The Ontology of the OSS Business Model:
An Exploratory Study

Spyridoula Lakka, University of Athens, Greece

Teta Stamati, University of Athens, Greece

Christos Michalakelis, University of Athens, Greece

Dracoulis Martakos, University of Athens, Greece

ABSTRACT

This study focuses on theory building providing a holistic conceptual framework that consists of an ontology based OSS business model and an OSS business model taxonomy. The study extends existing theory in OSS business models and corresponding taxonomies, based on the structured-case methodological approach. An exploratory study is conducted in two research cycles, for the identification, validation, and evaluation of the critical constructs of an OSS business model. Results reveal that OSS business models differ from traditional software business models, having specific features that affect the software value chain, the infrastructure, and the revenue model of an OSS oriented firm.

Keywords: *Business Model, Business Model Taxonomy, Information Systems, Interpretive Approach, Open Source Software, Structured-Case Method*

INTRODUCTION

When Richard Stallman first set the Free Software (FS) definition, FS was considered more as an ideological movement against commercial exploitation of software (Stallman, 2002) stressed that free software was more a matter of liberty rather than price. The recasting of Free Software as Open Source Software (OSS) or Free/Libre Open Source Software (FLOSS), emphasized on the importance of making source code freely available implying that a company can choose to make source code freely available and still serve its own business interests as a for-profit organization. The increasing number of profitable activities around the OSS ecosystem (i.e. open communities, standards, and technologies) proves that OSS is not only an innovative model of production, but also a sustainable business model.

It has matured to a point where there are growing numbers of business solutions delivering real business value today. At the same time, more and more IT and business decision-makers are identifying, pursuing, and succeeding with initiatives that employ elements of that eco-

DOI: 10.4018/jossp.2011010103

system to achieve meaningful immediate and sustained business benefits. From a managerial perspective, there are still risks associated to the OSS adoption, revealing issues of whether, to what extent and when it is best to change a business' strategy towards an 'open source' approach. Related research (Ågerfalk et al., 2005; Goode, 2005; Ven et al., 2008) showed that the lack of strategic planning and clear business model are among the inhibitory factors that shape their decisions towards OSS.

Although a number of researchers have studied the different OSS Business Model (OSS BM) implementations recorded in industry (Daffara et al., 2007; Dahlander, 2007; Fitzgerald, 2006; Koenig, 2004; Krishnamurthy, 2003; Rajala et al., 2006), none of them have considered of its structural elements. The OSS BM domain knowledge is fragmented and the concept is rarely clarified explicitly. Such clarification is therefore required to unify the different points of view into one comprehensive framework providing a common understanding, language, and labeling, so as to leverage our communication in this context and our utilization of the concept.

Towards this gap in the literature, the objective of this paper is to provide with a comprehensive and generic OSS BM framework that explicitly defines its structural elements, describing the deeper structure of what firms adopting an OSS strategy, actually do. The study focuses on knowledge and theory building by providing answers to critical research questions regarding the critical constructs and common characteristics of an OSS BM, as a linkage between empirical data collected and conclusions drawn. The paper reports on the findings of the use of the structured-case approach and proposes a holistic conceptual framework composed of two models; the ontology-based OSS BM and the OSS BM taxonomy, which is derived as a vertical decomposition of the '*Value offered'* structural element. Finally, the opportunities and threats stemming from the different OSS BM implementations are also discussed.

The rest of the paper is organized as follows: First, the theoretical background of the study and the research methodological approach are described. The next two sections report on the main findings of the two conducted research cycles. Finally, the conclusion section discusses the results and concluding remarks obtained from the study.

THEORETICAL BACKGROUND AND METHOD DESCRIPTION

The research focused on the key issues and challenges that affect a holistic OSS framework. In the spirit of the interpretivist school (Hussey et al., 1997; Lee et al., 2003; Myers, 1997; Orlikowksi et al., 1991; Remenyi, 1998; Walsham, 1995), the approach throughout the study was to understand existing OSS models and build a new theory, rather than to test established theories. This was achieved by studying a number of existing theories and OSS perspectives as different theoretical lenses through which a complex phenomenon might be viewed.

The research that has been undertaken proposes theory as a result of interconnected ideas that condense and organize knowledge (Neuman, 1991). The study involves a series of case studies of OSS oriented organizations by means of the structured-case research method (Carroll et al., 2000), which can be widely used to extend knowledge about existing theories in order to actually use them. The structured-case approach provides a focused but flexible methodological approach to the field research process, through outcomes integration allowing theory, knowledge and practice to emerge from the data collected; researchers guidance to follow and ensure accuracy; and ability to record the processes of knowledge and theory-building.

The method attempts to explain, predict and provide understanding, determining the relationships between concepts in order to build a knowledge guide with respect to various issues of OSS modelling. The development of conceptual frameworks namely, CF1, CF2… CFn is used to present the process of obtaining knowledge and theory building where CFn is the latest version of the theory built. The theory

building process is interrelated with practice (Carroll, et al., 2000). Applied research can lead to theory building, which can lead to further field research and theory building. Thus, each research cycle can lead to updates of the existing CF. As part of the hermeneutic circle each new CF expresses the pre-understanding for the next cycle (Gummerson, 1998) following the natural human action of interpretation and world understanding (Carroll et al., 2000).

Essentially, a spiral towards understanding is enacted as current knowledge and theory foundations for yet another research cycle, which will enhance, revise or evaluate the research understanding. This is particularly appropriate for OSS, as it is an area distinguished by rapid changes, which suggests the need for theory and practice to become closely intertwined. The structured-case will enable theory to be developed that will reflect the concerns, problems and issues facing OSS oriented organizations (Carroll et al., 2000).

In the field of business models theory building, there is a diversity of definitions and approaches. Chesbrough and Rosenbloom (2002) emphasize on the connections between technical potential and the realization of economic value, Amit and Zott (2001) describe the design of the transactions of a firm in creating value, Linder and Cantrell (2000) focus on the firm's core logic for creating value, Malone et al. (2006) offer an operational definition and distinguish different types of business models, while Osterwalder (2004), Gordijn (2003) and Morris et al. (2005) emphasize on the model aspect following an ontology- based approach. Osterwalder et al. (2005) classified business models' researchers into three main categories: (1) those that study the business model as an "overarching concept" of all businesses (i.e. the structural elements of a business model); (2) those that describe a number of different abstract types of business models with common characteristics (i.e. taxonomies); and (3) those presenting aspects of a particular real world business model (i.e. case studies). Considering Osterwalder's (2004) ontological approach for business models, the study aims at the identification of the structural parts of an OSS BM and the formation of an "overarching" ontological OSS BM as well as a taxonomy of the different types of OSS BMs.

Research Methodological Approach

In order to identify the structural parts of an OSS BM, two research cycles were applied. At the first cycle, a sample of 100 popular OSS related firms instances is considered as 'pilots' organizations, in order to explore the different possible business models cases. Appendix C presents the complete list of the selected samples and the market sector they occur.

The sample was chosen so as to reflect all three aspects of Information and Communication Technologies (ICT) markets, i.e. software, hardware and services market sectors. Thus, the instances concern sponsored OSS projects, or firms creating value out of OSS projects in terms of services, founded between 1984 and 2008. Projects not perceived by OSS developers as open source are excluded, e.g. Microsoft's 'shared source' projects, or other communities that use OSS development processes for a limited population without public release of intellectual property (Shah, 2006). Sample's instances were chosen according to their popularity in portals devoted to OSS technologies, such as SourceForge.net, Think Geek, LinuxDevices.com, DesktopLinux.com, as well as eWeek, CIOInsight and InfoWorld.

The second research cycle aims to validate, evaluate and further improve the initial findings. The data collection procedure followed the major prescriptions given by most textbooks in doing fieldwork research. A variety of secondary data sources, such as business reports and technical reports for standards and specifications, were used to collect data regarding the development of OSS models. All in all, a number of data sources, were used to derive the findings presented herein. These included workshops, interviews, illustrative materials such as newsletters and other publications of OSS oriented organizations.

A two-day workshop took place with the eighty two participants, experts from the Greek OSS market and Academia. The participants worked together in collecting all the information needed regarding the critical constructs of a holistic OSS framework. Protocols of procedures were defined beforehand in order to guide the group discussion and to document the OSS model scenario elements. Based on the workshops and the online consultation inputs the authors synthesized a set of key factors that are considered as important for the construction the OSS BM.

After the completion of the two-day workshop, short interviews were conducted on a one-to-one basis with the participants in order to stimulate conversation and breakdown any barriers that could otherwise have hindered the knowledge transfer between the interviewer and the interviewee. The authors acted as a neutral medium through which questions and answers were exchanged and therefore endeavoured to eliminate bias. Interviewers' purpose were to obtain the definitely opinion of participants on OSS critical issues. Results are explicitly illustrated on Appendices A and B.

The overall methodological procedure is summarized in Figure 1. As it is shown, a pre-step of the first research cycle, is the construction of an initial conceptual framework CF1. CF1 is based on bibliographic input of previous research in the field of OSS BM (Bonaccorsi et al., 2006; Daffara & Gonzalez-Barahona, 2007; Dahlander, 2007; Fitzgerald, 2006; Ghosh, 2006; Hecker, 1999; Koenig, 2004; Kooths et al., 2003; Krishnamurthy, 2003; Rajala, et al., 2006; Raymond, 1999; Riehle, 2009). Literature concerning the organizational processes in sponsored OSS communities was also included, due to the fact that organization processes are considered as strategic decisions over the implementation of a successful OSS BM (Fleming et al., 2007; O' Mahony et al., 2007; Von Hippel et al., 2003; West et al., 2008). Four commonly cited elements were identified and placed in CF1 (Figure 2) as the main research issues revealed from our analysis, namely *(1)* the kind of OSS license adopted; *(2)* the offering or value of the OSS product and/or service; *(3)* the OSS community; *(4)* organization of production policy. CF1 will be further refined through the methodological process described in Figure 1.

FIRST RESEARCH CYCLE

In the first research cycle, the sample of 100 OSS cases is explored in order to extract information for each of CF1 constructs. The results for each of these entities are presented in the following sections.

OSS Licenses: OSS Licenses are used as a way for protecting the openness of the source code. There is a large number of OSS Licenses which can be classified in three major categories according to the level of restrictions they impose to users (Fitzgerald, 2006; Lerner et al., 2005; Rosen, 2004), as follows:

Reciprocal licenses, that are characterised by the fact that although source code may be modified, any distribution of a binary file must make available all changes to the source and remain under the same license. They are designed to effectively confront the "free riding" problem, i.e. utilisation of publicly created software for the creation of closed source software. Such licenses are the General Public License (GPL) which is the first FOSS license enacted, the Lesser GPL (LGPL), the Affero GPL, which are less strict by permitting linking with non-free modules etc. *Corporate type licenses* contain restrictions "inherited" to derivative products, yet these restrictions mainly aim to ensure that a specific firm retains control of derivative works, i.e. to allow OSS code to be mixed with proprietary, e.g. the Mozilla Public License (MPL), Eclipse Public License (EPL), etc. *Permissive licenses* place no restriction on the use of the code, requiring only a notice of the original copyright in any redistribution in source or binary form. Examples are the MIT License, Berkeley System Distribution (BSD) license, etc.

Figure 1. Methodological procedure for the analysis of OSS Business Models (BM)

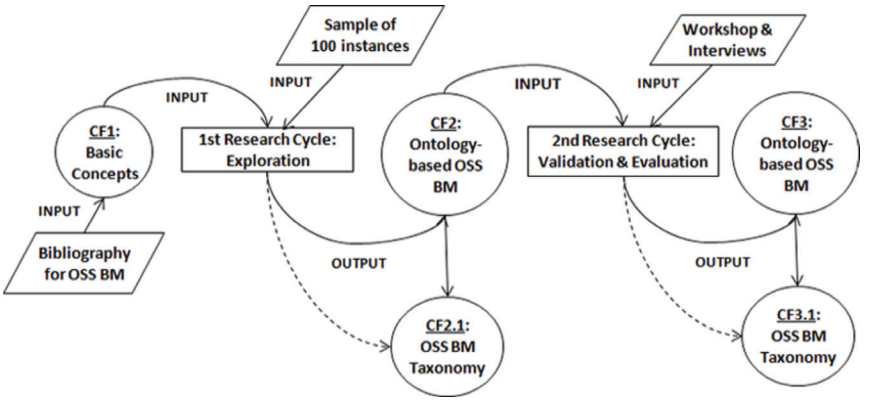
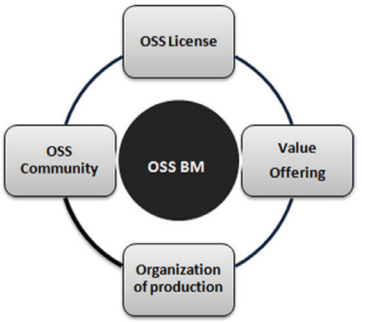


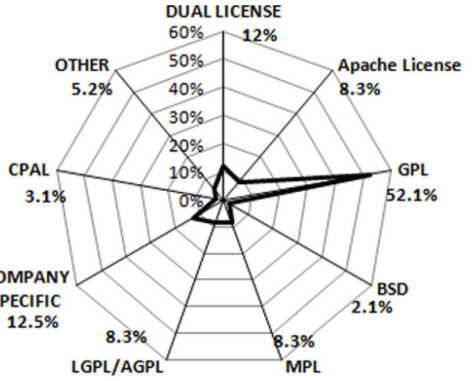
Figure 2. CF1: Basic Concepts for OSS BM



It can be deduced from the aforementioned classification that the choice of the OSS license is closely related to a firm's strategic approach towards the implementation of an OSS BM as it defines the level of risks a firm takes by opening the code to its competitors. Figure (3) presents the kinds of licenses adopted by the sample projects. Most of the instances have related their products to an OSS License[1]. Firms desire to avoid 'free-riding' problems by choosing the GPL (52%) and other reciprocal licenses like AGPL, LGPL, the Apache license, and the Common Public Attribute License (CPAL). These licenses also ensure good relationship with the OSS Community, as they are close to the FLOSS spirit. The second best choice is the Corporate type licenses (20.8% MPL included), which protect firms' Intellectual Property Rights (IPR) from being exploited by third parties. These licenses usually confer the firm's name, which is also a potential marketing strategy. Firms do not take the risk of leveraging permissive licenses (2.1%).

Finally, 12% of the firms in the sample prefer to apply two different licenses over the same product. The *dual license* approach is not an integrated license, or a different license type, but is rather a business strategy where a firm offers free use of open source code, or alternatively offers for a fee commercial distribution rights and a larger set of features for a product. Usually, the one of the two licenses is the GPL license, which prevents third parties from developing improvements that would rival the original software. Then the second license is an ordinary commercial license.

Figure 3. OSS Licenses encountered in the sample



Value Offering: Two clustering levels were applied in the sample instances, firstly according to the market sector they occur and secondly according to their value offerings. At the first level, out of 100 cases, 73 were found in the software sector, 15 in the services sector and 12 are software designed for specific hardware and thus supported by hardware firms (Appendix C).

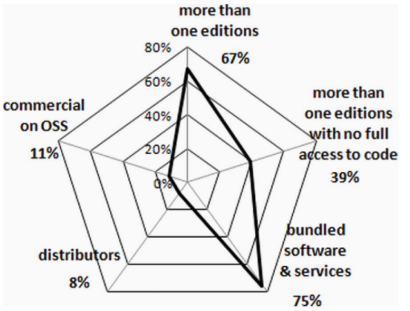
The different cases encountered in the S*oftware sector* are illustrated in *Figure (4a).* The most popular strategy in the sample is the offering of different editions over the same product with additional features and functions. In this case there is the 'Community' edition, with the basic functioning offered free of charge. More features and functions are given in subsequent editions usually named 'Enterprise' and/or 'Professional' and which require some kind of payment. This may be either by annual subscriptions, or more scarcely, by a per unit price. (e.g Alfresco, Opsview, Compiere, Jaspersoft, etc.). Driven by these results, we define as '*Level of openness'* the extent to which a firm allows the customer to access specific parts of the code, as well as features and functions. The associated business model is named *'Added value editions'*. The 67% of the sample follows this strategy. In addition, although the 'Community' editions on all of these projects are offered with full access to code, the subsequent editions do not necessarily convey this feature. In particular, 39% of them were found to keep some part of the code closed on the subsequent editions.

Bundling software with services: There is a high tendency (75% of the sample) to bundle products with services. Services vary from support, documentations, training, integration and migration offered for a particular software product. Pay method is not a per-unit-price, but in forms of subscription contracts. Contracts may present different levels of offerings in terms of the number of services and the duration of the time offered. Such cases are JBoss, Compiere, Alfresco, etc.

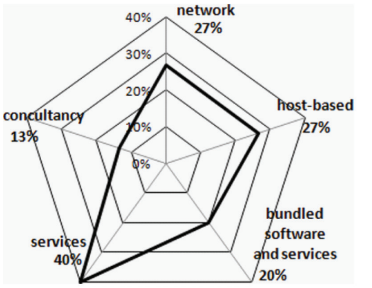
Distributors offer packaged distributions of OSS (usually the Linux operating system). Packages may include media distribution (e.g. CDs), installation upgrade and maintenance services and support. Firms adopting this model typically don't charge for the software but the rest of the package distribution in form of subscriptions. They may also capitalise on complementary software and applications that create on their own and that makes "best fit" with their distributions. This category accounted for 8% of the software cluster (e.g. Ubuntu by Canonical, Fedora and Linux by Red Hat, etc.).

Finally, 11% of the samples are cases offering commercial applications that run on

Figure 4. (a) Offerings at the software sector, (b) Offerings at the services sector



(a)



(b)

an OSS platform or complementary software which adds more features or enhances an OSS product (*Commercial on OSS)*. Examples of such instances are the Nusphere Corporations with the Nusphere PhP tools, Acquia Drupal. Also, the NoMachine company, a division of Medialogic S.p.A., is a Linux system integrator.

The *Services sector*'s offerings are illustrated in *Figure (4b)*. The cluster identifies 5 major categories. More particularly, 13% of the sample offer "*Consultancy"* for the implementation of OSS solutions. The '*Services'* cluster which accounted for 40% of the services sample, included all the ordinary kinds of services met with the commercial software, yet adapted for the OSS case: OSS systems integration, migration from one system to another (i.e. from commercial to OSS), education and training, customization of OSS software, support, information systems outsourcing, remote server management, security and maintenance. In addition, an exclusively OSS related service was identified. This was "certification", which is actually an insurance that an OSS software package complies with a specified set of rules, and is legally liable for such compliance (e.g. OpenLogic). As a subset of the '*Services*' cluster the *'Bundling software and services'* cluster is identified, where service oriented firms may develop OSS and offer it for free, aspiring at attracting customers for their services offerings. In all cases the pay method is based on subscription contracts varying in price according to the number of services and duration of time provided. (e.g. Infrae, Zenoss, Cloud.com, etc.).

The "*Network"* of firms cluster, concerns an association of organizations from different locations around the world, doing custom software and related services in vertical markets. Successful paradigms of OSS network model are Orixo, Zea Partners and Infrae.

Finally, "*Host based service"* cluster creates value in a rather indirect way. It includes companies that use OSS as a cornerstone to their IT platforms for web based services and applications. Firms can reduce implementation costs and/or further customize the OSS platform to their specific needs. Google, eBay, Amazon, application service providers (ASPs) like Cloud. Com, EyeOS, for cloud computing, etc. make heavy use of OSS for delivering services to their customers.

The *Hardware sector* cluster of our sample, identified 8 instances of tools and drivers for specific *hardware manufacturer*s (e.g. software tools for SONY VAIO, drivers for Hewlett Packard printers, etc.) and 4 instances concerning *embedded software* for specific devices, e.g. the popular Android sponsored by the Open Handset Alliance, Denx' Embedded Linux Development Kit(ELDK). *Hardware manufacturers* typically create in-house software for the functioning of their products, such as drivers, configuration tools, etc. As their revenues stem from hardware and not software, writing code is an additional overhead and cost centre. Thus, many hardware

manufacturers release their in-house code as OSS, or financially support OSS communities, so that to gain human resources for software development and maintenance. They also gain in popularity making this strategy an effective marketing practice. *Embedded open source software* is software adjusted for the functioning of embedded devices, i.e. devices processing computing capacity build for a specific purpose (e.g. mobile phones, machine controls). The most used OSS modified for embedded systems is the Linux operating system.

The 'value offering' clustering findings enables the specification of an initial OSS BM taxonomy CF2.1, illustrated in Figure 5. In addition to the 'value offering' clusters, the dual license strategy from the 'licenses' cluster was included as an OSS BM. This is a BM proposed in most of previous research in the field (Fitzgerald, 2006; Ghosh, 2006).

OSS community: All sample cases that develop OSS software have set up a community to interact with potential users and developers. The 'Community' element has a prominent place in the project's website, with considerable space and lots of functioning, revealing that all firms consider their relationship to the community of high importance. In such a community, potential users can find support, documentation, additional code or they can report requests for support and additional features. They can also take part in forums and actively participate with code development. Apparently, the firms devote additional effort and money to invest into a well structured and sufficiently strong community. This is an indication of the importance and necessity of firms to achieve the best partnership with the community, as this actually means a partnership with the users of a firm's product. The community can become a basic element of the firm's infrastructure, as it offers valuable resources of code, of developers and a continuous feedback from users (Lerner et al., 2000).

Feedback from OSS communities enables fast release cycles, which create the conditions for first mover advantages. Moreover, when the OSS community is strong, it can serve as a marketing device for the diffusion of the product in a short period of time, which is an important feature for a market with network effects. Thus, all OSS BMs are community oriented, and firms are seeking the best possible ways of connecting their products to a sufficiently strong community. The latter is supported by a number of researchers such as (Ågerfalk et al., 2005; Ghosh, 2006; Lakhani et al., 2003).

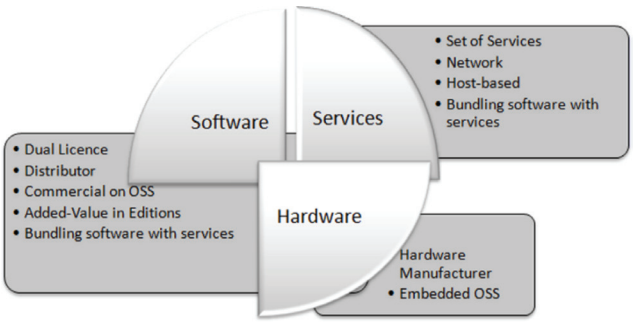
Organization (production and governance): Firms' practices on internal organization and relation to the corresponding OSS communities has been an objective of extensive research, (Baldwin et al., 2006; Capra et al., 2008; Dahlander, 2007; O' Mahony et al., 2007; West et al., 2008).

In terms of *production*, different working practices have been reported in the literature concerning the dispersion of project team's members, the access levels in code for inspection and validation to external participants, different levels of rights in the commit process, the rights for subprojects creation and the ability to observe or follow production processes. In terms of *governance* there are different working practices reported in the relevant literature, concerning the levels of access rights to community developers and formality imposed in processes such as becoming a community member, release authority and project leadership.

A subset of 20 instances of the sample was examined, excluding hardware and services sectors. Moreover, in some cases the relevant information could not be traced. The examination of our sample included the identification of any of the above practices in both aspects of the *production* and *governance* procedures. The results of the analysis are summarized in Table 1.

It can be shown that most restrictions are imposed in the governance procedures, where

Figure 5. CF2.1: OSS BM Taxonomy



firms want to retain control. In the production procedures, most firms prefer to follow a scalable access rights to users policy, however users have enough freedom to take part in the commit process.

CONCEPTUAL FRAMEWORK CF2

The findings of the first research cycle can be summarized as follows:

1. Choice of the *OSS license type*, according to the firms' strategy. Reciprocal licenses ensure good relationship to the community, corporate type license is best for a marketing strategy and finally the dual license strategy.
2. Creation and support of an *OSS Community.* Firms make use of community as it offers valuable resources of code, of developers and a continuous feedback from users, enabling fast release cycles. OSS community can also serve as a marketing device.
3. *Value offering* to potential customers can be traded in any part of the software value chain, i.e. the development, documentation, packaging, marketing and services. For instance, a firm may provide only the packaging of an OSS product which has been developed by an OSS Community, or can provide only technical support, or both. The different kinds of trading are explicitly defined in the proposed OSS BM taxonomy of CF2.1.
4. Value comes also from the '*level of openness*' a firm imposes to a product, a feature that is not part of the proprietary software's value chain. The different 'levels of openness' are implemented with the creation of different editions of the same product, each of which has different value offerings in terms of functioning and code openness.
5. *Revenue models*: Direct, mostly in terms of subscription contracts and indirect, in terms of cost savings and marketing strategies.
6. Configuration of the *organization* (production and governance), with different levels of restrictions.

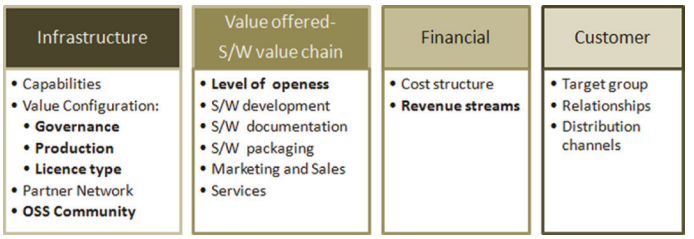
In order to form an 'overarching' OSS BM, we will have to relate these basic OSS constructs to the constructs of an ontology-based business model as defined by Osterwalder (2004). Following this definition, we propose the ontology based OSS BM as the conceptual model CF2 (Figure 6).

Infrastructure comprises of three components: '*Capability*', which outlines the resources as well as the core competencies necessary to execute the company's infrastructure business model. '*Partner Network'*, which portrays the network of cooperative agreements with other companies necessary to efficiently offer and commercialize value and finally *'Value Configuration*' which describes the arrangement of activities and resources. We extend

Table 1. Results of the production and governance procedures analysis

Production	Percentage*	*Governance*	Percentage*
Scalable access levels to code	65%	Formality in processes for becoming a member	15%
Levels of rights in the commit process	65%	Project leadership to members	20%
Ability to observe the production process	75%	Give release authority	45%
** In a sample of 20 projects S/W sector*			

Figure 6. CF2: Ontology-based OSS BM



the 'Infrastructure' construct to include the '*OSS Community*' construct, as explained in *(2)*. We also place '*Licence type*' and activities relevant to the organization model, namely the '*Production*' and the community '*Governance*', under the 'Value Configuration', as explained in *(2), (6)*.

Value offered is mainly the utility of a software product gained by the use, or the kind of service related to that product. We place under this construct the '*Level of openness*' -as defined above- and all the parts of the software value chain. *Value offered* can be vertically decomposed to the CF2.1 OSS BM taxonomy.

Customer: This part of the business model describes the segments of customers a company aims, the various means that a company employs to communicate with its customers and the kind of links a company establishes with its customers. No additional elements for OSS were found for this block.

Financial Aspects: The 'Cost Structure' corresponds to the aggregate monetary consequences of the means employed in the business model. Contrary to the proprietary software, in an OSS BM, *'Revenue models'* do not stem from IPR fees, but as discussed in (5), may have direct and/or indirect profit centres.

SECOND RESEARCH CYCLE

As described in Figure 1, the second research cycle accepts the findings of the previous cycle, i.e. CF2.1, CF2 as inputs to be validated and evaluated with the methodological procedure described. It aims at the enhancing of the conceptual models with possible additional features or rejection of others that might have not been encountered and/or omitted in the sample of the first research cycle.

Particularly, conceptual framework CF2.1 is firstly validated with comparison of results

to the corresponding literature (Appendix A). Although validation was affirmative for all business models found in the sample, some differences were also revealed. Firstly, the *'Added value editions'* business model, is a new OSS BM proposed and was not found cited in the corresponding literature. Secondly, three more OSS BM was identified in the literature, which had not been encountered in our sample. These are '*Ancilliary market*' model, i.e. the capitalization of OSS related products, other than software, such as books, or other publications about OSS, and other physical items associated with OSS (e.g. O'Reilly publishing house.) and two indirect revenue models, i.e. the leverage of OSS as a *Marketing strategy* and as a means of *Cost savings for R&D*. The last two startegies can be applied by all three market sectors of ICT. The '*Ancilliary market*' business model is placed under the hardware sector cluster in accordance to the literature findings.

The above taxonomy CF2.1 including the OSS BM that turned up in the validation process, was further evaluated by the workshop and Interviews, as described. Most of the respondents (72%) were aware of 9 out of the 13 OSS business models listed, although most of them (65%) characterized them as 'business practices' or 'strategies' and not 'models'. All of the models of CF2.1 have been identified by the respondents, with the minimum occurrence '*Ancilliary market*', with 28% and maximum occurrences the '*Added value editions*' and *'Distributor*' with 94%. Indirect OSS BM that stressed the value of OSS as a marketing policy to impose a *'Brand name'* (93%), and '*R&D cost savings'* (94%) were also identified. To the question 'what is their opinion about the advantages and disadvantages' of each of the models, there was a convergence of the opinions, the most cited of which are presented in Appendix A. Finally, there were reported interelations between sector clusters, i.e. the '*Bundling of software and services*' model, which is identified as a firm's practice in both software and services sectors and the '*Bundled OSS with a hardware*' resulting in a system of a much lower price. Taking into account these findings CF2.1 is refined to the CF3.1 OSS BM taxonomy (Figure 7).

As there is no previous attempt for the formation of an OSS ontology- based model and thus no relative literature for validation, the CF2 was only evaluated based on the workshop and Interview responses. The ontological approach of the OSS BM creation was explained to the respondents and they were asked to comment on the level of adequacy of the proposed constructs, i.e. weather these constructs should exist in the model, and if yes, whether they had been placed correctly. The results are analytically presented in Appendix B. The majority of the respondents agreed with the adequacy of the existence of these elements as OSS BM constructs. A small percentage found inadequacy of the constructs in *'Level of openness'* (18%), *'Governance'* (11%), while others were not sure of the adequacy of the *'Level of openness'*(5%) *and 'Production*' (12%).

For the placement evaluation, the process revealed a construct 'mismatch'. More particularly 79% of the respondents believed that *'License type'* should be under *'Value offered'* and not *'Value configuration'* construct. That was a correct output, as license type is more closely connected to the software value chain, than to the firm's infrastructure. For the rest of the constructs, the majority of the respondents agreed with their placement in the model. A small number were not sure with the placement of '*Level of openness*'(15%), and *'Production*' (12%). Finally, 20% of the respondents have suggested new constructs and their placements, which can be explored in a future research cycle. Taking into account these findings CF2 is revised to the CF3 ontology-based BM (Figure 8).

CONCLUSION

Following a specific methodological approach based on theory and experience about the OSS models, the research proceeds to propose an effective holistic framework for the OSS BMs that considers various parameters. It focuses on two main aspects of the business models literature,

Figure 7. CF3.1: OSS BM taxonomy

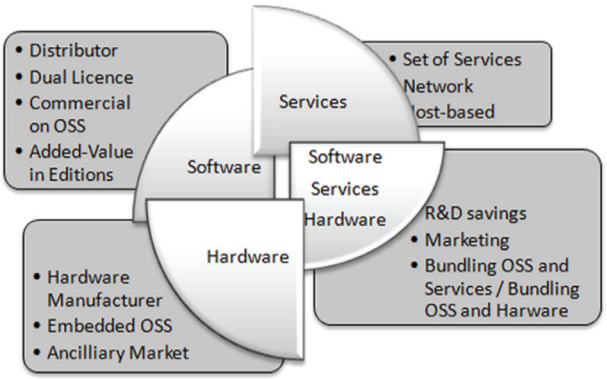
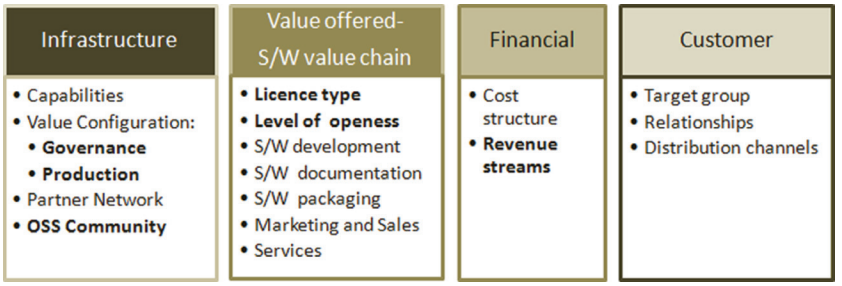


Figure 8. CF3: Ontology-based OSS BM



namely the formation of an ontology-based model that applies to OSS oriented firms and a taxonomy of the existing OSS BMs.

The outcomes from the data analysis of the case studies demonstrate that OSS BMs are influenced by a combination of technological and business elements. The authors follow the classification of OSS terminology, grouping the findings and allowing specific concepts to emerge within such groupings. The concepts revealed the structural elements of an ontology-based OSS BM. Furthermore, a vertical decomposition of the '*Value offered'* construct of the ontological OSS BM enabled the formation of a taxonomy for the different OSS BMs according to the market sector they occur. The taxonomy introduces a new OSS BM identified in the exploratory study, namely the *'Added-value editions*'.

Overall, the holistic framework provides with insights on the critical elements of an OSS BM ontology, an explicit taxonomy regarding the different BM implementations and their corresponding opportunities and threats. As OSS has been highly diffused over the last years, the research findings can become useful inputs for both researchers and practitioners. For researchers it can become the basis for building a common ontological OSS BM, clarifying and unifying the ambiguous constructs, elements and characteristics of the different OSS BM implementations. Also, the proposed taxonomy is not meant to be exhaustive or definitive as OSS BMs continue to evolve and new interesting variations can be expected in the future. As there isn't a previous framework of the kind, this study aspires to create an efficient basis for future research in the field.

However, even in its current form, the framework can also become a useful tool for managers and decision makers that would think and anticipate the risks of adopting a new OSS BM, or adapting their existing BM towards OSS. The tool summarizes the architecture insights, structural elements of an OSS BM, the different implementations and the opportunities and threats of already practiced OSS BM in the market.

Revealing the limitation of the study, the number of our datasets and potential sample's inequality of proportion of each market sectors' instances are stressed. As a consequence, some of the results should be further improved in future research cycles. Further research may also focus on the identification of factors influencing the successful implementation of OSS BMs.

REFERENCES

Ågerfalk, P. J., Deverell, A., Fitzgerald, B., & Morgan, L. (2005). Assessing the role of open source software in the European secondary software sector: A voice from industry. In *Proceedings of the First International Conference on Open Source Systems*, Genova, Italy.

Amit, R., & Zott, C. (2001). Value creation in e-business. *Strategic Management Journal*, *22*(6-7), 493–520. doi:10.1002/smj.187

Baldwin, C. Y., & Clark, K. B. (2006). The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Management Science*, *52*(7), 1116–1127. doi:10.1287/mnsc.1060.0546

Bonaccorsi, A., Rossi, C., & Giannangeli, S. (2006). Adaptive entry strategies under dominant standards: Hybrid business models in the Open Source Software Industry. *Management Science*, *52*, 1085–1098. doi:10.1287/mnsc.1060.0547

Capra, E., & Wasserman, A. I. (2008). A framework for evaluating managerial styles in open source projects. *IFIP International Federation for Information Processing*, *275*, 1–14. doi:10.1007/978-0-387-09684-1_1

Carroll, J., & Swatman, P. (2000). Structured-case: A methodological framework for building theory in information systems research. *European Journal of Information Systems*, *9*, 235–242. doi:10.1057/palgrave/ejis/3000374

Chesbrough, H., & Rosenbloom, R. S. (2002). The role of the business model in capturing value from innovation: Evidence from Xerox Corporation's technology spin-off companies. *Industrial and Corporate Change*, *11*(3), 529–555. doi:10.1093/icc/11.3.529

Daffara, C., & Gonzalez-Barahona, J. M. (2007). *Guide for SMEs (No. Deliverable D8.1.1)*. Retrieved from http://www.flossmetrics.org/sections/deliverables/docs/deliverables/WP8/D8.1.1-SMEs_Guide.pdf

Dahlander, L. (2007). Penguin in a new suit: A tale of how de novo entrants emerged to harness free and open source software communities. *Industrial and Corporate Change*, *16*(5), 913–943. doi:10.1093/icc/dtm026

Fitzgerald, B. (2006). The transformation of open source software. *Management Information Systems Quarterly*, *30*(3), 587–598.

Fleming, L., & David, M. W. (2007). Brokerage, boundary spanning and leadership in open innovation communities. *Organization Science*, *18*(2), 165–180. doi:10.1287/orsc.1060.0242

Ghosh, R. A. (2006). *Economic impact of open source software on innovation and the competitiveness of the information and communication technologies (ICT) sector in the EU*. Maastricht, The Netherlands: United Nations University (UNU) MERIT.

Goode, S. (2005). Something for nothing: Management rejection of open source software in Australia's top firms. *Information & Management*, *42*(5), 669–681. doi:10.1016/j.im.2004.01.011

Gordijn, J., & Akkermans, H. (2003). Value-based requirements engineering: Exploring innovative e-commerce ideas. *Requirements Engineering*, *8*(2), 114–134. doi:10.1007/s00766-003-0169-x

Gruber, M., & Henkel, J. (2004). New ventures based on open innovation –an empirical analysis of start-up firms in embedded Linux. *International Journal of Technology Management*, *33*(4), 356–372. doi:10.1504/IJTM.2006.009249

Gummerson, E. (1998). *Qualitative methods in management research*. Newbury Park, CA: Sage.

Hecker, F. (1999). Setting up a shop: The business of open source software. *IEEE Software*, *16*(1), 45–51. doi:10.1109/52.744568

Hussey, J., & Hussey, R. (1997). *Business research: A practical guide for undergraduate and postgraduate students*. Basingstoke, UK: Macmillan Business.

Koenig, J. (2004). *Seven open source business strategies for competitive advantage*. Half Moon Bay, CA: Riseforth, Inc.

Kooths, S., Langenfurth, M., & Kalwey, N. (2003). Open-source software: An economic assessment. *MICE Economic Research Studies*, *4*, 59.

Krishnamurthy, S. (2003). An analysis of open source business models . In Feller, J., Fitzgerald, B., Hissam, S. A., & Lakhani, K. (Eds.), *Perspectives on open source and free software*. Cambridge, MA: MIT Press.

Lakhani, K., & Von Hippel, E. (2003). How open source software works: "Free" user-to-user assistance. *Research Policy*, *32*(6), 923–943. doi:10.1016/S0048-7333(02)00095-1

Lee, A., & Baskerville, R. (2003). Generalizing in information systems research. *Information Systems Research*, *14*(3), 221–243. doi:10.1287/isre.14.3.221.16560

Lerner, J., & Tirole, J. (2000). The simple economics of the open source. *The Journal of Industrial Economics*, *52*(2), 197–234.

Lerner, J., & Tirole, J. (2005). The scope of open source licensing. *Journal of Law Economics and Organization*, *21*(1), 20–56. doi:10.1093/jleo/ewi002

Linder, J., & Cantrell, S. (2000). Changing business models: Surveying the landscape. *Accenture*, 1-13.

Malone, T. W., Weill, P., Lai, R. K., D'Urso, V. T., Herman, G., Apel, T. G., & Woerner, S. (2006). *Do some business models perform better than others?* SSRN eLibrary.

Morris, M., Schindehutte, M., & Allen, J. (2005). The entrepreneurs business model: Toward a unified perspective. *Journal of Business Research*, *58*, 725–735. doi:10.1016/j.jbusres.2003.11.001

Myers, M. D. (1997). Qualitative research in information systems. *Management Information Systems Quarterly*, *21*(2), 241–242. doi:10.2307/249422

Neuman, W. L. (1991). *Social research methods: Qualitative and quantitative approaches*. Boston, MA: Allyn and Bacon.

O' Mahony, S., & Ferraro, F. (2007). The emergence of governance in an open source community. *Academy of Management Journal*, *50*(5), 1079–1106. doi:10.5465/AMJ.2007.27169153

Orlikowksi, W. J., & Baroudi, J. (1991). Studying information technology in organizations: Research approaches and assumptions. *Information Systems Research*, *2*(1), 1–28. doi:10.1287/isre.2.1.1

Osterwalder, A. (2004). *The business model ontology - a proposition in a design science approach*. Lausanne, Switzerland: University of Lausanne.

Osterwalder, A., Pigneur, Y., & Tucci, C. L. (2005). Clarifying business models: Origins, present, and future of the concept. *Communications of the Association for Information Systems*, *15*, 2–40.

Rajala, R., Nissila, J., & Westerlund, M. (2006). Determinants of OSS revenue model choices. In *Proceedings of the 14th European Conference of Information Systems*, Gothenburg, Sweden.

Raymond, E. S. (1999). *The magic cauldron*. Sebastopol, CA: O'Reilly Media.

Remenyi, D. (1998). *Doing research in business and management: An introduction to process and method*. Thousand Oaks, CA: Sage.

Riehle, D. (2009). *The commercial OSS business model*. Paper presented at the Americas Conference of Information Systems, San Francisco, CA.

Rosen, L. U. (2004). *Open source licensing: Software freedom and intellectual property law*. Upper Saddle River, NJ: Prentice Hall.

Shah, S. (2006). Motivation, governance and the viability of hybrid forms in open source software development. *Management Science*, *52*(7), 1000–1014. doi:10.1287/mnsc.1060.0553

Stallman, R. (2002). *Free software, free society: Selected essays of Richard M. Stallman*. Boston, MA: GNU Press.

Ven, K., Verelst, J., & Mannaert, H. (2008). Should you adopt open source software. *IEEE Software*, *25*(3), 54–59. doi:10.1109/MS.2008.73

Von Hippel, E., & Von Krogh, G. (2003). Open source software and the "private-collective" innovation model: Issues for organization science. *Organization Science*, *14*(2), 209–223. doi:10.1287/orsc.14.2.209.14992

Walsham, G. (1995). The emergence of interpretivism in IS research. *Information Systems Research*, *6*(4), 376–394. doi:10.1287/isre.6.4.376

West, J., & O' Mahony, S. (2008). The role of participation architecture in growing sponsored open source communities. *Industry and Innovation*, *15*(2), 145–168. doi:10.1080/13662710801970142

Spyridoula LAKKA holds a Degree in Mathematics from the University of Athens (Greece) and an M.Sc. degree in Computer Science from the University of Liverpool (UK). She has a working experience in management position at the Managing Authority of the Greek Ministry of Education and a teaching experience in Information Technology in secondary education and vocational training. She is currently a PhD candidate and is conducting research in the field of Management of Information Systems with a focus on the managerial implications of Open Source Software (OSS). Topics of research interest include OSS business models, factors affecting diffusion of OSS and impact of OSS on software market structures and competition.

Teta STAMATI has obtained a Degree in Computer Science from National and Kapodistrian University of Athens (Greece) in 1998. She also holds an MPhil Degree in Information Systems from University of Manchester (UMIST) (UK), an MBA Degree from Lancaster University Business School (UK) and a Phd in Information Systems from the Informatics and Telecommunication Department from National and Kapodistrian University of Athens (Greece). She has extensive experience in top management positions in leading ICT companies of the Greek and European private sector. She is currently Research Associate at the Department of Informatics and Telecommunications of National and Kapodistrian University of Athens. Her current research interests include information systems, business models, electronic and mobile government, service science, e/m-commerce, SOA and cloud computing.

Christos MICHALAKELIS holds a Mathematics degree from the University of Athens, Greece, an M.Sc. degree in Software Engineering (The University of Liverpool, UK). He also holds an M.Sc. degree in "Administration and Economics of Telecommunication Networks" and a PhD from the University of Athens (Department of Informatics and Telecommunications), in the area of techno-economics and forecasting of new products growth in high technology markets. He has been working for 7 years with the Greek Ministry of Education, as an IT manager. He specializes in demand estimation and forecasting of high technology products, market competition, price modeling and the application of economic theories over the information and telecommunication technologies.

Dracoulis MARTAKOS is an Associate Professor at the Department of Informatics and Telecommunications at the National and Kapodistrian University of Athens. He received his B.Sc. in Physics, M.Sc. in Electronics and Radio Communications and Ph.D. in Real-Time Computing from the same university. He is a consultant to public and private organizations and a project leader in numerous national and international projects. He serves on the Board of the Hellenic Chapter of the Association of Information Systems (AIS) and he is the author or co-author of more than 70 scientific publications and a number of technical reports and studies.

APPENDIX A.

Table A1. Evaluation and validation of CF2.1

Model	Validation/ Reference in Literature	Evaluation/Interview response		
		C1*	C2**	Comments of the respondents relevant to the OSS BM
distributor	(Fitzgerald, 2006), (Krishnamurthy, 2003), (Ghosh, 2006), (Kooths, et al., 2003)	90%	94%	• Linux is used, tested and implemented for years and thus is of proven quality. • Low entry costs for a firm as most of the software is developed within the OSS Community. • High levels of openness enable good relation with the OSS Community, which will continue to enhance the features of the software and release new versions. • Fast releases give the firm a first mover advantage, over commercial firms. • low entry barriers lead to proliferation of versions and high competition, thus is more difficult to establish a position in the market.
dual license	(Fitzgerald, 2006), (Koenig, 2004), (Krishnamurthy, 2003), (Kooths, et al., 2003), (Daffara & Gonzalez-Barahona, 2007), (Dahlander, 2007)	84%	88%	• The GPL version of the software is favoured by the OSS Community. As a result can attract developers and users and create a relative advantage to an unknown commercial product • difficulties in the management between the two license types. • software from external contributions require an explicit author acknowledgement of both licenses.
commercial on OSS	(Daffara & Gonzalez-Barahona, 2007), (Rajala, et al., 2006)	12%	45%	• low entry costs • no OSS license implications/ source code can be closed. • Careful choice of the OSS platform is recommended
Added value editions	no previous work found for this business model	92%	94%	• This is also a marketing strategy, as users get accustomed to the open and free version, thus they are more likely to choose the advanced product edition, if they need to. • Closed parts of source code is not favoured by the OSS community and thus a firm may not establish a good relation with it.
bundled software & services	(Koenig, 2004), (Rajala, et al., 2006), (Daffara & Gonzalez-Barahona, 2007)	95%	93%	• Subscriptions have no license implications. • Strategy favoured by both software and services firms.
services	(Hecker, 1999), (Koenig, 2004), (Rajala, et al., 2006), (Daffara & Gonzalez-Barahona, 2007), (Fitzgerald, 2006), (Ghosh, 2006)	67%	91%	• Services have no OSS license implications • no obligation in revealing their modifications in code development. • low entry costs • Human resources are the most important asset in the services market. With a proper policy, a part of these resources might be found in the OSS communities.
host-based	(Koenig, 2004)	78%	82%	Promising sector because of the high Internet and E-commerce adoption

continued on following page

Table A1. continued

network	(Ghosh, 2006)	14%	32%	• Collaboration nature of OSS, facilitates such business requires trust between actors as well as synchronization costs.
embedded	(Gruber et al., 2004), (Koenig, 2004)	25%	49%	• Value added by proven technological quality switching costs.
hardware manufacturers cited also as *'Widget Frosting'* model	(Hecker, 1999),(Raymond, 1999), (Fitzgerald, 2006), (Koenig, 2004),(Rajala, et al., 2006)	45%	72%	• bundled OSS with a hardware (e.g. server) resulting in a system of a much lower price. Strategy preferred by system manufacturers like IBM and Apple – • ***INDIRECT OSS BM:*** *'bundled software &hardware'*
Marketing cited also as *'Brand enabler'*	(Hecker, 1999),(Rajala, et al., 2006) (Fitzgerald, 2006), (Dahlander, 2007)	93%	93%	• Firms release code as a marketing strategy, so as to prove the quality of their products and create a brand name and consequently a position in the software market, where can easily sell its commercial software.
R&D cost savings	Not mentioned as a stand-alone BM	94%	82%	• Cost savings in experimenting with code reuse and support from OSS Community
Ancillary markets Cited also as *'Accessorizing'*	(Fitzgerald, 2006), (Hecker, 1999), (Raymond, 1999), (Fitzgerald, 2006),etc.	22%	28%	• These products can be books or other publications about OSS, and other physical items associated with OSS.
Note:	**(*) C1:** Percentage of respondents that mentioned this BM, in the questions "*what OSS Business Models are you aware of*", "*what OSS BM would you suggest?*". Respondents may have mentioned more than two OSS BM. **(**)C2:** Percentage of respondents that were aware of this OSS BM.			

APPENDIX B.

Table A2. Evaluation of CF2

Construct:	should exist in the model			propose relevant construct		is at the correct place			propose new place/ other comments	
	Yes	No	Not Sure			Yes	No	Not Sure		
Governance	83%	11%	6%			89%	2%	9%	Under 'OSS Community'	6%
									At no place	4%
Production	84%	4%	12%	Development Model (DM):	7%	88%	0%	12%	(DM) under 'Value Configuration'	7%
				Modularity Level (ML)	5%				(ML) under 'value offering'	5%
OSS Community	98%	0%	2%			93%	0%	7%	Under 'Partner Network'	4%
Licence type	100%	0%	0%			12%	82%	6%	Under 'Value Offered'	79%
									At no place	5%
Level of openness	77%	18%	5%	code access	7%	82%	4%	15%	At no place	18%
S/W development	88%	2%	10%			88%	2%	10%	At no place	2%
S/W documentation	88%	2%	10%			88%	2%	10%	At no place	2%
S/W packaging	88%	2%	10%			88%	2%	10%	At no place	2%
Marketing &Sales	88%	2%	10%			88%	2%	10%	At no place	2%
Services	88%	2%	10%			88%	2%	10%	At no place	2%
Revenue streams	100%	0%	0%			100%	0%	0%		0%

Note: Results summarize both the workshop and interview responses

APPENDIX C.

Table A3. List of OSS projects

	Company/OSS Project*	Sector		Company/OSS Project	Sector
1	1bizcom/bizcom	H/W	51	Openflows Networks ltd	Services
2	Acquia	Services	52	Openlogic	Services
3	Adaptive Planning	S/W	53	Openmoko/FreeRunner	H/W
4	Alfresco	S/W	54	OpenTerracotta	S/W
5	Alterpoint	S/W	55	Open-Xchange	S/W
6	Apache Foundation/Celtix/Apache CFX	S/W	56	Opsera/Opsview	S/W
7	Apache Software Foundation/OfBiz	S/W	57	Optaros	Services
8	Apple/Darwin	S/W	58	ORACLE/VirtualBox	S/W
9	Black Duck Software	Services	59	Orixo	Services
10	Canonical/Ubuntu	S/W	60	OSAF Chandler	S/W
11	CentraView	S/W	61	Pentaho/ Pentaho BI	S/W
12	CiviCRM	S/W	62	Progress S/W Corporation/Atrix	S/W
13	CleverSafe/Accesser	S/W	63	Real Networks/Helix	S/W
14	Cloud.com	Services	64	RedHat/ Linux	S/W
15	Colosa Inc./Process Maker BPM	S/W	65	RedHat/Fedora	S/W
16	Compiere	S/W	66	Redhat/Jboss	S/W
17	Denx/Embedded Linux Development Kit (ELDK)	H/W	67	rPath/Linux	S/W
18	EmuSoftware/Netdirector	S/W	68	Scalix	S/W
19	EnterpriseDB/Postgres Plus Standard Server	S/W	69	Sendmail	S/W
20	Exadel/JavaFX plugin	S/W	70	Sleepycat/Berkley DB	S/W
21	EyeOS	S/W	71	Smoothwall/Smoothwall Firewall	S/W
22	Funambol	S/W	72	Sonatype	Services
23	GreenPlum	S/W	73	Sony/ 'Sony Controls' for SonyVAIO	H/W
24	GroundWork	S/W	74	Sony/ 'Sony Vaio FX Library'	H/W
25	Hewlett Packard/ 'HP Linux Imaging and Printing'	H/W	75	Sony/ 'ksblc' for SonyVAIO	H/W
26	Hewlett Packard/ 'XPMap'	H/W	76	Sourcefire (SNORT)	S/W
27	Hewlett Packard/ 'Check_hp_print'	H/W	77	Sourcelabs/SWIK.net	Services
28	Hyperic/Hyperic Application & System Monitoring	S/W	78	SourceSense	Services
29	IBM/Eclipse	S/W	79	Splunk	S/W
30	IBM/Jikes	S/W	80	SSLExplorer	S/W

continued on following page

Table A3. continued

31	Infrae	Services	81	SugarCRM	S/W
32	Jasper wireless	H/W	82	SUN/ORACLE/ OpenOffice	S/W
33	Jbilling	S/W	83	SUN/ORACLE/Glassfish	S/W
34	Jitterbit	S/W	84	SUN/ORACLE/Netbeans	S/W
35	KnowledgeTree	S/W	85	Symbiot/OpenSIMS	S/W
36	Lustre	S/W	86	Talend/ Open Studio.	S/W
37	ManyOne networks website	Services	87	TenderSystem	S/W
38	Mindquarry	S/W	88	Tetrain	Services
39	Mirth	S/W	89	UltimateEMR	S/W
40	MuleSource/Mule ESB	S/W	90	VirtualBox	S/W
41	Mysql	S/W	91	vTiger/vTiger CRM	S/W
42	Netscape/Mozilla	S/W	92	Vyatta	S/W
43	NightLabs GmbH/ Jfire	S/W	93	WSO2	S/W
44	NoMachine NX	S/W	94	XenSource (Xen)	H/W
45	Novell/ SUSE Linux	S/W	95	xTuple Norfolk USA	S/W
46	NuSphere Corp./Nusphere PhP Tools	S/W	96	Zea partners	Services
47	Open Handset Alliance/ Android	H/W	97	Zend (PHP)	S/W
48	OpenBravo/ OpenBravo ERP	S/W	98	Zenoss	Services
49	OpenClovis/ OpenClovis	S/W	99	Zimbra	S/W
50	OpenEMM	S/W	100	Zope/ERP5	S/W

* The name of the company is omitted when it coincides with the name of the OSS project.

1 Eight instances don't use any license, as they are services oriented firms. Although the sample consists of 15 services oriented firms, seven of them do produce some kind of software under the GPL.