Chapter X
Dependencies, Networks, and Priorities in an Open Source Project

Juha Järvensivu
Tampere University of Technology, Finland

Nina Helander
Tampere University of Technology, Finland

Tommi Mikkonen
Tampere University of Technology, Finland

ABSTRACT

Dependencies between modern software projects are common. Jointly, such dependencies form a project network, where changes in one project cause changes to the others belonging to the same project network. This chapter discusses the issues of dependencies, distances, and priorities in open source project networks, from the standpoint of both technological and social networks. Thus, a multidisciplinary approach to the phenomenon of open source software (OSS) development is offered. There is a strong empirical focus maintained, since the aim of the chapter is to analyze OSS network characteristics through an in-depth, qualitative case study of one specific open source community: the Open Source Eclipse plug-in project Laika. In our analysis, we will introduce both internal and external networks associated with Laika, together with a discussion of how tightly they are intertwined. We will analyze both the internal and the external networks through the elements of mutuality, interdependence, distance, priorities, different power relations, and investments made in the relationships—elements chosen on the basis of analysis of the network studies literature.

INTRODUCTION

Dependencies between modern software projects are commonplace. Jointly, such dependencies form a network, where changes in one project, or part thereof, cause changes in others. In using formal contracts applicable in the traditional industrial setting, these dependencies are defined by legalities and customer/subcontractor relations, which can be easily managed. However, in an open source project, dependencies are based not on some explicitly defined formalization but instead on how the different developer communities view and use each other and themselves. Furthermore, the issue of project priorities requires similar consideration.

In this chapter, we discuss dependencies, networks, and priorities in OSS development. As an example community we use Laika, an Open Source Eclipse plugin project that eases code creation for the Maemo platform. We discuss both external networks, consisting of communities that relate to (or are related to by) Laika, and internal networks that include the developers of the system. The contribution of the chapter and its underlying research question lie in establishing a connection between established network theory and practices in OSS development, on the one hand, and in discussing the organization, evolution, and values leading to the priority selection established in the Laika community, on the other. More precisely, we address the rationale of establishing a mode of cooperation between different developer communities as well as internal networking within a single community where several organizations are involved. This supplies a context in which to study the approach taken to work allocation, which will also be addressed. This chapter is inspired by the background in which two of the authors were directly associated with Laika, with the other having the role of an external observer.

The rest of the chapter is structured as follows. Next we introduce a related theory of networks that we use as a guide in analyzing the properties of Laika. We then discuss Laika and its internal and external networks, and provide a discussion of the goals of the chapter. Finally, we discusses future avenues of research, and offer final remarks.

BACKGROUND

Network Approach as Theoretical Framework

Networks are a contemporary topic that has been studied from several different perspectives and under various scientific disciplines. The term "network" can refer to, for example, an information network in the form of interconnection of layers of computer systems (Macdonald & Williams, 1992; Meleis, 1996); a social network in the form of a social structure among actors, mostly individuals or organizations (Baker, 1994; Barnes, 1972; Hill & Dunbar, 2002; Scott, 2000; Wasserman & Faust, 1994); or a business network in the form of a set of exchange relationships between organizations (Achrol & Kotler, 1999; Easton, 1992; Håkansson & Snehota, 1995; Möller & Halinen, 1999; Möller, Rajala, & Svahn, 2002). In this chapter, we use the term network theory to refer to the so-called network approach introduced by a group of scholars basing their work on theories of social exchange coupled with more economically oriented industrial insights (Möller & Wilson, 1995). The network approach discussed in this chapter aims at providing conceptual tools for analyzing both structural and process characteristics of networks formed among different open source projects and within a single specific open source project, Laika.

Early developers of the network approach, Håkansson and Snehota (1989) point out that the network approach takes into consideration the relationships among various actors. All of the actors, their activities, and their resources are bonded, linked, and tied up together, and in this way they build up a wide network. A basic assumption with the network approach involves the relationship as a fundamental unit, from which proceeds understanding of the network as a sort of cluster of relationships.