Chapter 2
Exploratory Analysis of Free and Open Source Software Ecology

K.G. Srinivasa

Chaudhary Brahm Prakash Government Engineering College, India

Ganesh Chandra Deka

M. S. Ramaiah Institute of Technology, India

Krishnaraj P.M.

M. S. Ramaiah Institute of Technology, India

ABSTRACT

Shared repositories provide a host of services to start and sustain a FOSS project. They also share the details of projects with researchers. Sourceforge.net is a popular and populous forge with total number of projects exceeding 400,000 and developers counting more than 3 million as of Jan 2015. The evolution of this forge is studied and it was found that there is a small slide in the number of developers since September 2011. The existence of power law in Sourceforge.net is confirmed. The visualisation of developer relations reveal that there is a separate core and periphery groups of developers in Sourceforge. net and this trend was found to repeat in other forges like Freecode and Rubyforge.

INTRODUCTION

Free and Open Source Software (FOSS) is characterised mainly by its licensing terms. The Free Software licences and Open Source licences, though different in their relationship with commercial software, together provides an alternate model of software distribution. But FOSS is also important for the development model it follows. The success of FOSS lies in demonstrating the feasibility of developing a complex artefact like software by involving global set of volunteers and using Internet as a communication medium. Ranging from a lone developer to literally tens of thousands of people and organisation, FOSS ecology today is probably world's largest virtual software development entity. But it is not necessary that

any person or organisations who want to develop FOSS must follow this process. They can develop the software in-house without involving public but still release the software under FOSS licences.

Given the advantages of public participation model as demonstrated by success of GNU-Linux, many FOSS projects are developed in similar way. In the early days of GNU project and even during the initial stages of GNU-Linux development, the project leader would normally release the source code to public. Interested people would download, use, test, find bugs and then either reported or send fixes to the leader. The leader would have final say regarding the inclusion of bug fixes and new features. This model of development is still followed today but in place of a single leader there is a team which is normally formalised in all mature FOSS projects.

One of the major contributions of FOSS which is normally understudied is how much it has contributed towards the system and software development tools. Given the fact that most developers are mainly involved in FOSS to write software for their own use, this is quite natural. The growth of Internet coupled with the spread of computers during late 1990's enabled much larger participation in FOSS projects. This necessitated a mechanism which can automate the build and release processes. Separately there was a need for a communication platform beyond usenet and irc which could connect developers and end users. Bug reporting, feature request and general support also needed to be supported. Therefore, FOSS projects slowly started moving away from niche environments to public platform like Internet.

Mature and popular projects mostly host their projects in dedicated websites which support a range of features for efficient project management. Almost all technical and pubic communication details of the projects hosted in these sites are available for researchers. But given the fact that many FOSS projects start with single developer, it is not practical for every project to have its own website with all features. Also, the visibility of a project decreases if it works independently. Therefore, there exists multiple repositories which provide common facilities required to start and sustain a software project. They also provide a platform for developers to interact with projects they are interested. This multiplicity effect attracts many developers and organisations to host their projects in such repositories which are also called as forges.

In accordance with the promise of public development model of FOSS, such forges make the data regarding the projects hosted by them available for researchers. Mining Software Repositories has become a standard research topic providing a host of interesting issues to work upon. The software engineering researchers have access to such exhaustive data sets for first time and there are many opportunities to learn about software development process. Design, quality assurance and project management studies today include data from FOSS forges. The richness of data which include communication details is attracting researchers from rainbow domains like anthropology, sociology, economics, law and political science. Together they are trying to interpret this wonder phenomenon where gifted programmers seem to work without pay to create industry grade software and then distribute it at no cost along with source code without any restrictions on further use, modification and redistribution.

Evolution of Sourceforge.net

Sourceforge.net is probably more well known and one of the biggest FOSS repositories. As on Jan 2015 it hosts 400,000 projects and has 3 million registered users. Several high-profile projects like Vlc player, eMule are hosted in this site. It shares the data regarding projects hosted in the site with researchers through University of Norte Dam, USA.

SourceForge.net uses relational databases to store project management activity and statistics. There are over 100 relations (tables) in the data dumps provided to Notre Dame. Some of the data have been

7 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/exploratory-analysis-of-free-and-open-source-software-ecology/286564

Related Content

The Chase for OSS Quality: The Meaning of Member Roles, Motivations, and Business Models
Benno Luthigerand Carola Jungwirth (2007). *Emerging Free and Open Source Software Practices (pp. 147-168).*
www.irma-international.org/chapter/chase-oss-quality/10086

Quantifying Reuse in OSS: A Large-Scale Empirical Study
Eleni Constantinou, Apostolos Ampatzoglouand Ioannis Stamelos (2014). *International Journal of Open Source Software and Processes (pp. 1-19).*
www.irma-international.org/article/quantifying-reuse-in-oss/150449

Prediction Models and Techniques for Open Source Software Projects: A Systematic Literature Review
M.M. Mahbubul Syeed, Imed Hammoudaand Tarja Systä (2014). *International Journal of Open Source Software and Processes (pp. 1-39).*
www.irma-international.org/article/prediction-models-and-techniques-for-open-source-software-projects/124002

Greasemonkey and a Challenge to Notions of Authorship
Brian D. Ballentine (2007). *Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives (pp. 12-22).*
www.irma-international.org/chapter/greasemonkey-challenge-notions-authorship/21175

Open Source E-Learning Systems: Evaluation of Features and Functionality
Phillip Olla (2007). *Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives (pp. 638-648).*
www.irma-international.org/chapter/open-source-learning-systems/21222