Chapter 1.7
On the Role of Public Policies Supporting Free/Open Source Software

Stefano Comino
University of Trento, Italy

Fabio M. Manenti
University of Padua, Italy

Alessandro Rossi
University of Trento, Italy

ABSTRACT

Governments' interest in free/open source software is steadily increasing. Several policies aimed at supporting free/open source software have been taken or are currently under discussion all around the world. In this chapter, we review the basic (economic) rationales for such policy interventions and we present some summary statistics on policies taken within the European countries. We claim that in order to evaluate correctly the consequences of such interventions one has to consider both the role and the administrative level at which such decisions are taken as well as the typology of software that is involved. Moreover, we argue that the level playing field cannot be taken for granted in software markets. Therefore, non-intrusive public policies that currently prevail at the European level in terms, for instance, of the promotion of open standards or in terms of campaigns aimed at informing IT decision-makers, are likely to be welfare enhancing.

INTRODUCTION

Governments' interest in free/open source (F/OS) software is steadily increasing. In Europe, this interest has become visible in the Lisbon Strategy and in the corresponding eEurope Action Plans 2002 and 2005 approved by the European Commission where it has been clearly stated the key role of open source software and open standards in pursuing the general objective of giving all citizens the opportunity to participate in the global information society.[1]

All over the world governments are considering various policies to support F/OS software; these policies go from the provision of "best practices" for the usage of open source to information cam-

paigns aimed at making markets participants aware of all software alternatives, from simple expressions of preference towards F/OS software to large scale adoption of open source solutions in governments' offices and schools.

The role of the public sector in the software market is of primary importance. Governments not only set the legal and regulatory framework where economic agents interact, but they are also big software purchasers;[2] this double role makes governments key players in determining the future evolution of software markets and it is therefore of crucial interest to understand both the motivations and the effects of governments' interventions in this sector.

This chapter critically reviews the main arguments in favor or against public intervention supporting F/OS; we also provide some empirical evidence about the various public interventions that are already in place in Europe. The chapter is structured in three parts: in the first part, we provide a general analytical framework; public interventions may occur at different administrative levels (i.e., from municipalities to national or supra-national level), and they may have different motivations. These complexities have not received enough attention in the previous analyses on public interventions towards F/OS; the aim of this section is to offer a possible taxonomy for governmental policies in the software market and to discuss the many rationales for intervention but also the counterarguments that often have been put forward. In the following section, we present some evidence concerning the main public initiatives in Europe. Rather than focusing on any specific case study, we have collected information from the European IDABC, the program documenting the major initiatives supporting F/OS within the European Union. In this way, we have been able to draw some general considerations on the motivations and the characteristics of governments interventions implemented all across the EU. The subsequent section concludes by bridging the theoretical discussion with the empirical analysis.

We claim that, if one considers that the largest share of the software market is represented by self-developed or customized products, the existing literature has placed too much emphasis on packaged software and arguments against public support of F/OS might be improperly grounded. Moreover, we believe that the level playing field cannot be taken for granted in software markets. Therefore, non-intrusive public policies that currently prevail at the European level in terms, for instance, of the promotion of open standards or in terms of campaigns aimed at informing IT decision makers, are likely to be welfare enhancing.

BACKGROUND: A GENERAL FRAMEWORK

It is useful to start our analysis by providing a general framework for discriminating the large heterogeneity of public interventions in the software market. In particular, we claim that, in order to judge correctly rationales, motivations, and consequences of public interventions, it is important to distinguish between the various roles played by policy makers and the various categories of software involved. We argue that many existing contributions, both in the scholarly and in the practitioners' debate, have not clearly taken into account these distinctions.

Public administrations, institutions, and governments play a double role in the software industry. On the one side, being big spenders for software licenses and software development, their adoption/use decisions represent a significant share of the demand thus having a major impact on market equilibrium. On the other side, by acting as legislators and regulators, governments do in various ways determine the evolution of the market; for instance, it is quite evident that the legislation towards intellectual property rights, either based on strong patent protection as in the U.S. or on weaker copyright legislations as it is within the EU, has a major influence on the functioning of

Related Content

Requirements Engineering in a Model-Based Methodology for Embedded Automotive Software
Jean-Louis Boulanger (2010). *Handbook of Research on Software Engineering and Productivity Technologies: Implications of Globalization (pp. 15-27).*
www.irma-international.org/chapter/requirements-engineering-model-based-methodology/37021

Reuse in Agile Development Process
Chung-Yeung Pang (2020). *Software Engineering for Agile Application Development (pp. 164-187).*
www.irma-international.org/chapter/reuse-in-agile-development-process/250441

Modeling Requirements With Diabetes Using Supervised Machine Learning Techniques
Priyal Dangi (2023). *The Software Principles of Design for Data Modeling (pp. 29-40).*
www.irma-international.org/chapter/modeling-requirements-with-diabetes-using-supervised-machine-learning-techniques/330485

Metadata Design of a Content Management System for Music Virtual Learning Environment
Agnes Wai Yan Chan, Elza Yin Ling Chan, Will Chi Kit Lee, Benny Yu Ming Leungand Dickson K.W. Chiu (2015). *International Journal of Systems and Service-Oriented Engineering (pp. 56-76).*
www.irma-international.org/article/metadata-design-of-a-content-management-system-for-music-virtual-learning-environment/125844

Natural Language Processing: An Inevitable Step in Requirements Engineering
A. Egemen Yilmaz (2014). *International Journal of Systems and Service-Oriented Engineering (pp. 68-83).*
www.irma-international.org/article/natural-language-processing/104655