

An Analysis of the Value that Open Source Contributes to Business Models

Neeshal MUNGA¹, Thomas FOGWILL²

¹*Meraka Institute, CSIR Site, Bldg 43, Meiring Naude Road, Pretoria, 0001, South Africa
Tel: + 27 12 841 3747, Fax: + 27 12 8414720, Email: nmunga@csir.co.za*

²*Meraka Institute, CSIR Site, Bldg 43, Meiring Naude Road, Pretoria, 0001, South Africa
Tel: + 27 12 841 3155, Fax: + 27 12 8414720, Email: tfogwill@csir.co.za*

Abstract: In the constantly turbulent and dynamic environments in which businesses today function, one has to be able to adapt and change one's overall strategy in order to remain competitive. Business models, when carefully constructed can be a valuable tool for firms. In addition to focussing a firm's activities on crucial areas from which they gain the most value, they play an important role in innovation. Innovation is the key to competitiveness and growth. The phenomenon of open source software(OSS) is gaining momentum and attracting much interest in the business world. Firms can take advantage of this situation by combining the power of business models with the opportunities surrounding OSS. In this paper we develop a business model framework to analyse the different OSS strategies available and provide clarity around these concepts.

Keywords: open source software, business model, framework, strategy

1. Introduction

Recent acquisitions of open source firms (Xen Source for \$500 million; Zimbra for \$350 million; Sleepycat for an undisclosed amount and JBoss for \$350 million [1]) indicate the growing significance of open source software (OSS) in today's markets and the increasing probability that the software sector is reaching a commoditisation stage, where the high profits of traditional software companies are threatened [2]. OSS represents the antitheses of a proprietary software strategy. Rather than using formal intellectual property rights (IPR) protection to set boundaries between vendors and their competitors and customers; open source enlists all as collaborators, maximizing adoption throughout the value chain but minimizing the options for appropriating rents from the software [3]. While the dinosaurs of the software production companies prepare to push the panic button, innovative entrepreneurs recognise the inherent opportunity.

The adoption of OSS by companies is testament to the fact that it provides benefits. However, an important question arises: how does it affect the bottom line? OSS by itself does not generate profits for a company. Rather, it is part of a process that takes advantage of the various components at play in the business environment. As Haruvy et. al.[4] state, "the decision to embrace the OS paradigm and the formulation of related strategies, such as price, quality and hiring critically, depend on the business model that is used by the firm to generate revenues for its products." It is ultimately the development of this business model, and how it utilises OSS, that will determine the success of the firm.

The business model has been hailed in the literature as the reason why certain endeavours succeeded and a good model is extremely valuable to the firm. Goth [5] explains that "JBoss's growth is based on a business model that combines software that developers liked for functionality, a license that partners like for its flexibility, and revenue

that customers like for its cost-effectiveness. It's emphasis on a subscription model rather than a consulting model kept costs low while ensuring customers got the latest updates and tools.“ Yet OS is not a business model. There remains confusion regarding OS and how it is valuable in business strategy. Business models themselves are ambiguous in terms of what exactly they provide and what their function is.

1.1 Objectives and Methodology

It has been pointed out that OSS supports national development goals in developing countries. It accomplishes this in four key ways: by reducing public spending on ICT and capital expatriation (OSS reduces total cost of ownership (TCO)), increasing real access to ICT reducing the digital divide, increasing economic development, and ensuring national security [6].

The aim of this paper is to provide clarity around the concepts of OSS and OS business models. Therefore the rest of the paper is structured as follows. The background around OSS and business models are explored in the section 2. Section 3 synthesises the literature to develop a taxonomy of the general OS business strategies and a framework for the analysis of business models. Section 4 uses this framework to determine the impact of OS strategies on business models. Finally, the work is concluded in Section 5.

2. Background and context

2.1 Open Source Software

OSS is software that is released with its source code under a licence approved by the Open Source Initiative (OSI). The two most prominent licenses are the Gnu General Public License (GPL) and the Free BSD license.

In 1991, Finnish student, Linus Torvalds, publicly released the first version of Linux, igniting the OSS movement. The subsequent success [7] of Linux highlights the key driving force behind the OSS philosophy, namely “community”, where the efforts of any number of volunteers collaborating over the internet are combined and coordinated. Ensmenger [8] refers to this counter-intuitive arrangement as a “kind of Rorschach blot in which everyone sees what they are already looking for. ”

OSS is synonymous with Free/Libre Software (FLOSS)(FOSS). OSS is not free when viewed from a total cost of ownership perspective. The “free” associated with OSS refers to the freedom of the user, who can run the program for any purpose, redistribute, probe, adapt, learn from, and customise the software to suit their needs, and to release improvements to others for the greater benefit of the community.

OSS, by its unique nature, has had a significant impact on the “Information Revolution”. The nature of OSS may be evolving, as described in Fitzgerald [9], but its impact has produced both hope and fear amongst various sectors. According to Bonaccorsi et. al.[10] OS is potentially disruptive of competitive equilibria in the software industry and considered to be a “radical innovation” in the way that software is produced and distributed. It is described as “creative destruction in action”[11]. While making considerable changes to the proprietary software landscape, OSS has become big business in itself, generating huge premiums. Numerous models and strategies have emerged [12][13][14][15], describing how companies can gain returns from OSS, each of these with a unique blend of customer, product, and production and distribution mechanism.

2.2 Business Models

Technology plays a significant role in creating value for an organisation, but it needs to be leveraged appropriately. A mediocre technological innovation coupled with a well thought out strategy can provide more returns than a ground breaking technological innovation with a mediocre strategy or business plan. As stated by Hemphill [16], “to extract economic value from the potential of technological innovation, a firm needs to identify an appropriate business model.”

The term “business model” gained notoriety during the dot-com era, and since then various definitions have been discussed. However, no real consensus has emerged [17][18] [19]. Alt & Zimmerman[20] conceive a business model as “an architecture for the product, service and information flows”. It is described as a blend of three streams. The value stream is concerned with the value proposition for buyers, sellers and market makers. The revenue stream identifies how the organisations will earn revenue, and the logistics stream details how supply chain issues will affect the organisations involved [21]. Mitchell & Bruckner [22] describe it as the combination of “who”, “what”, “where”, “when”, “how” and “how much” an organisation uses to provide its goods and services and develop resources to continue its efforts .

Various authors distinguish between a business model and strategy, and some see it as the missing link between strategy and operations in exploring entrepreneurial opportunities, justifying the need for it to be integrated into both value creation and strategy concepts [23]. Wu regards it as general vision or strategy, an abstraction of business, which is different from a business method or specific way of doing business. Its provides value as a planning tool, focussing attention on how all the elements fit into a working whole [24]. While the business model facilitates testing, analysis and validation of a firm's strategic choices, it is not itself a strategy. Choices are made after identifying relevant strategic areas and their options [18].

Combining the work of several authors, Alt & Zimmerman [20] distinguish 6 generic elements of the business model, namely the Mission, Structure, Processes, Revenues, Legal Issues and Technology. Essentially the functions of the business model are to: articulate the value proposition; identify a market segment and specify the revenue generation mechanism for the firm; define the structure of the value chain within the firm required to create and distribute the offering, and determine the complementary assets needed to support this; estimate the cost structure and profit potential of the offering, given the value proposition and chosen value chain structure; describe the position of the firm within the value network linking suppliers and customers, including identification of potential complementers and competitors; formulate the competitive strategy by which the innovating firm will gain and hold advantage over rivals [25].

A clearly stated and understood model is considered to be a prerequisite for success, but Keen & Quareshi [19] emphasise that ultimate success or failure rests on the capability of the firm to customise both the model and follow-on strategy to the dynamics of the market. The business model focusses the firms activities on crucial areas, and plays an important role in innovation, which is considered to be a key to competitive advantage. Mitchell & Bruckner [22] describe business model innovation as business model replacements that provide product or service offerings to customers and end users that were not previously available, as well as the process of developing these novel replacements.

Flexibility and adaptability are increasingly important in today's complex and constantly evolving environment, as is emphasized by the successes and failures of firms. The business model focusses the firms activities on crucial areas, and plays an important role in innovation, crucial to the success of the business. Therefore a model whose assumptions are transparent is more easily reviewed than one with limited linkages between

its elements, allowing for greater flexibility and adaptability [22] [25] [26].

2.3 Open Source Business Models

Various types of business models are discussed in the literature. Rappa, using the customer relationship as the primary dimension, defines the brokerage model, information intermediary model, merchant model, manufacturer direct model, affiliate model, community model, subscription model, and utility and hybrid models as categories of business models in technology [27]. Recent interest and growth has also led to a number of models defined and described for OSS, which view it as more than just a development strategy for software companies. The most influential of these includes Hecker's [13] classification, which is based on Raymond's [12] work, as well as Koenig's [14] taxonomy of possible OS strategies. More recently, after analysing the models of 120 different OS companies, Daffara [28] groups them into 6 main clusters. OS models have a number of similarities with Tapscott's [20] "business webs" (business innovation models), which are "inventing new value propositions, transforming the rules of competition, and mobilising people and resources to unprecedented levels of performance."

Fink [29] distinguishes between outbound and inbound OS dynamics. Outbound OS refers to the open-sourcing of proprietary software. Inbound OS refers to integrating OSS as part of a product or using it within an internal IT project.

Following the outbound approach could: de-value a competitor's product; make pervasive a complementary technology; decrease development costs; enable hardware sales; make services and support more readily available; accelerate the process of exiting a particular business category; and document prior art without having to apply for a patent.

Following the inbound approach could: leverage existing OS technologies in your product; drive one's product forward; focus resources on higher value-added capabilities to generate more revenue; allow selling of OS software. One should avoid bringing OSS into one's organisation when: the strategic direction of an OS project differs from one's business goals; one's technical team does not agree with its use; one needs absolute release control over time-to-market and a component has upcoming features critical to one's needs [29].

Hecker's [13] 8 classical business models for OSS include: the service support seller; loss-leader (OSS is used to strengthen the vendor brand, to improve its commercial products, and to raise familiarity with the total product line); widget frosting (hardware vendors who release drive software as OS to increase the base of developers); accessorising (revenue through the sale of books, computer hardware and other physical merchandise associated with and supportive of open-source software); service enablers (OS software created to allow access to revenue-generating on-line services); "sell it, free it" (existing commercial products are released as OS when the benefit of doing so outweighs the software license revenue they produce); brand licensing (software is released as OS, but the trademarks and IP are retained to generate revenue); software franchising (allowing other companies to do associated business, using your brand and trademarks).

Similarly, Koenig's [14] taxonomy includes the following strategies: optimisation (leverage commoditised technology by adding layers of value to it); dual licensing; subscription-based service provision; consulting services; patronage (contribute to OSS to place the business on a higher level of the software stack, or to eliminate competition by commoditising a particular layer); hosted and embedded (internal use of OSS as a hosting platform).

Daffara [28] identifies the following 6 clusters: twin licensing; split OSS / Commercial licensing; badgeware (the same as Hecker's Branding Model); product specialists (companies with specialist knowledge about an OS product generate revenue from services); platform providers (providers of technology platforms); selection and consulting.

It should be noted that most often business models are grouped into categories based on the licensing involved, viewing the company as either a software distributor, producer, or service provider [31][32]. In reality, however, firms can utilise a combination of strategies and models, and thus these models are not complete business models but rather strategies for gaining value from OS. Weber [33] claims that these are only ideal types and that real companies need to combine business models in their business.

3. The Framework

While OSS is by definition both software and a tangible product, from a business model perspective value from it is not gained in the traditional sense. OSS is not generally sold for a price, and does not follow standard economics where demand influences cost. The business strategies described above demonstrate that there is much ambiguity in how value is appropriated from OS for a business. Much of this is due to the fact that the models cited are not business models per se, but strategies that a company may apply in leveraging OS within their respective business models. Using Dahlander's framework for analysing the means of appropriating returns from OSS [34], and the business strategies described above, a taxonomy of strategies is proposed that can be applied to a firm's OSS business model.

3.1 A Classification of Open Source Business Strategies

The strategies are grouped into two main sections, similar to Dahlander [34]: Product Related and Service and Support Related. The two main categories are further separated by whether OSS is provided (sold, developed or implemented), or whether the existence of OSS provides advantages to the firm in terms of the firm's existing core competence. These strategies, in practise, are often complemented by other strategies and are not mutually exclusive.

Product related strategies include:

- Platform providers, who provide selection, support, integration and services on a set of OS technologies. This can include: the optimisation strategy, where the platform is the commodity on which value is built; brand licensing where the company builds up a brand around the platform; or franchising. It may also include the loss leader strategy, where components of the platform are released as OS, opening closed markets or jump-starting new products. Building a strong brand and reputation is important as revenue is generated from other related components or services. The patronage strategy may also be used.
- Niche and Speciality Providers focus on a specific market segment, providing either products geared towards that market or services related to that market. Services can include training, consulting, implementation, etc. These companies can employ one or more of the following approaches and strategies: dual or twin licensing; subscription; the "sell it free it" strategy; or the patronage strategy (where complementary products and services are built on the value generated by the OS product).
- Complementary product providers - most companies utilising OS in their business strategy fall into this category. The products are often closely related to a specific OS product, and may well be physical tangible products, such as hardware and books. It may also include software built around a popular OS application. This strategy incorporates the widget frosting and accessorising strategies, as well as the loss leader, optimisation and embedded strategies.

Service Related strategies can be employed separately or in combination, depending on the expertise and resources available within the firm. They include:

- Service and Support Provision, either by the company that produces the OS product or some other company, usually involves some model of subscription for services. It is a

model that can be used in conjunction with a product related model, and can be utilised by firms who are platform experts or niche specialists.

- System Implementation, Selection and Consulting focus on customisation and how OS products can be implemented, installed and configured within the client's environment. In this strategy OS is essentially a “service enabler”.
- Training or education can be a service or even a product (training courses and associated material) that can be used as a strategy alone or in conjunction with any of the above strategies. It can be closely associated with franchising, badge-ware and accessorising.

3.1 The Business Model Analysis Framework

Various frameworks have been developed in the literature [23][18][2] to analyse various business models. Notably, Rajala et. al.'s [35] conceptual framework for analysing software business models includes the following main elements: Product Strategy; Revenue Logic; Distribution Model; and Service and Implementation Model. Morris et. al. [17] describe a 6 component framework for characterising business models regardless of venture type, focusing on: how value is created; for whom value is created; the firm's internal source of advantage; how the firm is positioned in the market place; how it makes money; and what its ambitions are.

While OS is software, value from it is not derived in the same way as with commercial software. Therefore Rajala's framework is only partially suited for our analysis. Morris's framework expands on that of Rajala, but it does not take into account the unique characteristics of software and its impact on business. We propose instead the following framework combining their work to evaluate the impact and influence of OS business strategies on a firm's business model. The components of the framework are:

- The Value Offering – the product/service offering and how it creates value for the firm and to whom this value is applicable. It includes the firm's core competence, and those competencies around which an advantage is built.
- The Market – the firm's position in the value chain, whom it creates value for and how it can maintain an advantage over competitors.
- The Revenue Logic – How the firm will make money, how and from whom revenue is generated.

While the time, scope and size ambitions of the firm are important elements of a firm's business model, they are also very specific to the circumstances of the specific firm, and the impact that OS strategies have on these ambitions is similarly dependent on this. Therefore, without knowledge of the particulars, its impact cannot be discussed in a general way and will not be further analysed in this paper. If, however, one embarks on a specific case study, this component would be quite valuable in analysis.

In practise, a firm may employ a strategy as a whole, certain features of a strategy, or a combination of these strategies. Aslett [36] identified over 80 different such combinations of development model, vendor licensing strategy and primary revenue trigger. The next section describes the analysis of the OS Business Strategies and their impact on the business model of a firm, using the framework described above.

1. Analysing the Strategies

1.1 Product Related Models

The value offering, market, and revenue logic for platform providers, niche and speciality providers and complementary product and service providers are discussed below.

1.1.1 Platform Providers

The original Linux distributors, such as Red Hat, Suse (now owned by Novell), Linspire (now owned by Xandros), and Canonical can be regarded as a typical examples of platform providers. Their core business is to package and sell the Linux operating system. These firms aggregate, integrate, and optimise the newest Linux files that are freely available [10]. However, platform provision is not restricted to Linux - other popular platforms include: IBM's Eclipse development framework; and the Drupal content management system.

The Value Offering - Network effects, both direct and indirect, have an impact on the overall value provided by OSS in this strategy. The utility of a software package increases with the number of agents using the package (direct externalities). The decision to adopt a software product depends on a number of compatible applications, which is in turn an increasing function of software diffusion (indirect externalities) [10]. Value is derived for the firm through a reduction in development and marketing costs, as well as increased brand awareness for complementary products and services.

The optimisation and loss-leader strategies also play a big role in the value offering. For instance, after IBM invested \$40 million in Eclipse and released it as an OS product in 2001, it acquired 20-30% of the IDE market by 2005 [37][14] [38]. By commoditising the framework, IBM was able to add value higher up in the development tool chain. Nokia's \$410 million acquisition and release of Symbian as OS is aimed at neutralising an emerging Linux mobile platform [39] and closing the gap on Windows Mobile. It also hopes the openness will spur creative application development, giving people new reasons to buy high-powered phones. Nokia could have spent more on marketing and sales, but instead it invested in OS code [1].

The market – The mass market for standardised packages generally caters for small to medium enterprises (SMEs) and private consumers, whereas the market for customised solutions is generally aimed at medium to large corporate customers. Furthermore, the mass market for operating systems is separated into the desktop and server market. The popularity of the Apache Web Server has meant that Linux dominates the server domain. Apache is employed by 70% of websites and Linux holds 30% in the server operating system market [40][41]. The desktop market share is much smaller in comparison, and still dominated by Microsoft Windows while Linux products compete with one another and with Apple for what is left. The solutions market is completely different and offers more value, as services and subscriptions tend to generate more income than licensing [14]. This can even be covered through partnerships with other companies, such as seen in Dell's OEM partnerships with RedHat and Ubuntu [42]. As noted by Asterisk creator Mark Spencer [43], the biggest threat to firms are other firms that can offer the same services, as the source code is freely available. Differentiation is key and is mainly achieved by branding.

The Revenue Logic – In this strategy, only marginal profits are gained from the sale of licensing or software. Profits are mainly made up from the sale of complementary products and services. Considering the recent success of Red Hat [45], it becomes apparent that the success is mainly due to strategies for standardisation and quality control, customer retention and the blocking of their competitors [44].

1.1.2 Niche and Speciality Providers

These providers aggregate, integrate and optimise a wide variety of OS projects. Project evolution and social dynamics are critical factors of this strategy and firms must commit to coordinate individual developers [10].

The Value Offering - The firm has a mutually dependent relationship with an OSS project. OSS is collected, developed and maintained ,while one of the main functions of the

company is to coordinate the activity on the dedicated product. Whilst the firm enjoys the benefits of the OS development model, the value offering provided depends specifically on the niche area in which they operate. For example, MySQL is a set of OS database products. The recent acquisition of MySQL by Sun Microsystems provides benefits for both: Sun gains ownership of a successful OS project, while MySQL gets major IT vendor backing, which will increase its adoption [46]. Another example is the Funambol open source mobile synchronization server. By changing their business model from a classic software business model to an OS one, they achieved increased adoption, shorter sales cycles, and negated the need for a large sales team [31].

The Market – The market for these software providers is very different from that for platform distributors. Because they provide niche offerings, their markets tend to be narrower, and growing the customer base can be challenging. Funambol managed only 8000 downloads per month in 2004, but improved this by: continuously improving its product; making it easy to download and install; marketing the product extensively; synergising with other products; and improving its documentation [31]. MySQL used a franchising model and an external sales organisation to grow their market share [34][37].

Competition for these niche providers come from similar proprietary or OS products. Certain markets are entrenched by big software houses, for instance IT Management has been dominated by the likes of IBM, HP, CA, and BMC. However, if an OS equivalent emerges to provide the same or better functionality, with better performance, at a lower cost, the market may change, as demonstrated by Zenoss's growth [47].

The Revenue Logic - The main revenues are generated from services like training and consulting, leveraging the assumption that the most knowledgeable experts on the software are those that created it. MySQL generates its income from several sources, including online support and subscription, franchising, training programs, customisation and consultancy. Licensing accounts for approximately 50% of its turnover [34]. Funambol generates revenue from licensing, custom solutions and services. Both firms use the dual or twin licensing model, where the same product is released under 2 separate licenses. A split OSS / commercial licensing model can also be used, where different versions of the product are released under an OS or commercial license, with the commercial version having additional features. Examples of this include Sun's OpenOffice.org and StarOffice products, and Apple's Darwin and OSX products [48]. Another licensing approach that can be used for products which undergo rapid change is to release older versions of a product as open source, while new versions remain proprietary (Hecker's Sell-It Free-It strategy).

1.1.3 Complementary Product Providers

The profitability of OSS may come from a product that is a complement to the OS code. If the OS software enhances the usefulness or quality of complementary products or if the users of the software and the users of the complementary product belong to the same network, the complementarity can be exploited by dynamically managing price, product quality, network size, and hiring for both products [4].

The Value Offering – The way in which value is generated from optimisation and loss-leader strategies. The Optimisation strategy which encompasses Christensen's Law of “Conservation of Modularity”, leverages commoditised technology by adding layers of value to it. Typically with loss-leaders the bulk of revenue generated is through sales of other software products . OS helps to build the overall vendor brand and reputation, makes the traditional products more functional and useful, increases the overall base of developers and users familiar with and loyal to the total product line; contributing to a sales increase. This strategy is often used by firms to sell hardware, accessories and other merchandise.

The Market – The sale involves something tangible and the market may even be

affected by demand, as is the case with hardware. This strategy is very dependent on reputation and branding, as there might not be much differentiation in the product offered and competition may be quite strong.

The Revenue Logic – With hardware, the patronage strategy is influential, as the costs associated with assembling an OSS product are much lower than those of developing the software from scratch. This provides the incentive for firms with large stocks of hardware trademarks to back OSS. They are relieved of the burden of servicing and updating software, and the customisation of embedded software may make the bundle as a whole more difficult to replicate increasing its value to the firm [49]. IBM recognised this when deciding to support Apache [50]. Major OEMs like HP and Dell support OS projects that provide tools, utilities and solutions making it easier for their customers to deploy their products [14]. Building on the popularity of OS, firms like O'Reilly, CNET and Crazy Penguin sell OS related publications, books, manuals, t-shirts etc., as well as take part in the organisation of conferences and trade fairs [10].

1.2 Service and Support Related Models

The provision of such services as consulting, system implementation and integration, support, maintenance, remote administration, training and application management for OS products is the successor to the for-profit business based on treating software as intellectual property [10].

The Value Offering – Lowered development costs, access to source code and the zero license costs involved in packaging OS products and offering services related to them, all contribute to lower sales volumes needed to realise a profit. This means that service provisioning can be offered by not only the big platform providers, but also by small independent enterprises who make this their core competence.

The Market – Customers range from small to large organisations and have different needs and function on different levels. They often pay for the solution and not the product, hence service is often project related. Services on software products are a classical people-selling business with constant and high marginal costs. The critical factor is not the sales volume but the availability of talented human capital. Process and product know-how are the differentiating factors amongst firms.

The revenue logic for each of the service related models are described below.

1.2.1 Service and Support Provision

Maintenance and support contracts are usually time and level based, with varying degrees of guaranteed service. Standard support services are within the range of SMEs, but 24/7 support may only be possible for large companies with sufficient resources. The actual developers of the software usually gain some competitive advantage, as they are more familiar with the technology and thus more able to support it.

1.2.2 System Implementation, Selection, and Consulting

Implementation (or installation) is a common support activity in the OSS community, and arises from the modularity of the software, requiring that various different components be installed to produce a working system. The availability of advanced package managers and installers in most of current systems has greatly reduced the complexity of installation.

Selection is a multi-stage process that involves an analysis of the needs of the customer and the selection of the various components that will address these needs in the most appropriate way. This selection needs to minimise the amount of code that needs to be developed, while also taking into account other factors including: reliability, technical

dependencies, availability of support and documentation, etc. To provide software selection services, it is important to keep track of the latest developments regarding the software.

Consulting and integration support refers to both the configuring of the system to fit into the existing infrastructure, and the development effort required to address those issues unique to the firm. This often requires significant time and effort. Consulting strategies may be based on knowledge in specialised areas or on specific software, for example consultation on legal matters and on content management or business intelligence software.

1.2.3 Training / Education and Assurance/Certification

This is one of the more versatile strategies. Training providers can offer their courses in the classical seminar style, or they can offer them online. Customers occupy different levels and have different needs. While most of the big distributors (Red Hat, JBoss) have officially sanctioned training programmes, there is a wide variety of products that do not, and for which specific training and certification can be created. Training is usually personnel intensive and may require a significant amount of effort to develop and test the training material, depending on the product, whom it is aimed at, and the depth that is covered.

Technical certifications are usually provided by integrators and external consultants, it can involve certification in compliance with a specific standard or the certification of suitability for a specific environment.

2. Conclusion

The term agility has received much interest in the business world, and is recognised as prerequisite for success in dynamic and turbulent environments. It includes two main attributes: responding to change promptly and appropriately; and capitalising on the opportunities created by change [21]. OSS has changed the software landscape and in order for firms to succeed in this environment they need to understand and take advantage of the opportunities that OSS provides. In technology driven industries, and with products freely available and limited protection, as in the case of OSS, the primary source of value for the firm shifts from product innovation to business innovation, where companies who focus on the whole picture succeed [2].

Properly crafted business models, have great power and can serve as an essential strategic tool for a firm, they focus a firm's activities on critical areas, while maintaining the big picture. Our analysis leads us to conclude, that while the business model is key, it is the manner in which a firm can reshape and align its business models to its environment and circumstances, that ultimately guarantees success. More significantly for firms dealing with OSS, the manner in which OSS is incorporated into and aligned with the firm's business model and strategies influences the success of the strategy. The generic models provide firms with a guideline, but it is up to the firm to determine which model or combination of models can be best aligned with the firm's overall business strategy to fully take advantage of the benefits afforded by OSS. This is demonstrated by the various examples discussed and mentioned in the paper. By providing a taxonomy of the general OS business models we highlight the various ways in which OSS can be used in business. Further, by analysing this taxonomy within a business model framework we provided a description of how these strategies are applied and demonstrate the value provided by these strategies. The intention being that firms can use this analysis to better understand the implications of OSS on their business strategies. From this they will be better equipped to formulate related strategies as a whole and adapt and mould their respective business models so as gain the most value from OSS.

Chesborough & Rosenbloom [25] note that the process of reshaping an initial business model, creates opportunities to discover new mappings between technical potential and

economic value, and that these novel mappings may contribute significantly to success. We need to become more aware of how we can continually adapt the elements of our business models and we need to understand the reasoning's behind them in order to avoid some of the pitfalls associated with them. Shafer et. al. [18] highlights how flawed assumptions of the underlying core logic, limitations in the strategic choices considered, misunderstanding value creation and capture, and flawed assumptions of the value network can be dangerous to a firm.

Through our analysis it becomes apparent that, OSS as a strategy is extremely useful in contributing to a firm's bottom line, provided that it is utilised promptly and appropriately in the given circumstances. The models that have been described provide a basis from which innovative firms can begin to adapt their business models, understanding the impact of OSS on the various elements, and how it can ultimately provide value to the firm. In addition, the models also provide a framework from which firms can re-evaluate and reshape their business models, possibly discovering new avenues of value.

3. References

- [1] C. Babcock, Open Source for SALE. In: InformationWeek, ABI/INFORM Global, 2008, pp. 25-30.
- [2] T. Pykalainen, Model for profiting from software innovations in the new era in computing. In: Technovation, 2007, (27)(4), pp. 179-193.
- [3] J. West, How Open is Open Enough? Melding Proprietary and Open Source Platform Strategies. In: Research Policy, 2003, (32), pp 1259-1285.
- [4] E. Haruvy, S. Sethi, & J. Zhou, Open Source Development a Commercial Complementary Product or Service. In Production Operations Management, 2008, (17)(1), pp. 29-43.
- [5] G. Goth, Open Source Business Models: Ready for Prime Time. In: IEEE Software, 2005, pp 98-99.
- [6] Bridges, Free/open source software (FOSS) policy in Africa: A tool kit for policy-makers and practitioners. In: 2005. bridges.org and (CIPESA). <<http://www.bridges.org/publications/92>>
- [7] Distrowatch Website [Online] <<http://distrowatch.com>>
- [8] N. Ensmenger, Open Source's Lessons for Historians. In: IEEE Annals of the History of Computing, 2004, (26)(4), pp. 102-104.
- [9] B. Fitzgerald, The Transformation of Open Source Software. In: MIS Quarterly, 2006, (30)(3).
- [10] A. Bonaccorsi, S. Giannangeli, & C. Rossi, Entry Strategies Under Dominant Standards: Hybrid Business Models in the Open Source Software Industry. In: Management Science, 2006, (52)(7), pp.1085-1098.
- [11] R.T. Watson, P.T. Boudreau, M.E. Greiner, & D. Wynn, The Business of Open Source. In: Communications of the ACM, 2008, (51)(4) pp. 41-46.
- [12] E.S. Raymond, The Magic Cauldron. In The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary. In: 2001. <<http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/>>
- [13] F. Hecker, Setting Up Shop: The Business of Open-Source Software. In: IEEE Software, 1999, (16)(1).
- [14] J. Koenig, Open Source Business Strategies. In: Riseforth, Inc., 2004.
- [15] S. Shah, Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development. In: Management Science, 2006, (52)(7), pp. 1000-1014.
- [16] T. Hemphill, A Taxonomy of Closed and Open Source Software Industry Business Models. In: International Journal of Innovation & Technology Management, 2006, (3)(1), pp. 61-82.
- [17] M.Morris, M. Schindehutte, J. Allen, The entrepreneurs business model: toward a unified perspective. In: Journal of Business Research, 2005, 58, pp. 725-735.
- [18] S.M. Shafer, H.J. Smith, J.C. Linder, The Power of business models, In: Business Horizons, 2005, (48).
- [19] P. Keen, & S. Qureshi, Organizational Transformation through Business Models: A Framework for Business Model Design. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences, 2006.
- [20] R. Alt, & H.D. Zimmerman, Introduction to Special Section, Business Models. In: Electronic Markets, 2001, (11)(1).
- [21] J. Feller, P. Finnegan, J. Hayes, Open Source Networks: An Exploration of Business Model and Agility Issues. In: Proceedings of the 14th European Conference on Information Systems. 2006.
- [22] D. W. Mitchell, & C.C. Bruckner, Business model innovation breakthrough moves. In: Journal of Business Strategy, 2004, (24)(1), pp. 16-26.
- [23] S. Makinen, & M. Seppanen, Assessing business model concepts with taxonomical research criteria. In: Management Research News, 2007 (30)(10), pp. 735-748.
- [24] Y. Wu, A competitive analysis of business models in e-commerce. In: Engineering Management

- Conference, IEEE International, 2002, pp. 848- 853.
- [25] H. Chesborough, & R.S. Rosenbloom, R.S, The role of the business model in capturing value from innovation: evidence from Xerox Corporations technology spin-off companies. In: *Industrial and Corporate Change*, 2001, (11)(3), pp. 529-555.
- [26] G. von Krogh, & E. von Hippel, E. The Promise of Research on Open Source Software. In: *Management Science*, 2006, (52)(7), pp. 975-983.
- [27] M. A. Rappa, The utility business model and the future of computing services. In: *IBM Systems Journal*, 2003, (41)(1) pp 32-42.
- [28] C. Daffara, Business models in FLOS-based software companies. In: 2007 , <<http://opensource.mit.edu/papers/OSSEMP07-daffara.pdf>>
- [29] M. Fink, The business and Economics of Linux and Open Source. In: Prentice Hall: Upper Saddle River.
- [30] S. Rahtz, & R. Metcalfe, Support Models for Open Source Deployment. In: *Ariadne*, 2004, (40). <<http://www.ariadne.ac.uk/issue40/oss-watch-rpt>>
- [31] A. Onetti, & F. Capobianco, Open Source and Business Model Innovation. The Funambol case. In: *Proceedings of the First International Conference on Open Source Systems*, Genova, 2005, pp. 224-227.
- [32] S. Krishnamurthy, An Analysis of Open Source Business Models. In: *Perspectives on Free and Open Source Software*, MIT Press, Cambridge, 2003, pp. 279-296.
- [33] S. Weber, The Political Economy of Open Source Software. In: *Berkeley Roundtable on the International Economy*, 2001, Paper BRIEWP140.
- [34] L. Dahlander, Appropriation and Appropriability in in Open Source Software. In: *International Journal of Innovation Management*, 2004, (9)(1), pp 259-285.
- [35] R. Rajala, M. Rossi, & V. Tuunainen, A Framework for Analysing Software Business Models. In: *Proceedings of the ECIS 2003, Conference on Information Systems*.
- [36] M. Aslett, Open source is not a business model. In: 451 CAOS Theory, 2008, <<http://blogs.the451group.com/opensource/2008/10/13/open-source-is-not-a-business-model>>
- [37] M. Brydon, & A. Vining, Adoption, Improvement, and Disruption: Predicting the Impact of Open Source Applications in Enterprise Software Markets. In: *Journal of Database Management*, 2008, (19)(2), pp. 73-94.
- [38] J. Duenas, C. Parada, F. Cuadrado, M. Santillán, & J. Ruiz, Apache and Eclipse: Comparing Open Source Project Incubators. In: *IEEE Software*, (2007), (24)(6), pp. 90-98.
- [39] B. Violino, Open Source's New Frontier. In: *CIO Insight*, 2008, <<http://www.cioinsight.com/c/a/Strategic-Tech/Open-Sources-New-Frontier>>
- [40] L. Lin, Impact of Users' Expertise on the Competition between Proprietary and Open Source Software. In: 39th Hawaii International Conference on System Sciences (HICSS'06), 2006, Track 8, p.166a .
- [41] A. Bonaccorsi & C. Rossi, Why Open Source software can succeed. In: *Research Policy*, 2003, (32).
- [42] I. Thompson, Dell teams with Red Hat on enterprise Linux. In: *Vunet.com*, 2008, <<http://www.vnunet.com/vnunet/news/2228219/dell-teams-red-hat-open-source>>
- [43] R. Gedda, Open source identity: Asterisk founder and Digium CEO Mark Spencer, Software telephony systems in for a shake up. In: *Techworld Australia*, 2008, <<http://www.computerworld.com.au/article/268874>>
- [44] M. Assay, Red Hat: The mother of all open source business models. In: 2006, <<http://asay.blogspot.com/2006/01/red-hat-mother-of-all-open-source.html>>
- [45] M. Assay, Some intriguing data behind Red Hat's 29 percent growth. In: 2008 <http://news.cnet.com/8301-13505_3-10050186-16.html>
- [46] P. Krill, Sun, MySQL synergies stressed in merger. In: *InfoWorld*, 2008. <http://www.infoworld.com/article/08/01/16/sun-mysql_1.html>
- [47] J. Burt, Open Source and IT Management. In: *eWeek*, 2008, (25)(21), www.eweek.com.
- [48] R.E. Hawkins, The economics of open source software for a competitive firm, Why give it away for free? In: *Netnomics*, 2004, (6), pp. 103-117.
- [49] A. Fosfuri, M. Giarratana, & A. Luzzi, The Penguin Has Entered the Building: The Commercialization of Open Source Software Products. In: *Organization Science*, 2008, (19)(2), pp 292-305.
- [50] N. Yan, D. Leip, & K. Gupta, The use of open-source software in the IBM corporate portal. In *IBM Systems Journal*, 2005, (44)(2). pp 419-425.