

Open Source Software Ecosystems in Health Sector: A Case Study from Sri Lanka

Roshan Hewapathirana¹(✉), Pamod Amarakoon², and Jørn Braa¹

¹ University of Oslo, Oslo, Norway

roshan.hewapathirana@gmail.com, jbraa@ifl.uio.no

² Nutrition Coordination Division, Ministry of Health, Colombo, Sri Lanka
pamod@pamod.net

Abstract. A software ecosystem consists of a software platform, a set of internal and external developers and domain experts in service to a community of users that compose relevant solution elements to satisfy their needs. Open source is well-known for its potential to frame software ecosystems with its networking tendency and provision for further customization with access to software source code. Open source is increasingly becoming the choice for health information system implementations in low resource settings.

This longitudinal case study was designed to study the research question, how a software ecosystem is being built around an open source health information system implementation. Empirically the study was positioned in a multi-sector initiative identifying and support nutritionally at-risk households to eliminating malnutrition. The discussion reveals how new dependencies between health and non-health sector actors were created with the emerging software ecosystem based on an open source framework and supplementary custom-built web and mobile components.

Keywords: Software ecosystem · Free and open source software · Health information system

1 Introduction

Compared to the traditional software project perspective, software ecosystem (SE) is an emerging trend within the software industry [1]. A SE typically consists of a software framework, internal and external developers, domain experts and a community of users that compose the relevant solution elements [2]. The SE is the choice to construct large software system on top of a software framework by composing components developed by actors both internal and external [3]. Hence, Free and Open Source Software (FOSS) provides a viable framework for growing a SE from the angles of implementation technology, development methodology and governance [4]. Being a relatively new concept, SE has not been discussed adequately in the IS discourse [5]. Hence, SE literature needs more empirical studies from various domains, such as open source and health [6].

Sri Lanka possesses a well-established health care delivery model, with most of the health indicators are at a comparable level to those of the developed world. However,

the nutritional indicators were lagging behind compared to other health indicators. Malnutrition has a multi factor contribution including both health and non-health denominators. Thus, the revised National Nutrition Policy (NNP) of Sri Lanka [7] suggested inviting non-health sector stakeholders to the nutrition management tasks giving them active roles in eliminating malnutrition. This demanded an integrated information system to monitor nutritional status and to track health and non-health intervention to coordination across different sectors. An open source HIS with supplementary custom developed web and mobile components was introduced to overcome the challenge of integrated information need across different sectors. This FOSS HIS implementation around multi-sector participation was an ideal empirical setting to study a domain specific SE. Hence, a longitudinal case study was aimed at understanding how a SE is being built around an open source HIS implementation, expecting to contribute to the growing body of knowledge on SE.

The organization of the rest of the paper is as follows. The second section reviews the current literature on the theoretical underpinning of the study while the third section elaborates the research approach and methodology. The next section reveals the findings of this longitudinal case study. Followed by which, the fifth section presents the analysis and the discussion leading to the conclusion of the study which is presented in the section six.

2 Theoretical Background

2.1 Software Ecosystems

A SE is a means to construct a large software system on top of a software framework by composing components developed by actors both internal and external. For the purpose of this study the SE was defined as to “*consists of a software platform, a set of internal and external developers and community of domain experts in service to a community of users that compose relevant solution elements to satisfy their needs*” [2]. This perspective differs from traditional software project approach in several important aspects. In a SE, the initiating actors (the client organisation) don’t necessarily own the software produced by the contributing actors and may not hire the contributing actors [3]. In comparing traditional software projects to SE, it was shown that the scope of a traditional software project typically is intra-organizational. Whereas the scope of a SE is much broader and is including external developers and the further extensions that they provide as well as contributions from other parties [8]. SE are mainly categorised in three broad categories as being operating system-centric, application-centric and end-user programming centric SEs [9]. The application-centric software ecosystems, such as the empirical setting of this research, is organised around a domain specific application.

The general composition of a SE is the software firm (framework developer), (3rd party) software suppliers, client firm, intermediaries and client firm’s customers. According to Dittrich [1] framework developers and 3rd party application developers are both important in a SE. This is particularly the case in the FOSS domain, where the core FOSS firm plays the role of framework developers. In this context 3rd party

developers and FOSS implementers play the role of application developers working on extending the generic functionalities of the open source framework by aligning it with the needs of the implementation domain.

2.2 Open Source

FOSS is a well-established practice to manage both software development and distribution. It permits access to the software source code, together with the permission to modify the source code as well as to redistribute the derived works¹. Given this kind of end-to-end control, FOSS is generally a good a framework for building SEs [4]. Open source software provides the capability to develop complex systems on freely available source code and enables constructing a SE without large initial investment [10]. FOSS reduces system implementation costs by eliminating vendor monopoly. Furthermore, it promotes indigenous technology development by allowing access to the source code which facilitates the global to local transfer of knowledge [11]. Additional benefits of FOSS include vendor neutral technology through free access to the source code and reduced total cost of ownership with no licence fee [12].

The open source phenomenon has undergone a significant transformation from its free software origin to a more mainstream, commercially viable form which is referred to as *Open Source 2.0* [13]. Clients are willing to pay for customizing Open Source Software for organizational business needs because customization related services are critical factors influencing the OSS adoption in many organizations [14]. Hence, FOSS firm also look to 3rd party software service providers to add specific functionalities to the core framework, which is beyond the capacity of FOSS firm alone. Several FOSS governance models are suggested to describe this 3rd party contribution in open source adoption, such as the Third Party Service Provider model proposed by Krishnamurthy [15]. This FOSS business models can be regarded as a stakeholder participation model in SE around open source adoption.

3 Research Approach and Methodology

This longitudinal interpretive case study was conducted in the State health sector of Sri Lanka over a period of two years from 2014 to 2016. It was empirically situated within a large scale FOSS HIS implementation effort, which is aimed at establishing a multi-sector stakeholder network consisting of health and non-health sectors around the implementation of a nutrition information system. We positioned ourselves within the qualitative research practice [16] with a case study approach [17]. The empirical work was guided by the research question, how a SE is being built around an open source HIS implementation. The reflection of the findings followed the interpretive tradition.

¹ Open Source definition, <https://opensource.org/osd>.

3.1 Data Collection

The data collection was done focusing on stakeholder behaviour of the SE around the open source HIS implementation for the Nutrition Monitoring and Intervention Tracking project. The empirical setting included the State health sector institutions in three districts, two public administrative settings and a central coordinating unit. The multi-method approach included participant observation, interviews, focus group discussions and document analysis [18].

Participant observation was a main approach of gathering data providing an overview of the stakeholder behaviour and the evolution of the SE. The observation were done during the project steering meetings, HIS and non-health IS design and implementation meetings and web and mobile application training sessions. The settings for the participant observation sessions included the central coordination unit, three regional and 17 peripheral health units and two peripheral administrative units. In-situ interviews were conducted during the participant observations to clarify the decisions taken on the SE trajectory and the stakeholder participation.

When interviewing the multi-sector organizational actors, semi-structured interviews and focus group discussions were used. Health managers and non-health sector administrators were the key informants in the semi-structured interviews. They provided rich insights to the process of decision-making during the HIS implementation. This study used data from eight interviews with health managers, 11 interviews with the administrative sector managers, five interviews with the representatives of the funding agency and 12 interviews with FOSS implementers and 3rd party developers. Medical Officer of Health (MOH), Medical Officer – Maternal and Child Health and Public Health Midwives (PHM) were the participants in focus group discussions on mobile app and FOSS HIS back-end at peripheral level. The health sector group discussions included 17 MOH areas. Five group discussions were conducted with participation from the health and public administrative sector actors, funding agency and FOSS implementers. Participant observation and interviews were supplemented by the document reviews for a deeper understanding. The documents analysed comprised of email communications, project steering meeting and evaluation meeting minutes, official letters and policy documents related to the HIS ecosystem.

3.2 Data Analysis

During this study the raw data was recorded as manual field notes at the time of interviews and participant observation sessions, which were later transcribed into complete manuscripts. Interview data was compared and triangulated with other evidence such as participant observations and document analysis. The data analysis follows the interpretive tradition [19]. A basically inductive approach [20] was followed when interpreting field notes to understand the FOSS HIS ecosystem trajectory.

4 Research Findings

In Sri Lanka a nutrition policy was first introduced in 1986. However, the nutritional status of children were not satisfactory although a wide range of programmes from growth monitoring to nutrient supplements had been ongoing for many years.

4.1 Multi-sector Stakeholder Network

Hence, in August 2008, Department of Health appointed a task force to revise the NNP [7]. The committee apprehended the fact that the nutritional well-being of a population is influenced by determinants that cut across the areas of responsibilities of different sectors which extends beyond the scope of the Department of Health. The revised NNP was expected to provide a framework for inter-sectoral coordination in order to accelerate efforts to achieve optimum nutritional status. However, Department of Health alone could not achieve the multi-sector coordination. In this regard, the conventional paper based reporting system was not sufficient to facilitate the required multi-sector coordination to achieve the objectives laid down by the NNP.

In 2013, NNP was revised again and the National Nutrition Secretariat of Sri Lanka (NNS) was established to achieve a better coordination of multi-sector activities prescribed by NNP. The NNS was positioned directly under the Presidential Secretariat of Sri Lanka giving it the capability of inter-departmental coordination. A major task of the NNS was to develop the Nutrition Action Plan targeting the priority areas for action. NNS was entrusted to monitor and evaluate the progress of activities under the Nutrition Action Plan at National, Provincial, District and Divisional levels. Three districts, where malnutrition was prevalent, were selected to launch the pilot project. Under this project, MOH and Divisional Secretariat were the main coordination points for the health and non-health sectors respectively at the lowest administrative level. Field level multi-sector coordination was assigned to the Village Committees, which has the PHM as the focal person to identify nutritionally at-risk households. ‘Grama Niladhari’ (government officer to the village), ‘Samurdhi Niyamaka’ (government appointed social service officer), Agricultural Extension Worker and Development Assistant helped PHM to identify root causes for malnutrition during Village Committee meetings.

4.2 Implementation of the IS

NNS facilitated the implementation of an information system to realize the multi-sector coordination. Initial meetings were coordinated by the NNS and attended by health and non-health sector stakeholders, funding partners and HIS implementers. The open source public health information system framework, District Health Information System² (DHIS2) was used as the HIS back-end. Selection of FOSS was due to several reasons including the encouragement from funding agency for its potential sustainability with

² <https://www.dhis2.org>.

global contribution, satisfying the guidelines of national eHealth policy on software source code ownership and not having a recurrent licensing cost. DHIS2 was customized as per the requirements of the Nutrition Action Plan under the supervision of the NNS and the Department of Health. A significant customization was needed to adapt DHIS2 to cater the specific requirements laid down by Nutrition Action Plan. Sub-components of the IS were shaped by the functionalities prescribed by the Nutrition Action Plan.

Initially the system architecture was designed as a single component. However, there were some concerns among health sector stakeholders, such as, “Health information is too sensitive to be seen by ‘Grama Niladhari’ or ‘Samurdhi’ officer. So, the two systems cannot be a single integrated solution”. Hence, later it was decided to keep the health and non-health components of the information systems separated due to the sensitive nature of health information and the information system was then designed as two separate sub-systems within a single SE. The selected information of the families with malnourished children supposed to be entered to the system by PHMs, who were appointed as the field level data collection operatives. It was agreed to share the data gathered to the HIS component with the non-health sector component only after removing the socially sensitive information. To assure the privacy and confidentiality of health data, only the minimum essential data set required for nutrition interventions were shared with the Village Committee and other non-health sector stakeholders.

The proposed HIS design demanded PHMs to enter data during home visits. This required a portable solution for PHMs instead of the standard web interface of DHIS2. Hence, NNS suggested PHMs to be given a mobile device for field level data collection. However, due to several unique requirements Nutrition Action Plan laid down, the native DHIS2 mobile app was not adequate for this purpose. Further, the DHIS2 mobile app was not fully developed to the potential of its web counterpart at the time of implementing this multi-sector nutrition IS. After several rounds of discussions, it was decided to develop a custom smart phone based mobile app as the field level data collection tool. The mobile app development was an iterative process where prototypes were created and feedback was received for the interfaces from NNS, Department of Health and the funding agency. The DHIS2 web Application Programming Interface (API) was used to communicate between the mobile app and the central server. The mobile app design was shaped by the inputs from the PHMs as well. The coding of the mobile app was outsourced to a third party software development firm by the HIS implementers. 600 smart phones and 70 laptops were provided by United Nations Children’s Fund to pilot the system in the three selected districts. In-service, on-site training programme was conducted in each MOH area for MOHs and PHMs on mobile application and the DHIS2 based data analysis back-end. The pilot was supervised by NNS and the Department of Health. The development of the non-health components was negotiated in parallel to the piloting of HIS component. The non-health sector system was designed to track interventions done by the multi-sector stakeholders. A custom web app for the DHIS2 back end was developed to facilitate easy visualization of the intervention taken by the Village Committee and Divisional Secretariat level coordinators.

We observed that the implementers used to express their concerns about the weak technical documentation during custom component development. “We need support on integrating the custom modules/apps through web API. If support is available, we can

speed up the development. Otherwise, it is a very time-consuming to study the API calls, especially when the API changes rapidly with frequent release cycles [of DHIS2]” was a such concern. We noted that client organization and the funding partners were also questioning about the support implementers get from the FOSS firm. “What would be the support you [implementers] are getting from DHIS2 community? If their support is readily available, we believe that this implementation would be more sustainable” was such a quote made by the funding partner.

5 Analysis and Discussion

In this section we discuss how a SE is being built around an open source HIS implementation. In this study the software framework is the DHIS2 open source HIS framework and the solution element it built was the nutrition monitoring and intervention tracking system. The internal developers are the HIS implementers employed by the NNS and the external developers included core DHIS2 team and the 3rd party software firm who developed the mobile and web components. Domain experts were the MOH and Divisional Secretariats who supervised the nutrition assessment and interventions. The community of users mainly included PHM from health sector and Village Committee members representing non-health sectors.

5.1 Emergence of the SE and Its Composition

Software implementation exercises need to consider the end user requirements as well as the needs of client organization commissioning the software customization [1]. In this study, the most important requirements leading to the inception of the SE were the need for the field level nutrition surveillance and the tracking of the multi-sector nutrition interventions enabling a collaboration across domains. This stakeholder integration in the SE emerged on top of the IS. Otherwise, these actors would have operated with fragmented information flows. According to Hanssen [5], SE emerges through the use of a technology focus, which in this case study was the open source framework, DHIS2. The selection of FOSS as the candidate technology was decided not only by the ability to align with the business requirements, but also the ability to comply with the policy and financial considerations. The scope of the SE is much broader than a single IS project, through the software extensions (e.g. web and mobile apps) provided by the external contributors [2]. The integration of 3rd party components with the core software framework makes the SE to expand beyond the conventional organizational boundaries. In the empirical setting this was evident from the use of custom components, which were integrating multi-sector stakeholders to a single SE.

In application-centric SEs, the aligning of software architecture to the organizational structure also plays a crucial role [2]. This was the case for DHIS2 and the multi-sector nutrition intervention tracking effort as well. FOSS doesn't incorporate special features that are catering only for minor sub-sets of users. Instead, FOSS framework such as DHIS2 aims at generic solution that can be accessed through APIs, on which custom component development may be used to develop special features by

customising and extending generic functionalities. In this context, it was important to simplify the contribution of 3rd party developers. Not having a sufficiently detailed API documentation was a major drawback which delayed FOSS implementers developing 3rd party components.

In additions to its particular technical characteristics, the evolvment and behaviour of the stakeholder network is also a unique feature of the SE [5]. In general, a FOSS SE would comprise core FOSS developers providing a framework for 3rd party developers together with several layers of actors customizing and configuring the software product [5]. Similar features have been demonstrated in this paper, where behaviour and perspectives representing multiple organisations have been interacting and forming the SE. In a SE there will, at any time, typically be a leading or central organizational actor, which is referred to as the central referent organization [5]. The NNS emerged as the central referent organization in this case study. However, towards the later phase of the study, the HIS implementer played this role.

The network organization in this case comprised health sector and non-health sector organizational actors as domain experts and end users. The global DHIS2 implementer community and 3rd party mobile/software developer teams were indirectly involved in the implementation effort through the HIS implementers. As Bosch and Bosch-Sijtsema [2] mentioned, software ecosystems build new dependencies between components and their associated organizations that did not exist earlier. In this case study, the health and the non-health components of the system formed new links and dependencies between health and non-health actors, which were not there before. The overall objective of the project is to deliver coordinated nutritional services. For this objective to be achieved, the 'new' interdependent cooperation between different sectors that have evolved within the SE will need to be further strengthened and sustained.

FOSS implementers and 3rd party solution developers are important players in the FOSS SE [5]. The 3rd party developers are key actors as they are aligning and adapting the generic FOSS solutions to the specific domain needs by developing custom components. However, encouraging the 3rd party developers to contribute back to the FOSS code base is also important. The HISSL implementation team have contributed back to the code base by providing feedback and new requirements, which is as important as 'code' in a literal sense. DHIS2 core developers and developer community have supported 3rd party developers in Sri Lanka to understand the API, which is a key technology enabling integration of 3rd party contribution to the FOSS framework. However, active support is needed from the FOSS firm towards 3rd party developers and implementers in this regard. Evidence of the FOSS firm providing active support to the implementers improved the client organization's trust on the FOSS product as well as on FOSS implementers.

IS projects are in constant negotiation of boundaries within a SE [21]. In a software project, the technical negotiation happens on the boundary between local and global software development networks. For the domain specific negotiations in this case, the inside is the health domain and the outside is the non-health domain. Domain specific negotiations take place at the boundary between health and non-health through the PHM in this case. Over time, the stakeholders experienced the SE and its components were influenced by such negotiations. As a result, new spaces for negotiation, such as the Village Committees, emerge as the organizational structure of the SE is stabilizing.

PHM functions as a domain specific boundary spanning agent [22] in the nutrition intervention domain, freely moving between the health sector and the non-health sector. In the technical space of the SE, another boundary spanning role was noted for the role played by the local DHIS2 implementers. They bridge the gap between FOSS developer community and the 3rd party component developers to whom they outsourced custom component development.

In the long run, uncertainty about whether the 3rd party components will be integrated with the FOSS code repository will also be an important factor in motivating external developers. Third party components with sufficiently generic use cases are potential candidate components to be merged with the FOSS code repository.

6 Conclusion

SE is a new business model which needs the contribution from new empirical domains to be further developed to include domain specific behaviours. Hence, we expect the lessons presented in this paper to help FOSS firms, implementers and clients to better understand the ecosystem building processes and how a sustainable ecosystem in the FOSS HIS domain may be developed. Some findings may be applied to custom software development ecosystems in the health domain and others may be applied to FOSS ecosystem development in general.

In LMIC context, multi-sector SE can be developed to enhance local technical capacity, which would otherwise be impossible to maintain in the State sector. Application-centric FOSS SEs contribute to this aim by providing access to the code repository and developer community. However, the presence of FOSS implementers and 3rd party component developers are essential for a viable FOSS ecosystem to emerge in LMIC contexts. The FOSS firm and domain experts (client organization) alone are not sufficient.

A SE has internal and external stakeholder networks which could be either domain specific or technical. In particular an open source SE needs to have a central FOSS framework and custom components developed that are extending generic open source software functions. It is important to apprehend the role of the FOSS implementers, which a client firm can employ to customize an open source software. Whether internal or external to the client organization, the FOSS implementer has a boundary spanning role bridging the FOSS firm and the 3rd party component developers. Identifying the 'central referent organization' [5], who manage the participation of different stakeholders, was an important step in governing the stakeholder interactions in the SE in our case. However, the role of the central referent organization may be played by different organizations during SE evolution.

The role of 3rd party developers is also noteworthy for a viable SE around an open source implementation. The FOSS SE should maintain a good support for 3rd party developers. These include rich API documentations and a clear path for 3rd party components in the FOSS road map. Similarly, the FOSS firm needs to expose 3rd party FOSS development and implementation channels to prospective FOSS customers.

References

1. Dittrich, Y.: Software engineering beyond the project—sustaining software ecosystems. *Inf. Softw. Technol.* **56**(11), 1436–1456 (2014)
2. Bosch, J., Bosch-Sijtsema, P.: From integration to composition: on the impact of software product lines, global development and ecosystems. *J. Syst. Softw.* **83**(1), 67–76 (2010)
3. Manikas, K., Hansen, K.M.: Software ecosystems—a systematic literature review. *J. Syst. Softw.* **86**(5), 1294–1306 (2013)
4. Kilamo, T., Hammouda, I., Mikkonen, T., Aaltonen, T.: From proprietary to open source—growing an open source ecosystem. *J. Syst. Softw.* **85**(7), 1467–1478 (2012)
5. Hanssen, G.K.: A longitudinal case study of an emerging software ecosystem: implications for practice and theory. *J. Syst. Softw.* **85**(7), 1455–1466 (2012)
6. Pettersson, O., Svensson, M., Gil, D., Andersson, J., Milrad, M.: On the role of software process modeling in software ecosystem design. In: *Proceedings of the Fourth European Conference on Software Architecture: Companion Volume*, pp. 103–110 (2010)
7. Ministry of Healthcare and Nutrition. National Nutrition Policy of Sri Lanka (2010). <https://extranet.who.int/nutrition/gina/en/node/7974>
8. Scacchi, W., Alspaugh, T.A.: Understanding the role of licenses and evolution in open architecture software ecosystems. *J. Syst. Softw.* **85**(7), 1479–1494 (2012)
9. Bosch, J.: From software product lines to software ecosystems. In: *Proceedings of the 13th International Software Product Line Conference*, pp. 111–119. Carnegie Mellon University (2009)
10. Walton, C.: The open source software ecosystem. In: Sierra, C., Augusti, J. (eds.) *IIIA Communications. Institut d'Investigacion en Intel. ligencia Artificial, IIIA, Barcelona* (2002)
11. Câmara, G., Fonseca, F.: Information policies and open source software in developing countries. *J. Am. Soc. Inform. Sci. Technol.* **58**(1), 121–132 (2007)
12. Subramanyam, R., Xia, M.: Free/Libre open source software development in developing and developed countries: a conceptual framework with an exploratory study. *Decis. Support Syst.* **46**(1), 173–186 (2008)
13. Marsan, J., Paré, G., Wybo, M.D.: Has open source software been institutionalized in organizations or not? *Inf. Softw. Technol.* **54**(12), 1308–1316 (2012)
14. Nagy, D., Yassin, A.M., Bhattacharjee, A.: Organizational adoption of open source software: barriers and remedies. *Commun. ACM* **53**(3), 148–151 (2010)
15. Krishnamurthy, S.: An analysis of open source business models (2005). <http://141.2.2.194/downloads/teaching/e-bmw/bazaar.pdf>
16. Creswell, J.W.: *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. Sage, Los Angeles (2013)
17. Yin, R.K.: *Case Study Research: Design and Methods*. Sage, Thousand Oaks (2003)
18. Flick, U.: *An Introduction to Qualitative Research*. Sage, London (2012)
19. Walsham, G.: Doing interpretive research. *Eur. J. Inf. Syst.* **15**(3), 320–330 (2006)
20. Thorne, S., Kirkham, S.R., MacDonald-Emes, J.: Focus on qualitative methods. Interpretive description: a noncategorical qualitative alternative for developing nursing knowledge. *Res. Nurs. Health* **20**(2), 169–177 (1997)
21. Elbanna, A.: Rethinking IS project boundaries in practice: a multiple-projects perspective. *J. Strateg. Inf. Syst.* **19**(1), 39–51 (2010)
22. Perrone, V., Zaheer, A., McEvily, B.: Free to be trusted? Organizational constraints on trust in boundary spanners. *Organ. Sci.* **14**(4), 422–439 (2003)