Chapter 56
Open Innovation:
Assessing the Socio-Economic Factors of Global Software Development

Noel Carroll
University of Limerick, Ireland

ABSTRACT

Software engineering is largely concerned with the methodical, systematic production of quality software to support organizational goals. Despite significant advances in technology over the last decade, software engineering still heavily relies on human efforts and human interaction to generate economic contributions for an organization. In more recent years, the question of software service complexity has become central to Global Software Development (GSD) with a view of developing software across a global landscape. However, few efforts have surfaced to challenge the complexity of the relational infrastructure of software teams that support the development of software architecture. This is important in order to sustain and support lean software development and organisational structures particularly in an open service innovation environment. However, from a socio-economic perspective, there are few theoretical efforts that attempt to introduce new insights on how the human factors contribute towards a GSD value co-creation. The objective of this chapter is to examine the application of Social Network Analysis (SNA) and i (i star) modelling techniques to examine how we could model the economic impact of software relational structures. The chapter explores how i* models leverage SNA concepts to model GSD development. This chapter also offers a discussion on the theoretical development of the socio-economics of GSD in an "open innovation" context.*

INTRODUCTION

Open innovation is an emerging paradigm which exposes organisations to external or networked resources to avail of newfound service capabilities (Chesbrough, 2003). While open innovation continues to receive more attention, it also highlights the importance of assessing the socio-economic contributory efforts of software teams developing software architecture. Nowadays, organisations are tasked with uncovering "*what is out there*" to combine external capabilities with internal innovation resources. Thus,

DOI: 10.4018/978-1-5225-3923-0.ch056

over the past decade we witnessed two important key service developments: 1) increase in technology capability and accessibility, and 2) the emergence of new business models. Similar to the growth in information technology (IT) throughout the 1990's, these developments opened up new opportunities to apply technological capabilities to address business needs. This is particularly evident within the field of software engineering (for example, Šmite et al. 2010) especially in global software development (GSD). Thus, uncovering what stabilises a GSD team is a critical element within the software system since it defines the structures and relations between them to develop software architecture. Software architecture is defined as "*the fundamental organisation of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution*" (IEEE, 2000). The relational properties are of significant interest as this chapter focuses on the need to understand how GSD can facilitate communication and interaction patterns to reuse software components for software architecture in an open innovation environment. This allows us to understand 'the system', organise its development process and plan for methods to reuse software components and support its evolution. At a most basic level, the software architecture will be specified using diagrams which have modelled subsystems, interfaces between them, components diagrams, classes, diverse descriptions, and the basic group of use cases. The architecture it typically validated by clients and developers to achieve the desired functionality of a service. A service may be defined as "*a change in the condition of a person or a good belonging to some economic entity brought about as a result of the activity of some other economic entity, with the approval of the first person, or economic entity*" (Carroll, 2012). For the purpose of this chapter, the author is primarily interested in uncovering the socio-economic dynamics which contribute to the evolution of a GSD team. It examines how the relationship between stakeholders and decision-making influence the socio-economic dynamics of software development. This chapter offers a discussion on the need to apply social and economic considerations to software architectural development and decision-making tasks.

Research Focus

The modern service environment represents a paradigm shift away from technology as an organisational asset to technology enabled service. GSD is one such development which is becoming the normal practice in the software industry (Conchúir et al. 2009). GSD may be defined as software development processes undertaken at geographically separated locations across national and organisational boundaries in a coordinated fashion involving real time (synchronous) and asynchronous interaction (Conchúir et al. 2009). Often the benefits of GSD described throughout literatures are summarised as follows:

1. To achieve cost benefits;
2. Reduced development costs;
3. To manage shortages in specific skills (larger and better-skilled developer pool);
4. Greater need of customer and market proximity;
5. Attraction of national legislations;
6. Greater time zone effectiveness;
7. Improved flexible resource capacity;
8. Availability of open source development;
9. Possibility of greater innovation;
10. Learning and transfer of best practices.