

What is community?

Communities are simply groups of individuals sharing common interests. Both closed and open source software projects have communities of users, most of whom are relatively passive in terms of interaction with other community members. However either type of community may have members who decide to take on more active roles through, for example, reporting bugs, helping other users, writing documentation or evangelising.

Although these members' efforts are sometimes rewarded in closed source projects (e.g. Microsoft's Most Valued Professional programme), there is a clear limit on the types of contributions they can make. Because the code is not open for inspection, there is ultimately no way for users to actually go ahead and fix problems or develop new features and contribute code back. By contrast, in open source projects a flow of information (code and documentation) is welcomed from any member of the community into the centre.

Why build community?

Building an active and diverse project community provides a number of advantages. Testing and maintenance of code, often a significant proportion of the project costs, can be shortened and made more efficient with the help of the user community. Also for any given problem, including software support, one has the possibility to harness the brainpower of the entire community. Finally the more diverse the members of a community are, the more potential it has for innovation and long-term sustainability.

Attracting users and contributors

Open source projects may be initially very small, perhaps with one or two developers and hardly any users. However they should not hesitate to make their materials available online as early as possible. Moodle for instance, one of the most successful open source Virtual Learning Environments, partly owes its community success to this practice of encouraging contributors to release often and early.

Both presentation and functionality need to convince prospective users that the software being developed does something significantly better than the competition. Once users' interest has been secured, their barrier in joining the project must be low. Simple things, like the installation procedure, need to be extremely straightforward. But signing up users is not the end of the story. In the long run contributors are also needed. These may emerge from the immediate user-base, but may be also attracted by their business interests, or the technical challenge, kudos, or opportunities to improve their programming skills.

Managing community contribution

It is the responsibility of the leader to ensure that the full potential of open source can be realised within the new project. This does not happen automatically and has to be carefully managed. Developers will only contribute if the project remains a place where they want to return to. This involves rewarding hard work by giving credit where it is due and increased responsibility for more important contributions to those who want it.

Many early stage projects struggle with the inevitable support burden. Handled badly this can lead to users and contributors turning away. The leader need to employ appropriate people to do this critical support work, and at the same time encourage users to help each other by writing documentation and fixing bugs. A well documented technical infrastructure needs to be made available to facilitate project collaboration. An optimal set of tools should include a project website, a searchable mailing list, an issue tracker, and a version control system.

Structuring community

In many open source projects, members other than the project leader gradually become responsible for making key decisions. This results in a transition from the initial leader-centred structure to a consensus-based democracy. As the procedures associated with collective decision making become more complex, newcomers need more detailed instructions on how to take part in this process. While the body of community knowledge captured in the mailing lists can initially help, eventually a proper governance framework needs to be drawn to capture this shared understanding in a concise documentary form. The governance document helps ensure that the community has a life of its own that can survive for as long as there is a genuine need for the software.

Who is OSS Watch?

OSS Watch provides unbiased advice and guidance on the use, development, and licensing of free and open source software. OSS Watch is funded by the JISC and its services are available free-of-charge to UK higher and further education. If you want to find out more about open source software, we're the people to ask.

<http://www.oss-watch.ac.uk/resources/buildingcommunities.xml>

<http://www.oss-watch.ac.uk/>

info@oss-watch.ac.uk

