
Clear governance structures within the community	- Clear and visible order of trusted maintainers of code versions - Clear delineation of responsibilities among developers for specific modules and tasks - Ability to discern leaders
Willingness to accommodate corporate interests	- Searchable history of past community decisions on product versions (related to any company) - Revealed desire by community to tap into a company's customer base for testers and users - Community's willingness to commit to agreed-upon release dates

Evaluating an Ecosystem’s Health

Characteristics	Detailed perspective
Strength of ecosystem partners	- Number of partners - Types of partners - Reputation of partners - Degree of commitment to the ecosystem
Level of support by partners	- Code and hardware donated - Paid employees working on the project
Commercial acceptance of the chosen license regime	- The use of the same license regime used for distributing the core code in distributing products developed by ecosystem partners - Number of dual licensing schemes set up by partners (presumably to avoid using the license regime of the core code, or to change the business model)
Modularity of the platform	- Number of APIs - Number of modules shared and reused by partners - Number of modules developed by partners
Ability to reuse components and complementary products	- Level of reciprocity needed by the license - Generic versus specific nature of components - Degree of component embeddedness in partner’s products - Number of competing versions (forks) of the product
Ecosystem governance structures	- Type of leadership model in the ecosystem - Reliance on open source foundations in governance - Clear rules and regulations for negotiations - Historical basis for making decisions (merit or influence-based)
Powerful influencers in the ecosystem	- Number of influential partners - Number of smaller players in the ecosystem and their alignment with influential partners - Relationship between influential partners and the focal firm - Merits for gaining influence

What is a healthy ecosystem?

A sustainable, expanding set of diverse participants with a stake in the success of a given open source product, governed by transparent and meritocratic means.

How do organizations sustain open source communities?

- Need to choose the 'right' community
- If it's right for you then other companies and organizations will recognize the worth of the community
- Joint investment of resources and bearing of any risk ensures long-term health
- Allows organizations to re-direct the energy and requirements of the product to suit their needs – which in turn extends organizational interests
- More organizations using the open source product is equal to sustained interest and sustainable communities

Thank you