A Resource-Based Explanation for the Apache Consistent Dominance in the Web-Server Industry

Autoria: Carlos Santos Jr., Marcio Augusto Goncalves

ABSTRACT: Open source communities such as Linux and Apache became well known for providing high-quality software at free acquisition cost. Because of that, they have been widely adopted by many organizations. As a matter of fact, Apache has dominated its market in terms of share for 13 years now, competing with corporations as big as Microsoft. The resource-based view (RBV) of firms posits that for an organization to outperform its competitors (as Apache has done consistently), it must have resource(s) (1) valuable, (2) rare, (3) imperfectly inimitable, and (4) non-substitutable. Accordingly, one may conclude that Apache is an organization holder of such resource(s), which are Apache's source for sustainable competitive advantage. Our current available literature does not explain specifically what those open source organizations resources might be. This paper is effort in that direction. We ask, "How can an open-source software organization outperform its for-profit research-oriented competitors?" To answer this question, we develop three propositions based on the RBV. We demonstrate how each of these propositions are found in the web-server industry case and propose methods for their empirical evaluation. Future research directions are also provided.

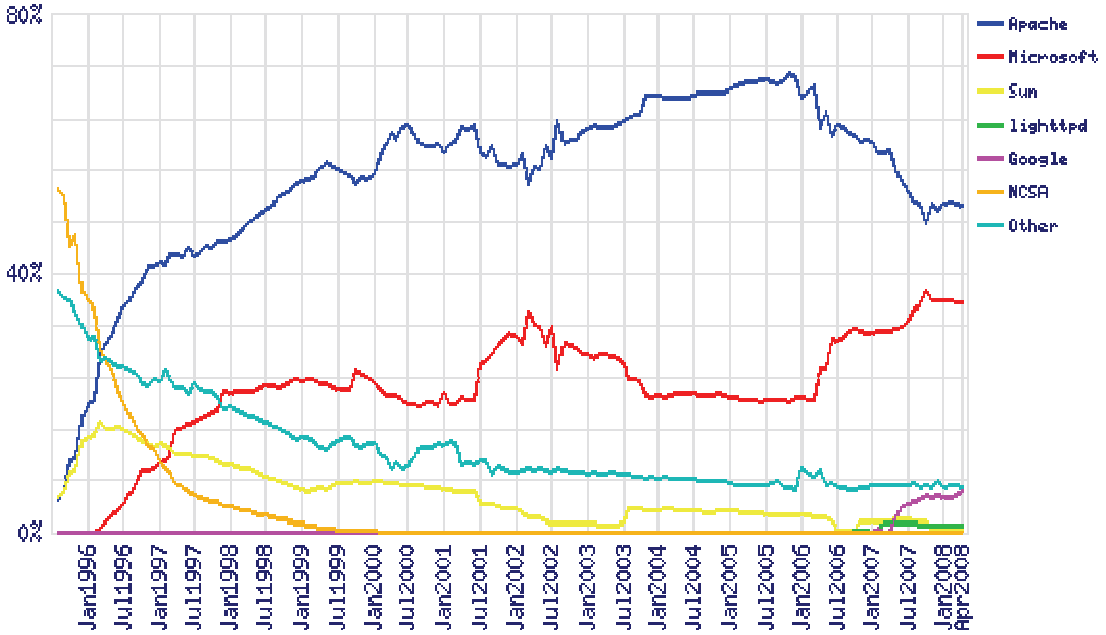
Introduction

The industry of web-servers has been dominated in terms of market share by an open source product for over 13 years now[i]. A recent survey, which collected data from more than 166 million websites, showed Apache with about 50% of the market, followed by Microsoft[ii] with over 35% (Netcraft, 2008). Other participants in the web server market include Sun[iii], Macintosh and the National Center for Supercomputing Applications at the University of Illinois-Urbana-Champaign, accounting together for the remaining 15% (see Figure 1 for details).

In terms of market share, clearly Apache has competitive advantage in the web server industry. Therefore, from a Resource-based viewed (RBV) perspective, one question of interest is what Apache's valuable, inimitable, non-substitutable and rare resource(s) enable its positioning as the market leader for such a long period of time competing inclusive with for-profit research-oriented corporations.

This paper is a theory-building attempt to explain Apache's competitive advantage based on its organizational structure, strategies and policies. We identify Apache's traits that arguably meet all theoretical criteria needed for an organization's resources to provide a company sustainable competitive advantage, in a RBV fashion. Specifically, we separate Apache from its for-profit competitors in the web server industry through its traits, mainly Microsoft due to its second position and its importance in the overall software industry, providing Apache a source for competitive advantage. Alternatively, we seek answer for the following research question: How can an open-source software organization outperform its for-profit competitors?

Figure 1- Web-server Market Share Distribution over Time.



Source: Netcraft (2008)

This paper's main goal is to develop propositions to guide future empirical research, and promote awareness of open source organizations capabilities, outperforming corporations of the magnitude of Microsoft.

The increasing involvement of corporations (e.g. IBM and Sun) in the open source arena has already attracted the attention of researchers. This, along with the assumption that corporate efforts are rationally driven by economic motivation, gives us clues that the corporate world has recognized the potential to reap economic benefits by the adoption of some of the managerial practices and the organizational model found in the open source organizations and, consequently, not yet existent in the for-profit corporations (resource heterogeneity). Thus, we take the risk and draw an initial conclusion from this scenario to be used as a general thesis in this study: Open source organizations can possess valuable resources that their for-profit competitors do not possess. The case explored here (i.e., the web-server market) is intentionally chosen to illustrate the point just made.

Miller's (1996) research justifies a study of this nature, since he found a lack of descriptive studies about configuration of organizations, meaning the overall interdependent system brought together in an organization by its "theme". It is these configurations of resources and strategies within the organizations of the web-server market that will be brought to bear here. Similarly, Robinson & Lundstrom (2003) stated that "research related to market expansion strategy has primarily focused on assessing the 'fit' between the organization and external factors such as market attractiveness, competitive advantage and risk, with minimal attention to internal factors such as organizational structure, management systems and corporate culture." In conclusion, these authors claimed that those external factors are necessary to properly formulate a plan, but the organization resources are the things that make possible a strategic plan to be implemented.

This paper is organized as follows. First, a literature review about the resource-based view of the firm (RBV) is presented. Second, a brief rather in-depth discussion of the open source community and software and its many research streams that exist nowadays are presented. Third, the theory here proposed is developed through a discussion section where Apache's structure and strategies will be argued to meet each and every requisite that the RBV approach claims to be necessary for a resource to give a company (sustainable) competitive advantage is presented. Finally, a brief discussion section is provided, followed by some considerations for future research along with the limitations of this research.

Literature Review

The Resource-based View of the Firm

The resource-based view (RBV) of the firm has its roots on the theory of imperfect competition (Melville et al., 2004). Prior to its development, researchers were normally justifying companies' ability to be profitable according to their positioning in a certain industry (Porter, 1985). This approach is normally referred to as industrial economics (IO). Although it has provided many insights to explain firm performance, it failed to explain why companies in the same industry experience different amounts of return, and why some go bankruptcy, while others don't. Furthermore, the amount of variance explained by environmental (external) variables wasn't satisfactory, creating the necessity for researchers to look for other matters. That scenario provided the basis for Barney (1991) to introduce the resource-based view (RBV) of the firm.

In the RBV theory, resources are traits that might be viewed as either strength or a weakness of an organization. These resources are assumed to be heterogeneous across organizations, which is a necessary condition for the theory to exist. Otherwise, if resources were homogenous, we would be returning to a condition firstly described by IO researchers, where the positioning of the corporation within an industry would determine its returns, not its internal resources.

Assuming then that resources are firm-specific, we now turn to the description of what characteristics a resource must possess to be elected as a potential source of (sustainable) competitive advantage, aside the fact that so far the RBV literature wasn't able to define the terms valuable and competitive advantage independently (Priem & Butler, 2001).

Barney (1991) claims that those resources have to be (1) valuable in the sense that they should provide the company the opportunity to conceive or implement strategies that enhance the corporation effectiveness and/or efficiency. Second, they have to be (2) rare, meaning that competitors willing to acquire it would have to pay premium prices. Third, these resources have to be (3) imperfectly inimitable, avoiding competitors to develop it, instead of buying. And finally, they have to be (4) non-substitutable, in the sense that competitors can not use something else in an attempt to avoid its premium price to be acquired or its difficulty to be developed.

Accordingly, a firm that does not possess competitive advantage, in order to catch up with its competitor (leader), has to identify what it is that gives that competitor competitive advantage. That situation is far from a trivial, and Barney (1991) discusses some of the issues that make this assessment a non-trivial activity.

A resource that meets some or all requirements presented above is also expected to be (1) causal ambiguous, meaning that most of the times not even the company that possesses and deploys it (not to mention an external entity such as a competitor) understands exactly how the resource they are now taking advantage from ended up being what it is. Second, this resource is expected to be (2) path dependent, in the sense that in order for it to occur some things (a scenario) have firstly to be in place. Then, a sequence of events has to occur, making a replication rather complicated. Additionally, these resources might be (3) socially complex, in the sense that they can be dependent of something almost non-replicable such as organizational culture or a country specific conditions. Therefore, to understand and replicate a resource and/or to possess the ability to develop a strategy taking advantage of a resource is a non-trivial and costly task, creating the opportunity for the competitive advantage holder to enjoy its positioning where it can reap abnormal returns.

Open Source Software and Communities

The Internet relies on open source software and practices surprisingly more than an inattentive user perceives. According to The Economist[iv] "[every] time [I]internet users search on Google, shop at Amazon or trade on eBay, they rely on open source software. More than two-thirds of websites are hosted using Apache, an open source product that trounces commercial rivals. Wikipedia, an online encyclopedia with around 2.6 million entries in more than 120 languages, gets more visitors each day than the New York Times' site, yet is created entirely by the public. There is even an open source initiative to develop drugs to treat diseases in poor countries."

An open source community is composed of volunteers, in the sense that they are not necessarily paid or employed. Those volunteers are dispersed geographically and brought together through some sort of IT structure, mainly the Internet (Hertel et. al, 2003). They are sometimes broadly referred as virtual teams, which can be defined as "a group of people who interact through interdependent tasks guided by common purpose[, ...] across space, time and organizational boundaries with links strengthened by webs of communication technologies" (Lipnack and Stamps, 1997). By the same reasoning, virtual organizations were defined by DeSanctis and Monge (1999) as "[...] a collection of geographically distributed, functionally and/or culturally diverse entities that are linked by electronic forms of communication and rely on lateral, dynamic relationships for coordination."

Hertel et. al (2003) described open source communities, as differing from collaborative networks: "Communities usually include a large number of people, and are open to anyone who wants to join as long as s/he obeys some general behavior rules. Collaborative networks are more restrictive in their access policy, relying on referral or reputation and develop a more specific community code including sanctions for violating this code."

Open source communities are composed of "hobbyists", but the number of paid developers working within them seems to be increasing, especially people connected to IBM and Sun Inc. Sometimes these communities have hundreds of collaborators. For instance, Egyed and Joode (2004) stated that "the Apache community roughly comprises 630 contributors of which about 90 belong to the core developer group". The product (software), as well as its content (source code[v]), produced by those communities are always made available on the Internet free of charge (Hertel, 2003) and the reasons why (motivations) those developers get involved in such activity have also been discussed in the literature.

Individual motivations are hard to be observed empirically, but some studies have found that most of the effort of the volunteers is due to the status attributed to and the enhanced reputation received by them as part of the group, referred to sometimes as signaling (O'Mahony, 2003; von Hippel, 2001a). Along similar lines, another characteristic attributed to those developers was that they "[were] not driven by monetary rewards but by competitive motives of status and reputation" (Hertel et. al, 2003). As a way to support those statements, Lee and Cole (2003) observed that the proportion of people joining the Linux group which does not, in fact, develop software has grown faster than the ones that do. Moreover, the authors stated that every component (file) of the software comes with a credits file, recognizing and describing the work of each member publicly.

Moon & Sproull (2002) and Hertel et. al (2003) pointed out some characteristics they have observed in open source communities. They have identified (1) "a general culture in which authority comes from competence", (2) the presence of a "delegative and participative leadership principles combined with clear responsibilities", (3) "a modular project structure that decreases unnecessary complexity", (4) "a parallel release policy that simultaneously enables rapid development and a stable working system", (5) "a motivating credit policy that not only acknowledges the contributions of developers but also, for instance, documentation work", (6) presence of "clear rules and norms of the community that are communicated online", and (7) a "simple but reliable communication tools that are available worldwide (e-mail, file transfer, Usenet discussion groups)."

Lee and Cole (2003) stated that open source communities violate the norms of regular communities in five dimensions: "(1) the assignment of intellectual property rights promotes trust building and knowledge sharing; (2) the membership is open and consequently the size of a community is much larger than that of a firm; (3) worker incentives and motivations shift from those of employees to those of volunteers, and unlike in a firm setting, there isn't an authority relationship to regulate the behavior of community members; (4) individuals are organizationally as well as geographically dispersed; and (5) the knowledge creation platform is based on a many-to-many communications technology."

The first point of divergence identified by Lee and Cole (2003) is due to the products as well as their contents (source code) generated by the volunteers be public. It is possible to have access to the software source code as well as to the e-mails exchanged among the community members free of charge via the Internet. In the case of Linux, access occurs through mail lists, "which is a virtual environment where Linux developers send their contributions, discuss implementation details, and interact with other developers" (Lee and Cole, 2003). But the second point can be somehow questioned, because projects can have leaders (informal or formal) who decide what is going to be included in the official version of the software (like the core group of Apache), and the community itself judges the work of each member, through critiques and recognition of the work. In addition, the status of a member increases when one's work is recognized by the leader and the community, possibly generating opportunities in the "real world" to the member.

As mentioned in the introduction, there is at least one case of an open source software being a market leader, the web server Apache. Others such as Linux and OpenOffice.org are seen as good candidates to substitute software developed by giant corporations like Microsoft (Hert et. al, 2003). For example, Lee and Cole (2003) pointed out that "[a]s a natural experiment, the Linux project has demonstrated the feasibility of a large-scale, online collaboration effort where developers and users can be one and the same", suggesting a high-quality product. Nevertheless, many other open source projects have neither expressiveness nor competitiveness.

Accordingly, one important issue in the IT industry and in the academia recently is how to make use of open source software practices (or resources, in RBV terms), as a building block for delivering business value. For example, Hertel et. al (2003) stated that "[o]ne of the most compelling aspects of Open Source Software projects is that they are predominantly based on voluntary contributions from software developers without organizational support in a traditional sense." This trend connects open source community and corporations, creating a scenario new to researchers, worthy of attention.

An Overview of the Apache Organization

Apache hosts web sites and provides content to the general Internet user through their web browsers. Its initial source code was developed at the University of Illinois and further released to the community in order to the development be continued. Its version 1.0 was release on Dec. 1, 1995 (von Hippel, 2001a). The last version of the system is available on the website for download (http://httpd.apache.org/) and is numbered 2.2.8. Apache maintains in its website (http://httpd.apache.org/contributors/) the list of all its contributors and the main contributions received, as a form to compensate volunteers for their time and effort.

Fielding (1999) described Apache as "a collaborative software development effort aimed at creating and maintaining a robust, secure, efficient, extensible, and open source implementation of an HTTP (Web) server. The project is managed by the Apache Group, a geographically distributed set of volunteers who use the Internet and Web to communicate, develop, and distribute the server and its related documentation." IBM is one of the members of Apache, but its participation is not different from any other "volunteer's".

Theory Development

Web-server Environment: Apache and Its Competitors

We consider here Miller's (1996) definition of organizational configuration as an organizational resource in the RBV term. Miller (1996) stated that "configuration, in this sense, can be defined as the degree to which an organization's elements are orchestrated and connected by a single theme." His research had a focus on organization's goals guiding their actions towards goals achievement. In summary, he concluded that configuration is more likely to be the source of competitive advantage than a strategy per se. Furthermore, Miller et al (2002) explained that these characteristics of organizations are not normally clear, but they can be to some extent captured by organizational processes and designs. We argue that it is in this arena that Apache differentiates itself from its competitors, creating a beneficial market "configuration" for Apache to hold competitive advantage. Accordingly, we first present three different areas, (1) productivity, (2) organizational complexity and (3) the nature of the relationship between an organization and its customers. Then, we demonstrate how Apache differs from its competitors in each of those areas, and how these differences are likely to be sustained over time.

Productivity

Right at first glance, to superficially distinguish the Apache organization and its competitors (our focus is on the Microsoft Corporation) isn't very difficult. Apache is mainly composed of volunteers, whereas Microsoft is an employer in the ordinary sense. This leads us to identify a series of differentiations between them both. For example, the hiring (or attracting in the Apache case) is substantially different in each organization. Also, the types of control mechanisms available to the hierarchically higher participants are also different in nature due to the non-employee characteristic of Apache's contributors.

Prior research has found that open source volunteers are homogeneous in a demographic sense (e.g. age, education, gender, etc.), and in a motivational sense (i.e. what they expected to take out of their volunteering). That, one could argue, suggests a bigger diversity of employees' motivations to be engaged in the Microsoft's activities than in Apache's, creating, consequently, a larger occurrence of conflicts within Microsoft, hindering decision-making processes, task distribution and the achievement of consensus, for instance.

This issue of consensus was also discussed by Miller (1996). He explained that when high levels of it are present, "strategy, structure, process and culture are all shaped by a central goal or focus and hence complement one another." Furthermore, this focus is reflected in the structure of power and the shape of the top management team, facilitating decision-making processes.

Moreover, Apache volunteers have no contractual obligations with the organization, implying that they do not receive direct monetary compensations by simply being members of the organization. Apache participants got engaged in the organizational activities already knowing that to be a volunteer work. The simply fact of not being paid for their work, we argue, makes volunteers to focus on results in a more productive manner (because they are timely restricted) (Fielding, 1999). The virtual situation of these volunteers also helps this managerial "mechanism" to work, since volunteers can not pretend to be working to their “bosses” (since they wouldn’t be able to see!), it forces them to strictly focus on getting their work done. These things together provide us ground to argue that Apache volunteers tend to be more productive than Microsoft employees, without the need for formal control, reducing also managerial costs. By productive here, we mean achievement of the same results in a less costly (time and money) manner. Thus, in a form of proposition we have,

P1: Given a same software development task, the Apache approach to develop software is expected to be more efficient than its for-profit competitors’ ones.

Now, we shall turn to the discussion of the odds of Microsoft or any other for-profit competitor to be able to catch up with this first Apache advantage. First, we assume, based on our proposition 1, that an organization with volunteers instead of employees is a valuable desirable condition. Second, to possess a reputation level powerful enough to attract people in to develop software and contribute with time without expecting to receive any monetary compensation is rare in the software industry (other companies in this sector are Apple and Sun). Third, we do not believe this Apache’s advantage could be possibly imitated, for it would be necessary that a corporation of the size of Apache’s competitors to back off and develop the reputation necessary to attract volunteers. A price we do not believe any these corporations would be willing to pay. Fourth, there’s no readily substitute to this reputation Apache possesses. As Miller et al. (2002) pointed out, “popular brands and unique capabilities help sustain a company’s competitive advantage. However, they cannot be built by imitation.” Given that, we believe this advantage Apache has is likely to be maintained over time. Consequently, we argue this is a candidate advantage to be (at least) one of Apache’s sources of competitive advantage.

Organizational Complexity

It is our opinion that Apache is a case of extreme focus on core competences. Apache is a simplified type of organization. To begin with, it does not have to run payroll functions, manage finance, accounting or human resources department (hiring or firing processes). This oversimplified type of organization facilitates the achievement of synergy, a situation where processes and structures highly complement one another. Consequently, to direct and coordinate people becomes easier, and the speed to react increases, as well as the cooperation among members, because the organizational goal is simpler to the members, directing their efforts effectively. Furthermore, there are no stockholders to be factored in during decision-making processes. And, on top of it all, demand, competition and regulations are variables that do not matter much, since members are users, and an increase or a decrease in “sales” (market share) does not change anything in the organization activities. Member activities won’t change, since they do it for other reasons than economic return. In other words, their activities are not dependent on market variables such as sales.

In a nutshell, the environmental uncertainty of this industry is likely to be perceived quite differently by a for-profit standpoint (more uncertain). A fair amount of environmental uncertainty disappears in the open source organization case.

Apache seems to be a case where an extreme focus on core activities occurs. Every single effort can easily be related to the organization’s final goal by its members, a connection not so easily made in for-profit and highly diversified organizations. These things together make us believe that the management of Apache is perceived to be, and in fact is, much simpler than its counter for-profit parts. This is expected to be translated into more complex (i.e. time consuming to be implemented) and costly policies in for-profit corporations than in Apache. Thus, we have that,

P2: Apache's rules and policies regarding volunteers' (employees') work are significantly less complex and costly than its equivalent for-profit competitors' ones.

This expected situation is also likely to be sustained over time. First of all, unquestionably, a situation where a company is able to make use of less complex and costly policies and procedures is valuable. Second of all, the necessary conditions (e.g. to have a simplified organizational structure without payroll or hiring) for those policies and procedures to be successfully implemented are rare in a competitive market as software development. Third and fourth of all, for that same reason, this procedures and structure fit can not simply be copied (developed) or acquired, being then non-substitutable and non-imitable (Miller, 1996). Thus, this Apache advantage is expected to be retained over time by the organization in comparison with its competitors.

The Customer and Organization Relationship

As it was pointed out by Siggelkow & Levinthal (2006) and Hart (1995), organizations have to look for configurations that are consistent with its available resources as well as with its environment. It is this fit that is believed to be the source of competitive advantage of firms. This task of monitoring the environment is costly and complex. Apache, again, has here an advantage not likely to be copied by any of its competitors.

The task of managing the relationship between the organization and its clients (part of the environment) is significantly simplified in the Apache way of production. The web-server Apache is developed by its own clients. As von Hippel (2001a) pointed out, those open source "communities run completely by and for users." That gives them "a great advantage over the manufacturer-centered development systems that have been mainstay of commerce for hundred years." In practice, there's no such thing as sale process or quality assessment through the customers' point of view.

Apache's developers have at least one "real" job (Fielding, 1999). Thus, if we assume that Apache's volunteers work as software developers in their real jobs, or at least have a close relationship with the IT department of their companies, it isn't very difficult to conclude that they know what a software of that kind must do and how. Furthermore, they potentially have influence on their companies' web-server choice decision. Put simply, the Apache developers are also Apache customers, or the customer develops his own software (von Hippel, 2001b). This equivalence facilitates the assessment and implementation of users' perceptions in the software, helping Apache to meet client's expectations and needs far easier than its for-profit competitors, which have users (customers) and developers as separate entities. In other words, there's a great concentration of resources and attention to the very core activity of Apache, narrowing and customizing its organizational activities (Miller, 1996). That situation does not occur in an ordinary for-profit corporation. In a nutshell, "eliminating" the client from the developmental process might increase the likelihood of achieving better quality, since quality will be assessed by the user who also developed the software. One would be assessing its own product. Thus, we have that,

P3: Customers' evaluation and satisfaction towards web-server software are higher for Apache's software, in comparison to its for-profit competitors'.

For Microsoft or any for-profit corporation to bring its customers to a position as close as the one they occupy in the Apache case would be something not at all easily achieved. Basically, it is a necessary condition for a company to reap economic returns by selling a product to keep its customers to a minimum distance where there's still dependency of the customer towards the corporation, creating or maintaining willingness to pay for the product or service. In the Apache case this willingness does not exist, since developers are also users because the piece of software can be obtained for free. The same strategy could only be pursued by a for-profit corporation by releasing its software to the community; a situation that would immediately reduce corporate profits. Thus, this last Apache advantage here discussed is also likely to be sustained over time, if its main competitors keep being for-profit corporations.

Having developed our model, composed of three different organizational arenas and their respective propositions, we will now turn to a brief discussion on how they could be tested in future research. Our intention is to facilitate future research endeavors, providing discussion on possible data collection and analysis methods.

Suggested Methods for Empirical Study

We divided this section in three different sub-sections, separating each theoretical proposition developed. As an overall observation, the study here described would be composed of comparative case studies, given that our focus is on major organizations' characteristics and processes from the web-server industry and their respective performance in the marketplace, measured as market-share distribution.

Proposition 1*: Given a same software development task, the Apache approach to develop software is expected to be more productive than its for-profit competitors' ones.*

This proposition requires the researcher to collect output information, such as time-to-market performance a both types of organizations have. The ideal situation would be to set up an experiment in two or more organizations, being Apache necessarily one of them. In this case, the researcher would create different tasks (problems) related to software development and they would be given to different organizations to be solved. Efficiency (time) and effectiveness (right or wrong answer) would then be evaluated and compared among the organizations. Unfortunately, this ideal situation is not likely to happen due to many operational reasons. Thus, we provide an alternative approach.

A researcher could focus on the resolution time of bugs (problems in the software) identified by corporations (users) or by the own researcher. Bugs identified in both the Apache and in the Microsoft software would be reported through the same media, let's say e-mail, and then a lag from the report time to the resolution time would be stored. Over time, these data could be used as a proxy for efficiency on solving problems in the software of focus, making an objective comparison possible. In this scenario, it would be a field research, not an experiment as it was presented like an ideal situation.

***Proposition** 2: Apache's rules and policies regarding volunteers (employees) are significantly less complex and costly than its for-profit competitors' ones.*

This proposition can be tested through a first stage of survey research aimed at the decision-makers of the web-server industry organizations, intending to describe organizational policies regarding hiring, promotion, and laying off; and procedures regarding reporting bugs and proposing changes in the software or in the organizational routines, for example. In a second stage, a focus group with organizational policies' implementation and maintenance specialists would be performed to evaluate the complexity of the policies, comparing one organization's policies with the others and then ranking ordering them.

***Proposition** 3: Customers' evaluation and satisfaction towards web-server software is higher for Apache's software, in comparison to its for-profit competitors'.*

This proposition could be tested through surveys aimed at a sample of websites' managers (web masters). The only intention of this survey would be to evaluate the perception of quality, in its many variant forms (e.g. downtime, customer service, etc.), of the managers regarding the web-server currently in use. Having collected the data, an ANOVA-type table could be constructed and differences in quality perceptions could be evaluated.

Discussion: Apache's Advantages as Potential Drawbacks

As previously stated, short-term advantage does not necessarily mean a long-term one. Apache seems to be a case of extreme focus on core competencies, which can be seen as a double-sided sword. Miller (1996) made this point, stating that "there is a danger that such very highly configured firms will become too simple – too dominated by a single world view, too monolithic, too driven by one theme or function." Thus, on one hand, this characteristic might be leading the Apache to achieve competitive advantage, as it was discussed in this paper and is also believed by its authors. However, on the other hand, prior research on innovation has shown that the lack of diversity tends to hinder innovation levels (Galunic & Rodan, 1998). This type of strategy, which is very narrow and surrounds very few elements, has also been found to lead organizations to lose their "resilience" and "relevance" (Miller, 1996). Moreover, an excessively homogenous group of people, as the open source organizations were found on prior research, might also hinder innovation. Consequently, future likelihood of Apache sustaining this competitive advantage might be diminished by this short-term advantage.

Limitations and Further Research

Being a theoretical paper, the main limitation of this study is to draw conclusions based only on prior researches and on authors' observations. Further empirical research is clearly needed in order to evaluate the accuracy of the propositions here developed, hopefully extending our actual understanding of the open source phenomenon. Thus, we shall now turn to the discussion of some further researches that we believe would be able to extend our understanding on this topic, proving basis to question the conclusions here presented.

First, further research is needed to understand why we still are able to find organizations (websites) using other web-servers besides Apache. Given that this software is claimed to be of high quality, the fact that it can be obtained free of charge, and the assumption that managers tend to be economically rational, how can we explain some organizations paying for Apache's competitors products? Is/Are any of our assumptions wrong? Are they valid only for some cases? What are those cases?

Second, as it was discussed in the end of the literature review, the open source phenomenon is being more and more influenced by the corporate world. This identification led Fitzgerald (2006) to suggest that we should even change our way to refer to open source. He claimed that OSS2.0 would be a better label to reflect these changes. Thus, has this transformation affected the open source way of production to a point that its differentiation from the ordinary way does not make sense anymore? What is the nature of this transformation? Or is it just a matter of separating open source organizations in two different groups (e.g. traditional and OSS2.0)?

Finally, we can not avoid asking the question on whether any comparison of open source software with corporate-proprietary ones is legitimate. Since these open source organizations do not compete in the regular sense for profits or for survival, how can we reconcile the traditional economic perspectives that permeate the strategic management field with the indeed existent open source phenomenon that affects regular corporations' activities?

References

Barney, J. (1991). Firm Resources and Sustained Competitive Advantage. Journal of Management, 17, (1), 99-120

DeSanctis, G., Monge, P. (1999). *Introduction to the special issue: Communication Processes for virtual organizations*. Organization Science. 10 (6) 693-703.

Egyed, T. M. and Joode, R. W. (2004) *Standardisation and Other Coordination Mechanisms in Open Source Software*. International Journal of IT Standards and Standardization Research, Vol. 2, No. 2.

Fielding, R. (1999). Shared Leadership in the Apache Project. *Communications of the ACM*, *42*(4), 42-43.

Fitzgerald, Brian. The Transformation of Open Source Software. MIS Quarterly. 2006; 30(3):587-598; ISSN: 02767783.

Galunic, D., & Rodan, S. 1998. "Resource combinations in the firm: Knowledge structures and the potential for Schumpeterian innovation." Strategic Management Journal, 1193-1201.

Hart, S., 1995. "A natural-resource-based view of the firm." Academy of Management Review, 986-1014.

Hertel, G., Niedner, S., & Herrmann, S. (2003). *Motivation of software developers in open source projects: An internet-based survey of contributors to the Linux kernel*. Research Policy, (Special Issue on OSS).

Lee, G. K., Cole, R. E. (2003). *From a firm-based to a community based model of knowledge creation: The case of Linux kernel development*. Organization Science. 14(6) 633-649.

Lipnack, J. & Stamps, J. (1997). *Virtual teams*. New York: John Wiley and Sons, Inc.

Miller, D. (1996). Configurations Revisted. *Strategic Management Journal*, *17*(7), 505-512.

Miller, D., Eisenstat, R., & Foote, N. (2002). Strategy from the Inside Out: Building Capability-Creating Organizations. *California Management Review*, *44*(3), 37-54.

Moon, J.Y. and Sproull, L. (2002) *Essence of Distributed Work: The Case of the Linux Kernel,* In P. Hinds and S. Kiesler (eds.) Distributed Work, Cambridge, MA: MIT Press, pp. 381-404.

Nigel Melville, Kenneth Kraemer, and Vijay Gurbaxani, "Review: Information Technology and Organizational Performance: An Integrative Model of Business Value," *MIS Quarterly*, Vol. 28, No. 2, June 2004, pp. 283-322.

O'Mahony, S. (2003): *Guarding the Commons: How Community Managed Software Projects Protect Their Work*, in: Research Policy, vol. 32(7), pp. 1179-1198.

Porter, Michael E. Competitive Advantage. NY: The Free Press, 1985.

Priem, R. & Butler, J. 2001. "Is the resource-based view a useful perspective for strategic management research?" Academy of Management Review, 22-40.

Robinson, G., & Lundstrom, W. (2003). Market expansion strategy: development of a conceptual market expansion decision scorecard. *Strategic Change*, *12*(5), 259-272.

Siggelkow, N., & Levinthal, D. (2006). Temporarily Divide to Conquer: Centralized, Decentralized, and Reintegrated Organizational Approaches to Exploration and Adaptation. *Organization Science*, *14*(6), 650-669.

von Hippel, E. (2001a). Innovation by User Communities: Learning from Open-Source Software. (Cover story). *MIT Sloan Management Review*, *42*(4), 82-86.

von Hippel, E., (2001b) *User toolkits for innovation*, Journal of Product Innovation Management vol. 18, pp. 247-257.

[i] http://news.netcraft.com/archives/web_server_survey.html - Accessed on April, 2008.

[ii] "Sun is the sum of sites running SunONE, iPlanet-Enterprise, Netscape-Enterprise, Netscape-Fast Track, Netscape-Commerce, Netscape-Communications, Netsite-Commerce & Netsite-Communications."

[iii] "Microsoft is the sum of sites running Microsoft-Internet-Information-Server, Microsoft-IIS, Microsoft-IIS-W, Microsoft-PWS-95, & Microsoft-PWS."

iv Edition of Mar 16th, 2006 -http://www.economist.com/displaystory.cfm?story_id=E1_VGNQJQQ

v "[I].e., the human-readable commands [...] of a computer program." Hertel et. al (2003)