Strategy in the New Economy
Final Paper
December 17, 2003

Emiliano Abramzon
Ryan Bloom
Aaron Mohrman
Colin Williams

1 Introduction:

A Non-Profit Prophet Challenges Software Capitalists' Economic Model

One of the primary tenants of capitalism is that the entity responsible for creating valuable goods is a for-profit firm, whether it is a sole proprietorship or a publicly held corporation. Modern capitalist economies have significant functions that are fulfilled by government (e.g. enforcing property rights and policing anticompetitive behavior) and non-profit entities (e.g. helping for-profit firms coordinate their activities and providing for underserved populations). But the vast proportion of the productive capacity remains in the profit-seeking sector. It is rare to see non-profits competing directly with for-profit companies, and when this happens it is usually in the selling of services for which non-profit status gives a firm improved credibility, like education and health care.

Since its inception, the computer software industry has been dominated by large for-profit companies that utilize their customer relationships and technical expertise to produce and support sophisticated products. As the Internet bloomed in 1998, Microsoft, Netscape, IBM and other major software vendors focused on becoming the dominant provider of web server software. This is the software that allows computers to publish content onto the World Wide Web. But none of them succeeded. Within a few years, the web server market was dominated by a product made by a non-profit group called the Apache Organization. The story of how Apache displaced its larger competitors to become the dominant provider of infrastructure on the Internet is a quintessential New Economy fable, offering lessons about how networks can profoundly change the terms on which firms compete.

There are other interesting lessons from the Apache story. Economists have understood for years that the size and structure of a firm is a direct product of the firm's costs, specifically the "interaction costs" of working with others. One effect of the information technology revolution that drives the New Economy is the drastic reduction in the costs of communication and coordination—both within firms and between firms. This should change the ways companies are organized, driving current firms to become faster, smaller and more focused. Beyond these general guidelines, it is not clear how the firms of the future will be structured. We can see the beginnings of these changes in today's highly networked companies, but the adoption process is a slow one. Apache, by contrast, represents a group that has been operating on the Internet since its inception, and has little connection to traditional forms of organization. It provides a hint of what the companies of the future may look like.

2 Internet Generation Companies and Strategy in the New Economy

2.1 The New versus the Old Economy: Does the New Economy differ?

The term New Economy refers to the surprising qualities of the US economy in the 1990s, which showed a level of productivity growth that was unprecedented in the developed world. This remarkable performance is tied to two structural changes: the advent of globalization and improvements in information technology. Globalization represented a well-understood continuation of years of progress in reducing the legal and logistical barriers that prevented countries from capitalizing on their natural advantages. By contrast, information technology (particularly computer networks) has radically lowered the costs for firms to acquire information and coordinate business practices.

The result is that firms have the opportunity to share both information and entire business processes with partners in ways that would have been unimaginable even five years ago. This makes it likely that many functions that the firm once had to undertake can now be implemented by outside specialists. In some situations, the advantages of outside expertise in non-core business processes are so dramatic that it will be hard for companies to remain competitive *without* outsourcing these processes. Even beyond direct outsourcing, many New Economy companies make complex goods whose value depends on a wide range of complements. Therefore, the success of a firm will depend not only on its performance, but also on the health of the ecosystem in which it resides.

Firms adapt to the opportunities and demands of the New Economy at different rates, depending on their internal capabilities and the demands of their industry. The firms which are furthest along in this adoption are called Internet Generation Companies (IGCs).

2.2 What is an Internet Generation Company?

There four basic dimensions of an organization's operations:

- Management of internal processes
- Interaction with suppliers
- Interaction with customers
- Interaction with the environment (Industry, Competition, Politics, etc).

A pure IGC is the one that interacts and shares information over the network in these four dimensions. Today, one observes companies along the entire the spectrum—from Old Economy to IGC. Operational efficiency only buys entry into the game. Companies must go further and leverage their productivity, agility, innovation, as well as intellectual and human capital, to devise and execute a unique business strategy.

Management of Internal Processes

Organizations can achieve greater efficiency by using an Internet platform to optimally manage the following processes:

- *Human resources management*: Recruiting, career development monitoring, and staffing are just some of the key processes that can be enhanced using the Internet. Imagine a company like McKinsey winning a contract to do work for an energy company in Indonesia. If McKinsey does not have the resources in Indonesia (a very likely scenario), it must staff that team with experts from other parts of the world. Before the Internet era, the consulting firm had to call HR managers in every country in search of the necessary talent. Today, McKinsey posts a requirement on its company intranet or simply searches the career development database to locate experts in the energy sector. Through internetworking, this process became more efficient.

- *Knowledge sharing*: In today's global economy, knowledge sharing is a key component of firm strategy. Companies tend to operate in several locations under different circumstances. In the old economy, sharing experiences with colleagues was expensive, complicated, and cumbersome. Today, employees can post their experiences online so

that anyone can access this information anytime and anywhere. Moreover, applications built on top of the Internet platform, including e-learning tools, make this alternative even more attractive.

- *Corporate governance*: This is the *topic-du-jour* in the corporate world today. In the Internet era, companies have the potential to improve governance by sharing information with greater transparency. This does not mean that the Internet can discover information that management attempts to hide or obscure. However, when management wants to share information, it is much easer to do so. Additionally, the general availability of information at all levels makes people aware of the wealth of information they could obtain if so desired.

- *Internal communication*: Motivating an extensive sales force, organizing a global company meeting, communicating the company's new vision, or even generating a sense of urgency in anticipation of a significant organizational change—all these things can be done faster, easier, and cheaper by using the Internet. IGCs take advantage of new e-mail features, create intranet web sites, perform online surveys, launch corporate online newsletters, and more. With the right content and the ability to share it, companies can vastly increase their operational efficiency, which in turn can enhance their strategic efficiency.

- *Standards, procedures and new product development*: As one will observe in this paper, internetworking allows organizations to work through decentralized structures. Participants no longer need to be in the same place at the same time in order to work together towards a common goal. This paradigm shift provides some advantages and disadvantages. Companies must learn to operate with decentralized structures. To do this, clear processes throughout the organization must be designed, created, communicated, and most importantly, adopted. Clear incentive mechanisms and performance metrics must be defined as well. The more decentralized and bigger the organization, the more important it is to have clear standards and processes. As a result, if all of these resources are leveraged correctly, efficiencies in new product development and innovation can be expected.

Interaction with Suppliers

Companies can now integrate their supply chain with suppliers. Such possibilities include vendor selection, vendor inventory management, vendor qualifications, ordering, payment, and more. Before the advent of Internet, it was considerably more difficult for organizations without their own proprietary electronic communication networks to share information, internally and externally. Most transactions were executed phase by phase, while other processes like vendor inventory management were almost impossible to pursue. IGCs benefit from new communication technologies that exchange data both within and outside the organization. As mentioned above, the inflection point is the ability to exchange information quickly and inexpensively. To be sure, one can expect many more tools to be developed as supplements and complements to the information superhighway which will further leverage its benefits. Organizations that are willing to work collaboratively as buyers and suppliers, to understand

each other's needs, and to share information will likely be more efficient, and potentially more competitive, than those that work differently.

Interaction with Customers

The Internet era, together with the development of new software and the declining price of computing power, have enabled organizations to better understand their customers, interact with them more frequently and intimately, and enjoy greater profitability as a result. Customer relationship management (CRM) models and new product development processes are just some of the activities that depend on accurate customer information. Enhanced customer interaction and the valuable information that it provides equips IGCs with not only the ability to extract maximum customer value, but also to deliver optimal solutions to the customer.

One example of this is when we search for a certain type of book at Amazon. If it so desires, Amazon can trace our searches and then by using a sophisticated algorithm, can suggest books in similar categories that we may desire. However, if we stroll into a Barnes & Noble store, browse for business books but ultimately buy nothing, B&N will neither know that we were in the store nor that we were looking for business books. Absolutely no information value was created or recorded.

In the B2B (business to business) arena, where supplier-customer interactions are more frequent, such web-based tools are even more powerful. An IGC will take full advantage of these possibilities.

It is important to note that a stand-alone CRM software implementation is not the magic bullet that will allow companies to succeed in enhancing customer interaction. Rather, it is the whole set of well-defined process and procedures running in tandem with internetworking technologies that will generate the most value for the organization. In the Amazon example, it would be useless for the company to record customers' preference if they cannot interpret the information and use it to sell the customer something she wants.

Interaction with the Environment

In the Internet era companies can more easily discover valuable information regarding their relationship with the macro environment.

We define interaction with the environment as the set of responses to externalities that can influence strategy and profitability. Examples of this include new market entries, competitors' new strategies, new discoveries in the industry, category benchmarks, and new government regulations. All such information flows over the Internet minute by minute, second by second. In the venture capital word, start-ups that are willing to enter a new market now have the opportunity to do significant market research on the Internet and better gauge the size of their opportunity. Ten years ago, the same process would have either been impossible or too expensive, or require substantial time and effort.

Additionally, news of an event like a worker strike in Venezuela's oil industry will reach everybody in the energy business in real time. Decision makers can then either take advantage of the situation, or mitigate any negative impact on their organization. IGCs are vastly better-

informed of the market environment, with instant access to information that could potentially affect their behavior. To be sure, these companies must be ready to react quickly and adapt to changing circumstances as they unfold. Like global foreign currency exchanges, IGCs never sleep.

2.3 Strategy in the New Economy

It is a mistake to believe that the New Economy undermines what we have learned about corporate strategy. Companies still must be assiduous in developing strategic competencies, both through planning and experience. They still need to monitor changes in their competitors' activities and those in the broader environment. In fact, the principles of strategy are more pertinent now than they have ever been since the rate of change has accelerated. From a strategic perspective, the New Economy has intensified the relevance of concepts that have always been true. The Internet era will punish those companies that are not equipped to efficiently gather, process, and interpret information, and then leverage this information to gain strategic efficiency. Further still, execution of strategy is just as important a crafting it. According to Nohria *et al* in their strategy article "What Really Works," "It really doesn't matter if you implement ERP software or a CRM system; it matters very much, though, that whatever technology you choose to implement you execute it flawlessly."[1]

Because it is easier for firms to specialize and compete on a more intense level, it is vital for managers to understand the specific activities their firms perform better than others, and focus on them. While previous eras allowed firms to meander for a while and subsidize low-performing divisions that lagged in their industry, today's competitive threats are such that a lack of focus is no longer a luxury firms can afford. Additionally, because markets change quickly and unpredictably, it is more important than ever that companies be open to emerging strategies. There is still a case for basic strategic planning, but there are simply too many factors beyond a company's control for their entire strategy to be planned out and codified in advance. Lastly, while it has always been important to monitor one's customer, today monitoring one's competitors and other related businesses is even more important. This is because firms were comparatively less dependent on suppliers and partners, and their markets changed relatively slowly. However, in the New Economy it is impossible for a firm to undertake any strategy without understanding the changing context in which its suppliers, partners and complementors are operating.

3 IGC exemplar/illustration

3.1 Background on the firm chosen: who is the firm, background on its industry, its competitors, the firm's competitive position, and its recent performance etc.

> *"First, let's be clear on the meaning of innovation. It's a lot more than invention. Innovation is the intersection of invention and insight. It is the fusion of new developments and new approaches. Its potential is especially*

[1] Nitin Nohria *et al*, "What Really Works." Harvard Business Review, July 2003.

> *large today, thanks to the emerging knowledge based economy. Innovation is focused less on things and more on ideas, collaboration and expertise."*[2]
> —Samuel J. Palmisano, CEO, IBM

Background on the Firm

History of the Apache Server—A "Patchy Server" Grows Up

In the mid-nineties, web server software was shaping up to be one of the biggest new markets for software vendors. When Netscape realized that it could not make money selling a browser when Microsoft was giving one away, it shifted its strategy to emphasize selling web servers.[3] But it was not alone, as soon Microsoft, IBM and others were providing web server software. Paralleling the web browser market, Microsoft and IBM hoped that by giving the product away for free they could help sell their other products.[4]

Apache began in 1994 when one of the web's early system administrators, Brian Behlendorf, needed code enhancements for the web server written at the National Center for Supercomputer Applications (NCSA) at the University of Illinois. As he grew frustrated and tired of waiting for NCSA staff to respond to the many software patches (code improvements) that he and others made and posted to NCSA software, he and six others formed a mailing list to coordinate and track the changes and improvements each was making to the server. The group released the first version of Apache HTTP Server 0.8 in August of 1995.[5]

The Apache group provided a web server for free on the Internet, just like IBM and Microsoft. However, unlike its commercial competitors, Apache also provided the source code along with its server. Source code is the set of instructions that a developer uses to create a computer program. Access to source code allows an expert to change anything about how a program behaves. Products that include freely modifiable source code are called "open source." By contrast, most commercial software products are distributed only with executable code but without source code. This means that the programs can be run, but not easily changed. Having freely available source code gave Apache several distinct advantages over other providers in the web server market.

At the time Apache was released, system administrators were becoming frustrated with the problems of vendor lock-in. Once a company committed to a technology platform, the vendors realized that it would be expensive for the company to switch, and took advantage of that fact by charging exorbitant fees for support and maintenance. And the way that for-profit companies were giving away web servers made administrators nervous that such was an attempt to lock them in. By providing its source code, Apache allowed any interested company to support and maintain its products. This community ownership guaranteed that Apache users would not be unfairly locked in.

---

[2] Samuel J. Palmisano, "How the U.S. Can Keep Its Innovation Edge." BusinessWeek, November 17, 2003.
[3] Michael A. Cusumano and David B. Yoffie, Competing on Internet Time, Touchstone Press, 1998, p. 323.
[4] "IBM posts freebie Web server", CNet News.com, March 25, 1996.
[5] Josh Lerner and Jean Tirole, "The Simple Economics of Open Source." *National Bureau of Economic Research* March 2000, p. 10-11.

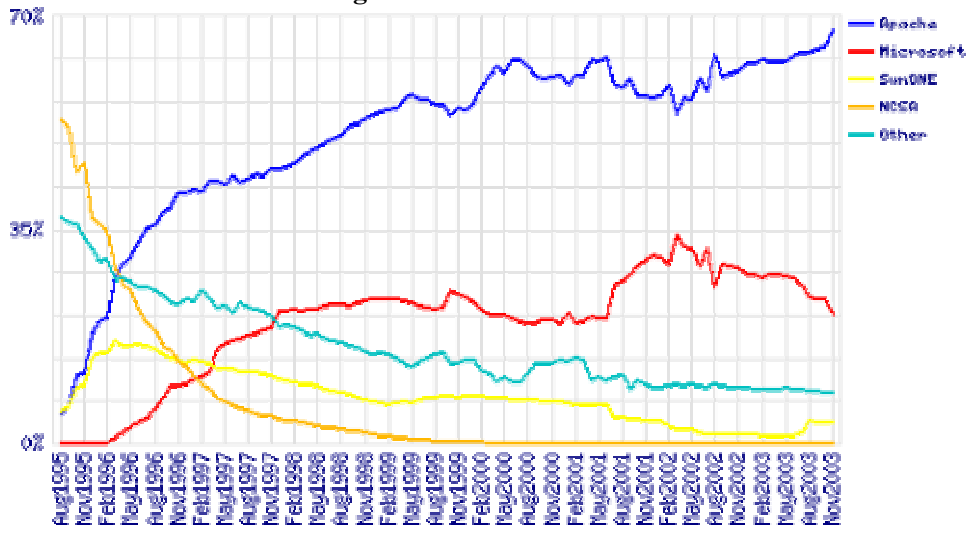
Many Apache users took advantage of their right to alter and improve Apache's software. Because many of the users were computer experts, they fixed the defects they found and added new features they needed. Most of these improvements were contributed back to the community of Apache users. As a result, Apache was able to improve quickly based on the ideas of users all over the world. This was critical to helping them innovate as quickly as their corporate rivals.

Beyond these areas of differentiation, Apache was a solid product. Web servers were judged based on their speed, stability, and ability to sustain many users. Even if Apache was not a leader across each dimension, it was certainly competitive with its competition.

System administrators flocked to Apache, which by 1996 became the world's leading web server. Once it had a lock on the market, most commercial competitors decided they were better off distributing Apache's product rather than reselling their own. In fact, IBM became a major contributor to the Apache web server project.[6] The only notable commercial competitor to remain was Microsoft, which continued to freely bundle its web server into versions of the Windows operating system.

According to Netcraft.com, approximately 68.6% of all the active web servers in the world run, the Apache web server. Over 14 million web servers run Apache server software today compared to just 4.9 million servers that run Microsoft's competing product.[7] Linux is destined to remain the world's most well know open source project because of its charismatic leader and direct threat to Microsoft. But no open source project has dominated a major market as effectively as has the Apache server.

Netcraft Market Share for Top Servers Across All Domains
August 1995 – November 2003



Source: http://news.netcraft.com/archives/2003/11/03/november_2003_web_server_survey.html

[6] "Big Blue continues freeware push", CNet News.com, October 15, 1998.
[7] Netcraft.com, http://news.netcraft.com/archives/2003/11/03/november_2003_web_server_survey.html, 12/1/2003, 8:20 pm.

Through its web server, Apache became widely known and developed expertise at managing large-scale open source software projects. It was able to apply both of these to a number of new projects, and now the Apache organization manages dozens of open source software projects. Most are either pieces of infrastructure for the web (like the web server) or are tools for software developers. Apache has also begun working more closely with partners from industry. Sun and IBM have both contributed large amounts of source code to the public by putting them in Apache's care.

The Roots of Open Source

The Apache project was not the first shot over the bow of proprietary software. The roots of the open source movement[8] are found in 1960s academia and corporate R&D labs and ironically, in the subsequent disagreements that emerged over the informal sharing of code. Institutions like MIT and Berkley with corporate research facilities like Bell Labs and Xerox's Palo Alto Research Center were highly involved in the creation of computer operating systems and often freely shared their developments. Numerous cooperative development efforts emerged in the early 1970s. Some of the most successful were at AT&T's Bell Labs which developed the UNIX operating system and the C language. The Unix operating system and the C language were shared with many other institutions who "made further innovations, which were in turn shared with others. The process of sharing code was greatly accelerated with the diffusion of Usenet, a computer network begun in 1979 to link together the Unix programming community. But the sharing and "informality proved to be problematic in the early 1980s, when AT&T began enforcing its (purported) intellectual property rights related to Unix."

Part of the impetus behind the increased claims on open source software was the transition of information technology value creation from the 1960s to the 1980s. In the 1960s, the real money in computing was made selling hardware. Given the enormous expense, the software was viewed as relatively cheap and thus, computer scientists and corporations willingly shared software code. By the 1980s, the value creation was increasingly software-related as the real costs of computing tumbled with advances technological innovation. Now the money was in software, and in arguments over ownership and rights, "free" software gained increased visibility.

With the rules of cooperative software development now in question, a new order was sought by Richard Stallman of MIT's Artificial Intelligence Laboratory. He founded the Free Software Foundation (FSF) in 1983 which introduced a formal open source GPL license (General Public License) by which the user of the open source GNU[9] software "had to also agree not to impose licensing restrictions on others. Furthermore, all enhancements to the code…had to be licensed on the same terms." The advent of the Internet greatly expanded the reach and power of the open source movement. This, coupled with other open source licenses, including the Debian Social Contract license, allowed users of open source code to make improvements but prevent

[8] There is, however, an argument to be made that the roots of open source go back even further. See **Appendix 1: The Old New Thing.**

[9] "Self-referentially, short for "GNU's not UNIX," a UNIX-compatible software system developed by the FSF. The philosophy behind GNU is to produce software that is non-proprietary. Anyone can download, modify and redistribute GNU software. The only restriction is that they cannot limit further redistribution. Linux systems rely heavily on GNU software and in the past, GNU systems used the Linux kernel. This close connection has led some people to mistakenly equate GNU with Linux. They are actually quite separate. In fact, the FSF is developing a new kernel called HURD to replace the Linux kernel in GNU systems." Source: Webopedia.com

commercial interests from exploiting them. Because these licenses are designed as a subversion to the commercial aspects of copyright law, they are entitled "copyleft" licenses.[10] Linux is offered under such a license agreement.[11] By contrast, Apache rejects copyleft in favor of licenses that allow companies to improve and resell their products.

The Economics of Participation: Does the "Create value-Capture value" Model Apply?
It is easy to see the appeal of software that has no cost, does not lock users in, and is easy to improve. It is much more difficult to comprehend why anyone would want to write a complex piece of software for which they are not being compensated. In understanding this, it is important to note that the cost of the tools and equipment needed to be a software developer has plummeted. Desktop computers have become both cheap and powerful enough to be used for projects like Apache's. The software tools that are needed to contribute to the project are freely available. The Internet has made it almost free for anyone to get access to the source code and distribute changes back to the community. Thousands of users are eager to test any improvements that are made to Apache's products. So the only significant cost is that of the developer's labor.

As it turns out, many skilled software developers are eager to spend their time contributing *gratis* to projects like Apache's. Jonathan Barnes cites the popular mistrust of "big corporations," the ability to receive quality feedback for a low cost, the ability to signal talents and abilities to the marketplace, and finally, the artist's prerogative as the principal motivations for individuals to participate in open source software (OSS) development. Of all the motivations, the artist's prerogative appears the most fundamental; developers seem driven by a desire to display or "show off" their work.[12] The instant offer of personal recognition that OSS provides is a powerful motivator. Eric Raymond echoes this sentiment in proposing that developers contribute code driven by "combination of reputation and ego arising from creating and distributing a software solution."[13] He further avers that three main benefits accrue to successful contributors of open source projects: "good reputation among one's peers, attention and cooperation from others, …[and] higher status [in the]…exchange economy."[14] Another powerful driver of participation is a developer's need to "scratch their itch." When a developer needs a web server for a new process or task, the individual simply downloads the open source software and improves it for her own benefit. Uploading the improved copy back to the OSS site is simply a customary community norm reinforced by the availability of the free software in the first place. In the most elementary economic analysis, "as long as the benefits [of participation] exceed the costs, the programmer is expected to contribute."[15]

[10] *Copyleft* is a general method for making a program free software and requiring all modified and extended versions of the program to be free software as well. For a more detailed explanation, see www.gnu.org.
[11] Josh Lerner and Jean Tirole. "The Simple Economics of Open Source." *National Bureau of Economic Research*. March 2000, p.4-7.
[12] Jonathan Barnes. "Open Source As A Computer-Specific Organisational Technology." 2003, p. 10-11.
[13] Asif Khalak. "Economic Model for Impact of Open Source Software." November 21, 2003, p. 2.
[14] Josh Lerner and Jean Tirole. "The Simple Economics of Open Source." *National Bureau of Economic Research*. March 2000, p.20.
[15] Lerner and Tirole cited by Karim R.Lakhani and Robert G. Wolf. "Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects." September 2003, p. 6.

3.2 Description of firm activities/actions/policies, including why the firm undertook these activities/actions/policies.

Early Years: Early Organization of Apache
The early organization of Apache was small, democratic, and entirely digital. The founding members utilized email lists as the exclusive means of communication and employed a quorum voting system to resolve conflicts. Any member could vote on the inclusion of new code in the system and coders desiring membership had to contribute to the project for approximately six months and then be nominated for membership by an existing member. Smaller groups of members served as core developers on various aspects of the project; the project had fifteen core developers. The software was architected in a series of individual modules which facilitated the division of labor and increased ownership and motivation by the core developers on that module. While there were only fifteen core developers on the project, approximately 400 developers contributed code that was incorporated into the software. Apache utilized a bug tracking tool named BUGDB to track all of the problems reported by users and or testers. Over 3,000 individuals submitted almost 4,000 problem reports via the BUGDB.[16]

Centralized Decentralization
While the testing population was quite large, the majority of the code was authored by a very select group. Research indicates "that the top 15 developers contributed more than 83% of the MRs [modification requests] and deltas [changes], 88% of added lines and 91% of deleted lines. Very little code, and presumably, correspondingly small effort [was] spent by non-core developers."[17] One of the keys to the success of the Apache project was the highly decentralized code review process coupled with the tightly controlled code change process. This process produced a product with superior performance and market adoption than did traditional proprietary software concerns, i.e., Microsoft.

Recent Performance
Although a significant portion of the Open Source programming community defines itself in opposition to the world of corporate interests and proprietary code, Apache is notable for the fact that is has enjoyed enormous corporate support. Working with code donated by IBM, also prominent for its use of Linux-based programming, Apache has moved beyond server software that delivers basic web pages towards creating software that produces significantly more complicated web pages using XML (eXtensible Markup Language). Entitled *Xerces*, this software extension is vital to the continued development of ever more sophisticated and robust Internet applications. The most recent version was made available in November 2003.

Apache has also benefited from a collaborative relationship with Sun Microsystems. Sun's Java programming language is perhaps the most widely accepted *lingua franca* for building the web pages that populate the web. Beginning in March 2002, Apache launched the Jakarta Project to create and maintain open source solutions on the Java platform for distribution to the public at no charge."[18] The hallmarks of the agreement include the following[19]:

[16] Audris Mockus. et al. "A Case Study of Open Source Development: The Apache Server." 2000, p. 3-5.
[17] *Ibid.*
[18] http://jakarta.apache.org/
[19] http://jakarta.apache.org/site/jspa-agreement.html

1. The right to freely implement specifications in open source.
2. The right for specification leads to release reference implementations and test kits in open source.
3. The right for specifications to be created more publicly.
4. The right to free access to test kits by open source, non-profit, and academic groups.

One can scarcely imagine Microsoft signing a similar agreement. The product of this collaboration was the *Tomcat* extension which runs Java programs on servers.

More recently in August 2003, Apache launched the Geronimo Project[20] to create software based on Sun's J2EE, Java 2 Platform Enterprise Edition. J2EE is a platform-independent (i.e. it can run on a Windows, Linux, UNIX-based or any operating system) programming language for developing, building, and deploying Web-based Internet applications online. In short, companies engaged in the transition from Old Economy to IGC will undoubtedly be relying on such J2EE applications, whether the CEO knows it or not.

ApacheCon: Shall We Gather at the River?

2000 witnessed the first annual face-to-face meeting of Apache developers. Sponsored by IBM, Sun, Fujitsu-Siemens, and a host of smaller open source companies, ApacheCon ("Conference") provided the opportunity for a dispersed community of once-faceless programmers to begin to learn how to communicate, not electronically, but in person. After spending late nights in the hotel bar writing code together, ApacheCon participants left the conference with a better understanding of their electronic colleagues' personalities and email writing styles. The most recent ApacheCon was in November 2003.[21] Although it does not have the widespread name recognition of Linux, it is not an overstatement to claim that Apache is at the forefront of some of the most important advances in Internet technology.

What Defines Apache?

Between Apache's loose organization and broad mix of offerings, one might wonder if by now it is little more than an unrelated product portfolio sharing a common name. So it is worth investigating exactly what Apache brings to its projects that enhances their success. When asked about what characterizes the group's projects, Apache board member Mark Cox cites two characteristics. The first is the process that Apache has developed for managing projects, which includes a product lifecycle managed by a small project management group. The second is Apache's business-friendly open source license, which distinguishes its projects form the more restrictive licenses of Linux and many other open source projects. It is notable that Apache does not provide any strategic planning resources to its projects, and it does not limit itself to working with one type of software. Rather, it depends on the projects to guide themselves based on the developer's interests and needs.

[20] David Becker, CNET News.com, August 7, 2003. http://news.com.com/2100-1012-5061303.html
[21] http://apachecon.com/2003/US/

Of Lawyers and Licensing
Software licenses typically involve the minutiae of intellectual property law, and it would seem unusual for a conventional software company to claim some text written by a lawyer to be a core part of its identity. This highlights the field in which Apache competes most intensely. It is not a battle over adoption of any particular product; rather it is a war of ideas over what rights users should have over software. This explains why Cox defines the biggest threat to Apache as not being a loss of customers or revenues, but as losing the interest of the programmers working on the projects. Apache's license represents a *via media*, or middle path, between commercial software licenses that grant users limited rights to modify software, and copyleft licenses that limit companies' ability to profit from improving software. These conflicts have been witnessed recently in areas such as J2EE applications server software, in which commercial, copyleft, and Apache offerings all compete.

Performance Metrics, Revisited
Apache defines success based more on its software's influence than on its overall adoption rate. Cox emphasizes that Apache never set out to have the most popular server; instead it was concerned with ensuring that important standards were correctly supported. This is why Apache was unwilling to modify its server to support Internet providers who did not "play by the rules," even the massively influential AOL. Apache could not have its influence without its widely used products. But building market share as a means of exerting positive influence is a contrast from most commercial entities, for which influence is the means and market share is the end.

3.3 Justification of these activities/actions/policies as typifying an emerging/ established or an aspiring IGC exemplar.

> *"In an abundant world of knowledge, not all smart people work for you. The next big idea may spring from a maverick start-up or a researcher working for a competitor. For many companies in the innovation business, the response has been to circle the wagons tighter still, wall off their own R&D effort to keep competitors from stealing their best ideas."*[22]—Henry W. Chestbrough

The Archetypal IGC
The story of most IGCs involves companies that take an existing business model (like retailing or manufacturing) and revolutionize it by network-enabling all aspects of their operation. In contrast, Apache came into existence on the Internet, which is such an integral part of its operations, products, and competitive advantage that it is impossible to imagine an unnetworked Apache. As a result, Apache not only demonstrates the characteristics of an IGC; it ends up serving as an extreme example for many of them.

Apache's primary business functions involve developing and distributing new software products. These activities involve the interaction of hundreds of developers and thousands of customers, all of which takes place online. Even more impressively, Apache has developed the capability to effectively manage its projects and personnel over the Internet. The entire lifecycle of a product,

[22] Henry W. Chestbrough, "A Better Way to Innovate." Harvard Business Review, July 2003.

from conception to completion, can take place without a single phone call or face-to-face meeting. The level of network-enablement that many conventional businesses strive for has been a part of Apache since its inception, and is critical to Apache's large-scale yet decentralized projects.

A Transparent and Open Community: No Inside Jokes, Please

The intensity of Apache's reliance on the Internet is only one aspect of what makes it so radically network-oriented. Having an entirely online business allows Apache to make itself completely open to the outside world. The most famous instance of this openness is Apache's willingness to share its source code. In a conventional commercial software development company, source code is generally the crown jewel of its proprietary intellectual property, and it is protected as intensely as the formula for Coke. By contrast, Apache utilizes an open source model, where the source code for its products is made freely available for any users to inspect or improve.

Many Hands Make Light Work

Apache's openness extends beyond its source code. Every scrap of knowledge created by every Apache project is freely available on its website. Users can inspect the current status of projects, see detailed designs of future releases and determine whether they are on schedule. And while commercial software companies derive their advantage from the propriety of their information, Apache's openness allows its customers to clearly gather information about the state of Apache's projects and to ensure that they are not "locked in" to a single vendor. Furthermore, Apache's openness allows it to shift some of the burden of software development onto its user community. A major component of the cost of building software involves exhaustively testing it and auditing the source code for security defects. In Apache's case, anyone can test out pre-release versions and evaluate its security, which alleviates the burden on Apache developers. As IGCs become more network-enabled, they tend to use networks to expose specific business processes to partners and customers. But few are in a position to take the benefits of openness to the same extreme as Apache.

(Un)bundles of Joy

An additional trend associated with IGCs is a move towards unbundling of business processes. The reduced interaction costs of operating on the Internet allow companies to focus on one aspect of delivering a product to customers, while depending on partners to fill in the others. Network-based coordination allows these companies to remain focused on their area of expertise, while presenting a complete and coherent single product to customers. Apache fits this mold by focusing strictly on developing and maintaining its products, and eschewing the customer-intensive tasks of sales and support. Apache represents a particularly dramatic case of unbundling, because it lacks formal recognition of, and interactions with, its selling partners. Unlike a commercial software company that contracts with partners to retain royalties of resold software and limit the ways it is sold, anyone interested can sell and support Apache products. And while these vendors can contribute improvements to Apache's projects, they lack any official designation or relationship with Apache itself. This allows distribution and support of Apache's products to be served by a competitive market and frees the Apache organization from concerns about distribution. Unlike Apache, most of these distributors are traditional for-profit companies. So Apache is able to unbundle its business not by contracting with another

organization to help deliver its product; rather it is able to take advantage of a host of organizations with which Apache has no formal relationship.

Apache consistently applies the principles of running an IGC in radical ways. As such, studying Apache helps us understand either where today's IGCs could end up or what separates their progressive approach from Apache's more radical all-digital organization.

3.4 Impact of the activities/actions/policies on the firm and its competitors, and the competitive position of the firm in its chosen market.

Apache's rise to become the dominant vendor of web-server software despite strong commercial competition shocked the software industry. Although open source software has been around for almost as long as computers themselves, these products have traditionally been of a modest scale and managed by a single developer or a small group. As a result, it seemed unlikely that the open-source community could create large scale products that compete with privately developed ones based on features and quality. In fact, open source software was generally used in markets too small to be served by any commercial vendors. The Internet both expanded the development capacity and competitive advantage of open source projects. The advent of Apache not only completely reshaped the market for web servers, but it had broad implications for how pieces of the Internet's infrastructure would be built and distributed. "The Web application server market plump with a history of high margin Internet deals is retrenching to focus on a quicker return on the dollar and lower life cycle operating costs."[23]

Commercial web servers were marketed based on their price, speed, ability to support a large number of users, and the tools they provided to build complex and interactive websites. Although Apache competed with commercial vendors in all of these areas (most notably price), its most significant differentiation came from the "openness" of the product and the organization. It is notable that Apache's primary competitor today is Microsoft's Internet Information Server (IIS). It is provided to users of Microsoft Windows at no additional cost, so it can be considered just as free as Apache is. Yet it does not provide the community with access to its source code or development plans. As a result, users have been migrating over to Apache, which has been accelerated by the heightened concern over Internet security. Users are now reluctant to accept pieces of the Internet infrastructure which are closed, which is why companies increasingly donate new products to Apache rather than try to sell or give them away. Notable examples of this include Sun's *Tomcat* server and IBM's *Xerxes* XML technologies, as mentioned above.

These donations highlight Apache's friendliness to business, which is another key innovation. Although Apache's products have displaced a number of commercial competitors, the organization is designed to work with businesses to develop and improve software products. This is a contrast to the early open source community, which was rabidly anti-business. They wanted all software to be open source, and wanted nothing to do with commercial software vendors. This is the community that developed the GPL, the business-unfriendly license which underlies many popular open source products, including Linux. Apache intentionally rejected

[23] Timothy Dyck. eWeek. January 6, 2003. EWEEK LABS; Pg. 35

this path. They adopted a license that makes it easy for businesses to work with, improve, and sell Apache software. As noted earlier, Apache has also been willing to work with businesses who want to contribute pieces of their software to the open source community, by combining Apache's expertise at, and credibility in, managing open source projects with a large company's resources. This willingness to work with business has not only been part of Apache's success, it has also educated the business and open source community about the benefits of collaboration with one another. This helped the open source community move beyond its anti-business origins, and today many major open source projects, including Mozilla, OpenOffice and KDE involve a substantial collaboration with businesses.

4 Conclusion

4.1 If an exemplar IGC what does the example/illustration teach us about the management of IGCs and the execution of strategy in the current competitive environment?

Lessons Learned

Competing on Openness

For years, computer software has been considered a more attractive investment than computer hardware because of the network effects and entry barriers that make software products less prone to becoming commoditized. But in the markets that Apache serves, this analysis becomes less applicable. In the web server market, users' needs are straightforward, open standards check network externalities, and Apache's extensive network of developers and testers reduces entry costs. This has allowed Apache to commoditize this segment of the software market by building a satisfactory product and distributing it at Apache's marginal cost, which happens to be zero.

It is a mistake to conclude that Apache's entire offering is based on commoditizing existing software. As discussed earlier, Apache's competitive advantage relative to other software providers is the openness of its entire business. Thus Apache has managed to change the rules of competition in the software industry from being just about making high quality and feature-filled products, to making a business open and encouraging customers to participate in it. This seismic shift is rooted in the availability of public networks. Apache's open source model allows it to be one of the most open businesses imaginable, impossible for a commercial organization to emulate. An organization's ability to provide the openness demanded by its users may become its most important strategic differentiator.

Competing on Organization

The rapid emergence of a loosely organized collective in the world of commercial software has demonstrated the power of the Internet to allow individuals to organize in new ways. Before the Internet, it would have been impossible for a group of Apache's scale to coordinate it activities other than through a large hierarchical organization. Today, the distributed organization of Apache is not only sufficient to keep it functional, it is also critical to governing dozens of projects and keeping them flexible enough to adapt to changing market conditions.

Historically, organizational structure has been seen as a product of a company's industry and its broader culture. As a result, internal organization has not been a major competitive feature among companies. But as the Internet provides people with more creative options by which to organize themselves, companies could end up with competitors whose major distinction is their mode of organization. This is exactly the kind of strategic shift that entrenched companies tend to overlook until it has overtaken them.

Unbundling through Markets

The current trends in corporate unbundling help us see how networks enable firms to become smaller and more focused. Most of these examples include contractual relationships and a high level of interaction among firms. One of Apache's advantages is that the product development organization is totally disconnected from (and indifferent to) its sales mechanisms. This allows the organizations interested in selling the product to compete freely and provide end users with the highest quality experience. As one might expect, Apache is a special case that would be hard to imitate. But given the high costs of monitoring and controlling firms with which one has a contractual relationship, companies would do well to consider replacing their one-partner contracts with market relationships comprising many competing partners. For example, instead of contracting with a sales organization, a software company could offer a fixed commission to anyone who wants to sell its product.

From Open Source to Open Innovation through Internetworking

Another lesson is that learning and innovation are accelerated by sharing and that such an "open phase…is necessary for adapting new technologies to useful purposes."[24] This suggests that firms undertaking frontier research might increase their eventual efficiency, accelerate the viability of new technologies, and thus fast-forward the development of new industries and markets by collaboration and sharing (see **Appendix 2**: "Lessons for the Private Sector: A Hybrid Approach to Internetworked Innovation"). "The collaborative development model popularized by the open source movement has far-reaching implications for commercial application development. Any development organization can benefit from a greater understanding of what made projects like Linux and Apache successful."[25]

[24] Peter Meyer. Episodes of Collective Invention. *BLS Working Papers U.S. Department of Labor*. Aug. 2003, p. 4.
[25] John Desmond, Editorial Director at Software Magazine. PR Newswire. September 12, 2000.

Appendix 1: Open Source—The Old New Thing

While the roots of open source appear to begin the 1960s as computer technology was emerging, the practice of sharing knowledge for the sake of advancing a new technology is much older. Robert Allen called the "free exchange of information about new techniques and plant designs among firms in an industry" collective invention. Allen notes that blast furnace owners in Britain from the 1850s through the 1870s willingly opened their facilities to review and inspection by visitors and consultants and accelerated the design and effectiveness of the furnaces during that time. The improvement of the design of the steam engine was facilitated by the practice of collective invention by Cornwall, England miners after the design went off patent in 1811. "Nuvolari (2000) establishes that the efficiency of steam engines improved substantially in Cornwall through this period, probably through many minor or unattributed innovations and discoveries by steam engine engineers." Tushman and Anderson suggest the "long period of time after an invention but before a substantial industry has stabilized around it a 'period of ferment'" and reference the Bessemer steel-making process, the advent of microcomputers and open source software as examples of technologies that have all required a gestation period. **(Source: Peter Meyer. Episodes of Collective Invention. *BLS Working Papers U.S. Department of Labor.* August 2003, p. 3-5.)**

Appendix 2: Lessons for the Private Sector: A Hybrid Approach to Internetworked Innovation

Company	Open Innovation Experiments	Rationale and Results
Importing Innovative Ideas		
Intel	Intel has opened four small scale research laboratories, or "lablets," adjacent to universities in the US and Britain to promote cross-pollination. A university professor bankrolled by Intel runs a lablet for two years, then returns to teaching so another faculty member can rotate in.	Intel gains informal access to a wide variety of faculty networks in a systematic way; faculty running the labs gain insight into Intel's R&D process. Products have yet to be developed by the newly opened lablets.
Eli Lilly	Lilly recently launched InnoCentive, an online knowledge broker. Lilly and other firms post R&D problems on the site and solicit solutions from individuals and companies worldwide.	Lilly has received over 200 proposed solutions from visitors around the world, including scientists in China and Russia. The company has paid more than a dozen "solvers" for their proposals.
Toy makers and retailers	Mattel, Wal-Mart, and other toy manufacturers and retailers use idea brokers like Big Idea Group to scout on their behalf for new toy ideas. Big Idea Group invites inventors to submit ideas and then refines them and pitches the promising ones.	Big Ides Group has placed a number of toys with companies like Basic Fun—which bought TinyTotes, a line of miniature fashion handbags—and Gamewright—which bought the games Snap and Fowl Play.

Source: "A Better Way to Innovate." Henry W. Chestbrough, Harvard Business Review, July 2003.