Chapter 5.5
Volunteers in Large Libre Software Projects:
A Quantitative Analysis Over Time

Martin Michlmayr
University of Cambridge, UK

Gregorio Robles
Universidad Rey Juan Carlos, Spain

Jesus M. Gonzalez-Barahona
Universidad Rey Juan Carlos, Spain

ABSTRACT

Most libre (free, open source) software projects rely on the work of volunteers. Therefore, attracting people who contribute their time and technical skills is of paramount importance, both in technical and economic terms. This reliance on volunteers leads to some fundamental management challenges: Volunteer contributions are inherently difficult to predict, plan, and manage, especially in the case of large projects. In this chapter we present an analysis of the evolution over time of the human resources in large libre software projects, using the Debian project, one of the largest and most complex libre software projects based mainly in voluntary work, as a case study. We have performed a quantitative investigation of data corresponding to roughly seven years, studying how volunteer involvement has affected the software released by the project, and the developer community itself.

INTRODUCTION

Volunteer contributions are the basis of most libre[1] software projects. However, the characteristics, and the way of working of volunteers, can be quite different from those of employees who are the main force behind traditional software development. Volunteers can contribute with the amount of effort they want, can commit for the time period they consider convenient, and can devote their time to the tasks they may prefer, if the context of the project makes that possible (Michlmayr & Hill, 2003). But even in this apparently difficult environment, many libre software projects have produced systems with enough quality and functionality to gain significant popularity. Therefore,

the fairly unstructured collaboration of volunteers has been demonstrated as a viable software development strategy, even if it is associated with certain challenges related to project management and quality (Michlmayr, 2004). In this chapter we explore how these voluntary contributions evolve over time in one of the largest libre software projects, Debian.

For our purposes, we will define volunteers as those who collaborate in libre software projects in their spare time, not profiting economically in a direct way from their effort. Volunteers can be professionals related to information technologies, but in that case their activity in the libre software project is not done as a part of their professional activity. Although the vast majority of participants in libre software projects comply with our definition, it is important to note that there are also non-volunteers, that is, paid people (normally hired or contracted), who produce libre software. German has studied paid employees from various companies in the GNOME project (German, 2004). He notes that they are usually responsible for less attractive tasks, such as project design and coordination, testing, documentation, and bug fixing. Also, "[m]ost of the paid developers in GNOME were, at some point, volunteers. Essentially for the volunteers, their hobby became their job."

The involvement of volunteers, of course, raises new economic and business model issues that have to be taken into account in commercial strategies around libre software. Collaboration from volunteers is difficult to predict, but if it is given, it may add value to a software system in very economic terms for a software company.

The structure of this chapter will be as follows. In the second section we discuss the nature and, in particular, the tasks performed by volunteers, paying special attention to those who contribute to Debian, the case study investigated in this chapter. Following this section, a set of research questions regarding volunteer participation will be raised. The primary goal of this chapter is to answer these questions based on quantitative data. The methodology for retrieving the quantitative data used in this study is first given. In this section, we also propose a number of measures that will allow us to answer the questions. The results we have obtained as part of this study will be presented and commented on in depth for the Debian project. Finally, conclusions, applicability of the methodology, and further research will be discussed.

THE DEBIAN PROJECT AND ITS VOLUNTEERS

Debian is an operating system completely based on libre software (Monga, 2004; O'Mahony, 2003). It includes a large number of applications, such as the GNU tools and Mozilla, and the system is known for its solid integration of different software components. Debian's most popular distribution, Debian GNU/Linux, is based on the Linux kernel. Ports to other kernels, such as Hurd and FreeBSD, are in development.

One of the main characteristics of the Debian distribution is that during the whole life of the project it has been maintained by a group of volunteers, which has grown to a substantial number. These individuals devote their own time and technical skills to the creation and integration of software packages, trying to supply users with a robust system which provides a lot of functionality and technical features.

Following our definition of volunteers, all maintainers in Debian are volunteers. Some employers of people who act as Debian maintainers in their spare time permit their staff to devote some of their time to Debian during work hours. Nevertheless, the majority of work by most Debian maintainers is performed in their spare time. In contrast to some projects, such as the Linux kernel and GNOME, there are no Debian maintainers who are paid to work on the system fulltime, even though a number of organizations have a com-

Related Content

A Consensus of Thought in Applying Change Management to Information System Environments

Jeffrey S. Zanzig, Guillermo A. Francia IIIand Xavier P. Francia (2015). *International Journal of Information System Modeling and Design (pp. 24-41).*

www.irma-international.org/article/a-consensus-of-thought-in-applying-change-management-to-information-system-environments/142514

How to Systematically Embed Cycles in Balanced Hypercubes

Hsuan-Han Chang, Kuan-Ting Chenand Pao-Lien Lai (2017). *International Journal of Software Innovation (pp. 44-56).*

www.irma-international.org/article/how-to-systematically-embed-cycles-in-balanced-hypercubes/169917

Automatic Recognition of Traffic Signs with 3D Distance Estimation for Intelligent Vehicles

Nadra Ben Romdhane, Hazar Mlikiand Mohamed Hammami (2017). *International Journal of Software Innovation (pp. 70-86).*

www.irma-international.org/article/automatic-recognition-of-traffic-signs-with-3d-distance-estimation-for-intelligent-vehicles/176668

Improving Credibility of Machine Learner Models in Software Engineering

Gary D. Boetticher (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications (pp. 2865-2882).*

www.irma-international.org/chapter/improving-credibility-machine-learner-models/29540

A Strategy for Managing Complexity of the Global Market and Prototype Real-Time Scheduler for LEGO Supply Chain

Bjorn Madsen, George Rzevski, Petr Skobelevand Alexander Tsarev (2013). *International Journal of Software Innovation (pp. 28-39).*

www.irma-international.org/article/a-strategy-for-managing-complexity-of-the-global-market-and-prototype-real-time-scheduler-for-lego-supply-chain/89773