

A GQM plan for the evaluation of the trustworthiness of open-source software

Michele Chinosi, Vieri Del Bianco, Luigi Lavazza, and Davide Taibi

Università dell'Insubria

{michele.chinosi|vieri.delbianco|luigi.lavazza|davide.taibi}@uninsubria.it

Abstract

The diffusion of the usage of Open-Source Software in industry, in the public administration and in private environment is limited by the lack of a precise notion of trustworthiness. In fact, often people do not know how to evaluate whether an OSS product is “good” enough to be preferred to competing commercial alternatives.

In order to ease the diffusion of OSS, it is therefore necessary to identify, quantify, and assess the quality factors related to the software products as well as to the artefacts produced during software development that affect trust in open source software products. This will lead to a quantitative body of knowledge and a set of criteria for establishing trust in open source software.

This paper presents a GQM-based approach to the definition of the “trustworthiness” of open source software products. The Definition of the GQM plan is still in progress. Here a preliminary and incomplete version of the plan is reported, however sufficiently detailed to describe the concepts that we have tentatively identified to represent the notion of trustworthiness. The GQM plan is expected to indicate also how to quantify the aspects of OSS trustworthiness.

1 Introduction

Organizations need to evaluate OSS before deciding to use it. Although some methods have been proposed for this purpose (see for instance [1]) there is not yet a general agreement about the characteristics of OSS that contribute to its trustworthiness.

Therefore, the paper focuses on defining an adequate notion of trustworthiness of software products and artefacts and identifying a number of factors that influence it. The activity described here will provide the bases for building models for the trustworthiness of products and artefacts.

The final goal is to provide both developers and users of OSS with an instrument that guides them in the decision whether a given OS program (or library or other piece of software) is “good enough” to be used in an industrial or professional context, possibly instead of commercial software. We name “trustworthiness” the set of qualities that are of interest for the users and developers.

The concept of trustworthiness is being investigated in the QualiPSo project (<http://www.qualipso.eu>). In particular, current activities are exploring the trust-related goals of OSS developers and users.

A GQM [2] plan provides the definition for the various dimensions of trustworthiness that is unambiguously understood and is suitable to build a widespread consensus on them. This is a necessary step, in that all too often in Software Measurement there is a lack of agreement about the real meaning of a number of software qualities. Based on these dimensions, a set of measures will be defined, so as to capture the various components of trustworthiness from different

viewpoints, and they will be collected based on both static (i.e., based only on the analysis of the source code or artefacts) and dynamic measures (i.e., based on the execution of the software code or, wherever possible, the software artefacts). The definition of metrics and the execution of the measurement will be object of future work.

Next section presents a preliminary version of the GQM plan, its definition being still in progress.

2 The GQM plan

A fundamental observation is that an organization perceives the trustworthiness of the OSS on the basis of the role that the OSS plays with respect to the organization itself. For instance, the organization could be a producer of OSS or a user, a customizer, a value adder, etc. Since the role of OSS is fundamental to determine the trustworthiness criteria, we decided to define a GQM plan for each relevant case of OSS role.

In the rest of the section we present a sketch of the GQM plan that we are currently defining. Note that often, in the current paper, the term “trustworthiness” assumes a broader meaning: indicating the whole set of qualities that a potential user seeks or appreciates in an OSS product.

Goal 1: Analyse OSS for the purpose of evaluating the trustworthiness from the point of view of users (people who use the SW as is).

Quality foci:

- Software quality (direct evaluation of functional aspects)
- Software quality (direct evaluation of non-functional aspects)
- Software quality (direct or indirect evaluation of internal qualities, e.g., code qualities)
- Software quality (both functional and non functional, evaluated indirectly, e.g., considering the reputation of developers, the opinions of the users community, etc. [3][4])
- Software quality (indirect evaluation, through evaluations, certifications and assessments made by an external organization)
- Economic factors (cost of deployment, training, etc.)
- Availability and quality of support, guidance, documentation, etc.
- Availability and quality of SW maintenance in the short term (e.g., bug correction)
- Perspective of support in the future (i.e., is it likely that some organization or community will provide services related to the OSS product in the future)
- Perspective of SW maintenance in the future (bug fixes, size of the community that currently use the product in a productive environment)
- Perspective of SW evolution in the future (new releases)

Variation factors:

- Problem domain (the problem domain of the software system could change the relative importance of many factors, or even change the way the factors will be measured)
- Software criticality (the software criticality could change the relative importance of the measured software factors)

Goal 2. Analyse OSS for the purpose of evaluating the trustworthiness from the point of view of developers (e.g., organizations involved in development, organizations whose business involves OSS development)

In this case the trustworthiness is perceived as the quality that the organization wants to build in the OSS to be released. The perception is complicated by the fact that often the development is carried out in cooperation with other organizations and individuals, therefore part of the trustworthiness derives from the work carried out by others.

Quality foci:

- Software quality (direct evaluation of functional aspects)
- Software quality (direct evaluation of non-functional aspects)
- Software quality (direct evaluation of internal qualities, e.g., code qualities)
- Perspective of support in the future by other organizations involved in the development
- Customer related issues

The variation factors take into account the properties that affect the way the development work is carried out, in particular the costs and amount of effort that the developers can afford:

- Economic factors (cost of development)
- Business model

Goal 3. Analyse OSS for the purpose of evaluating the trustworthiness from the point of view of integrators (e.g., organizations that integrate SW into their products, especially if modifications are required).

This goal combines the points of view of the first two goals. Quality foci are:

- How the internal code quality affects the trustworthiness of the final product.
- Economic factors (cost of familiarization, maintenance, integration, customization, etc.)
- Legal (mainly license-related) issues. OSS licenses are very important since they can constrain the usage or distribution of OSS-based software.
- Availability of support, guidance, documentation, etc.
- Availability of technical support, guidance, documentation, etc.
- Availability of SW maintenance in the short term (e.g., bug correction)
- Perspective of support in the future by other organizations involved in the development
- Perspective of SW maintenance in the future (new releases)
- Issues related to customers (either internal or external)

Goal 4. Analyse the OSS development process for the purpose of evaluating the relation between the trustworthiness of OSS and the characteristics of the process.

Quality foci:

- Trustworthiness of products (as defined in previous goals)
- Maturity of the software development process (including development release planning, creation and maintenance of product documentation, process modelling, etc.)
- Qualification and experience of developers
- Quality assurance practices (including testing, measurement of properties, etc.) used in the development process
- Development process retrospectives (the development process includes specific activities to rethink the development process itself, adapting to changed requests, environment, developers availability, etc.)

Variation factors:

- Business model of development

Goal 5. Analyse the OSS usage process for the purpose of evaluating the relation between the (perception of) trustworthiness of OSS and the characteristics of the process.

Quality foci:

- Trustworthiness of products (as defined in previous goals)
- Adaptations (type and amount)
- Extensions (type and amount)
- Functionality (e.g., API) used
- User interfaces used (type and amount)

Variation factors:

- Type and experience (including the familiarity with the OSS product) of the user
- Type of usage (customization, integration in larger products, extension, localization, ...)
- Effort and duration
- Methodology and tools employed
- Status of the OSS project
- Platform and language

3 Conclusions and future work


The GQM plan reported above is a –still in progress– attempt to capture the essence of OSS trustworthiness. We aim at contributing to the creation of a common understanding of the notion of trustworthiness, and more important, to raise the awareness that the success and diffusion of OSS depends on the fulfilment of a set of quality conditions, that the GQM plans should represent explicitly.

The work sketched here will proceed with the definition of questions and metrics, towards the completion of the GQM plan. As usual, the validation of the plan will be carried out as part of the measurement and analysis process.

Among the objectives of future work there is also the conclusion of the analysis of users needs with respect to OSS. This should provide indications about the adequacy of this preliminary version of the GQM plan for OSS trustworthiness.

Among other objectives, we look forward to understand to what extent a common indicator for trustworthiness can be built, and how the specificity of users needs must be taken into account in the evaluations. If the users, developers and maintainers needs vary greatly and cannot be reconciled, a different approach to the assessment of OSS products could be taken into account. In such cases we could define a set of basic GQM plans that define trustworthiness for frequent classes of users, developers, etc. Each of these trustworthiness classes would be based on a set of features considered relevant by all users, developers, etc. in the class. Then the new approach could support the identification of the quality factors to be considered in order to tailor one or more of the predefined GQM plans, in order to match the specific needs of the user or developer.

Acknowledgments

The research presented in this paper has been partially funded by the IST project  (http://www.qualipso.eu/), sponsored by the EU in the 6th FP (IST-034763).

References

- [1] D.Taibi, L.Lavazza, S.Morasca. OpenBQR: a framework for the assessment of Open Source Software, *Open Source Software 2007*, Limerick, June 2007.
- [2] Basili V., and Rombach H.D., The TAME project: towards improvement-oriented software environments, *IEEE Transactions on Software Engineering*, June 1988.
- [3] Crowston K., Annabi H., and Howison J., Defining open source software project success. In *Proceedings of International Conference on Information Systems (ICIS)*, 2003.
- [4] Crowston, K., Annabi, H., Howison, J. and Masango, C., Towards A Portfolio of FLOSS Project Success Measures. In *Collaboration, Conflict and Control: The 4th Workshop on Open Source Software Engineering, International Conference on Software Engineering (ICSE 2004)*, Edinburgh, Scotland, May 25.