ABSTRACT

Despite the presence of network effects and standardization, both of which tend to enhance market power, the computer industry remains largely competitive due to the ameliorating effects of coopetition and demand for complementary products to platforms. While industry innovation is by no means driven by the dominant firms that control the main platforms, it is uncertain whether the presence of these firms prevents innovation from reaching an efficient level. This paper examines the strategic interactions between firms that can stifle coopetition and allow the dominant firms to consolidate the development of complementary technology. It is found that a partner of a dominant firm cannot develop technology independently or form alliances with other firms to do so unless the dominant firm decides not to pursue the same technology.

Keywords: coopetition, complementary technology, computer industry, innovation

*I would like to offer my most heartfelt gratitude to Prof. Tim Bresnahan for his constant and patient guidance, insightful questions, and encouragement throughout the entire writing process. Special thanks to my friends and family and, of course, to Prof. Geoffrey Rothwell for their support.*
1. Introduction

In the computer industry, two aspects of industry structure are by and large responsible for maintaining the high level of innovation and competition between firms. First, it is accepted practice for firms to partner with each other in developing and marketing one technology while competing against each other in another technology. This firm behavior, commonly known as coopetition, is believed to be a primary reason for the competitiveness of the computer industry. Indeed, an alliance between high-tech firms, in theory, does not preclude either firm from developing an innovation with another party should doing so prove more profitable than expanding the existing relationship to cover more technologies. Thus, the competitive and technological inefficiencies that result from one firm controlling the rate of innovation in an industry are often avoided.

The second aspect is the ability of more than one firm to influence the development of a platform. While a platform can be an exclusive standard (compatible technologies can be developed only with a license and some standards remain incompatible with each other), it is usually not in a firm’s interest to keep out complementary technology. That is, it is more efficient for a firm that controls a platform to expand its user base by relying on other firms to develop complements to the platform. Therefore, the rate of innovation that results is more efficient than that which occurs when the components of a platform can only be developed by one firm.

However, the presence of complementors and coopetition by no means guarantees enhanced competition and innovation. In US vs. Microsoft, it was revealed that Microsoft believed that Netscape Navigator and Sun Microsystems’ Java, both complements to Windows, to be competing technologies. Court documents also showed that Intel’s attempt to cooperate with Sun instead of Microsoft in the Multimedia (MM) sector was also stifled by the latter.\(^1\)
Assuming that Microsoft was not acting irrationally, what must be the additional mechanisms behind the complement/competitor duality and coopetition that caused both to have anti-competitive effects? Furthermore, what factors decide whether these mechanisms are more influential than those that lead to pro-competitive effects?

With regard to the complement vs. competitor question, this paper hypothesizes that the ambiguity of whether a specific technology from one firm complements or competes with a technology from another firm drives the firm interactions that determine whether a relationship stifles or encourages innovation. This ambiguity is chiefly due to the fact that while a complementary technology cannot threaten a firm’s profits at the time it is first released, it can prevent the firm from reaping profits in the future from developing its own version of a similar technology. In short, a complement can weaken the future growth of a firm, thereby presenting itself as a potential competitor as well. If a firm chooses to identify a technology strictly as a complement, then it will be developed, perhaps even jointly. If the technology is identified as competition, then firm interaction will work to stop its creation.

The answer to the question of how coopetition can lead to inefficiencies becomes more clear when we consider the relationships of all firms involved in an innovation, not only the firms with the existing partnership. To go back to the Microsoft example, what is particularly puzzling is not that Microsoft was opposed to an MM technology it did not develop, but that Intel chose to work with Sun instead of its much more powerful long-time partner. Given that Microsoft has the largest distribution channel and the dominant position in the market, why did Intel choose to partner with another firm inferior in both aspects? This paper posits that in a partnership with a dominant firm, the weaker firm cannot control the terms by which the two collaborate unless it has other innovations whose profitability is not dependant on the strength of this partnership. A third party, even one weaker than the dominant firm, is a potential partner to
develop these innovations. However, a dominant firm can ultimately leverage its market power and prevent the formation of alliances between its existing partners and other firms, even if it is more technologically efficient to allow these alliances. With its superior bargaining power and the greater worth of the existing partnership between itself and the weaker firm, the dominant firm can stifle innovations and other firm collaborations that can challenge its position.

To establish a context for the discussion, two representative cases and their implications for the understanding of the questions this paper seeks to answer are examined. The first is that of the relationship between IBM, Hercules Corporation, and Lotus with regard to the video adapter technology in the early to mid-1980s. The second has already been introduced—the strategic interaction between Intel Corporation, Microsoft Corporation, and Sun Microsystems fuelled by the advent of MM technology. This latter case serves as the primary basis for the model of firm interaction developed in this paper. Through representing firm relationships as games, the model seeks to understand the underlying mechanisms that determine the effect firm interactions have on competition and innovation in the computer industry.

Section 1 of this paper contains a literature review of studies done on the same and related topics. Section 2 provides a historical overview of inter-firm tensions in the computer industry and presents the two cases mentioned above. The main model that explains these case studies is developed in Section 3. A discussion and brief conclusion follow in Section 4.

2. Literature Review

Studies on the computer industry have thus far relied on standards theory as a context in which to develop theories about industry performance and firm behavior. This is not surprising considering that standards are the analogue to the platform in the computer industry. Though this paper focuses largely on firm interactions rather than the entire computer industry, similar
mechanisms govern the operation of both. As well, the complement versus competitor issue is fundamentally based on platforms and their influence in determining properties of market structure. Thus, I will first survey the literature at the platform level and then establish its connections with the more recent studies on inter-firm behavior.

The seminal study on the market structure of the computer industry remains the historical analysis conducted by Bresnahan and Greenstein. Their paper posits that a unique trait of the computer industry is that though platforms persist, the production of their components is controlled by different firms. Since a platform, defined as a cluster of technically standardized components that buyers use together with other components to make applications, is the computer industry’s analog to the standard, it is no surprise that the same forces that prolong the dominance of a standard also do so with a platform. However, the vertically disintegrated nature of the computer industry allows multiple firms to influence how a standard evolves, and no single firm is able to become a monopoly based solely on its control over a standard. Second, the computer industry is relatively young and its rate of innovation is unusually fast. The new technologies developed cater to pockets of demand whose needs have not yet been met by existing innovations. Given time to expand, these market segments not only produce complements to the prevailing standard, but in doing so nurture the development of another platform. By way of its complementarity with the dominant platform, the firms in this smaller segment already gain a foothold on the larger market. Behind the force of a growing installed base and increasing completeness of its platform, these firms could theoretically introduce the new platform into the larger market and challenge the dominant platform (Bresnahan and Greenstein 1999).

Since Bresnahan and Greenstein, much work has been done on identifying the market mechanisms, in light of standards theory, that make these two key phenomena possible. Most
theoretical and empirical work has confirmed what is already known about the industry: entrants face high barriers to entry in the market for the dominant platform only. The abundance of other market segments and the interconnectedness of the industry’s products allow these entrants to service an industry niche in the short run and challenge the dominant platform in the long run. Spurred by the dot-com boom of the late 90s, most studies have focused on the start-up as the representative entrant. The game theoretic models created largely involve the entrant as the first-mover and the incumbent as the respondent. The outcome of these games is that the entrant either partners with the incumbent to supply a technology or stays out of the market.

In the industry, both demand and supply forces have lowered entry barriers for firms, but not platforms. Namely, platforms exhibit divided technological leadership, a structure in which a number of firms possess the capability to supply key platform components. As such, firm entry becomes much easier than the case in which a single firm controls the development of a platform (Bresnahan and Greenstein 1999). The “competitive crash” of the 1990s is largely responsible for this phenomenon. Because of the vertical disintegration of the industry following the fall of IBM in the 1980s, platform component supply markets became inhabited largely by specialized firms. Hardware and software markets became more competitive. Innovation not only improves computer hardware and software in platforms that service already-recognized groups of demanders, but it also occasionally invents new platforms to serve an entirely new body of demand.

Nevertheless, the basic forces of endogenous sunk costs and platform persistence still underlie the industry’s structure. The theory of indirect entry reconciles these two conflicting observations. It postulates that when a platform is young and without complementary product suppliers and an installed base, it tends to flee competition with an existing platform. Instead, the platform broadens the range of technologies and services so that it opens up new segments of the
market for itself. Entry into a high-rent, established platform segment will occur when there is a technically close, but competitively distant, incubator segment. The resulting industry equilibrium is characterized by rapid technological changes initiated by specialized, vertically disintegrated firms, which in turn devalue platform-steering expertise (Bresnahan and Greenstein 1999). Competition is further assisted by the fact that incumbents are constrained in meeting potential competitors directly because of a need to maintain backward compatibility for existing consumers. Observations also suggest that a new vector of entry threats to established firms comes from suppliers of complementary products (Greenstein 1998).

Evidence that industry incumbents were aware of this threat could be found in their attempts to expand or redesign the functions of their existing platforms such that outside firms would not be as able to challenge their dominant position through complementary applications. Davis, MacCrisken, and Murphy neatly summarize the new aims of software designers by naming four forces that shape the design of operating system products and determine their evolution. The first of these forces is the need to ensure that the development of applications software keeps pace with rapid innovation in other computer technologies and to make the operating system more user-friendly. Next is the need to maintain compatibility with existing applications while ushering in additional functions that support new applications. Finally, software designers are attempting to meet the demand for complementary applications on their own, and thereby consolidate the product valuations among consumers, by bundling multiple software features into a single package. A less important factor driving design is the desire of firms to reduce customer support costs (Davis, MacCrisken, and Murphy 2001).

Despite finding support for increasing control over a software platform by a single firm, the paper maintains that designs deemed anti-competitive should not be regulated. First, circumstances that give rise to anti-competitive product designs also give rise to other harmful,
anti-competitive strategies. Thus, it is not clear that product design restrictions can prevent or ameliorate all or even the most harmful anti-competitive conduct. In the same vein, it is difficult to distinguish anti-competitive product designs from pro-competitive designs or to determine whether the harm caused by an allegedly anti-competitive design outweighs the benefits. Namely, legal restrictions on designs can impede beneficial forms of integration and bundling (Davis, MacCrisken, and Murphy 2001).

Interestingly, many of these innovation concerns are largely balanced by the need for firms to form alliances in their design efforts. Chellappa and Saraf conducted an empirical analysis of the Enterprise Systems Software (ESS) market and found significant encouragement of cooperation between firms. Since ESS firms engage in competition with one another in many different sectors, and sufficient opportunities exist for leveraging resources across markets, the paper examines the market through the perspective of multi-market contact and economies of scope theories. In particular, when two firms simultaneously operate in two or more markets, the mutual recognition of their interdependencies might lead to cooperation, or mutual forbearance. But instead of engaging in forbearance, firms may end up competing intensely as they try to leverage these advantages (such as economies of scope). On the other hand, consumer driven externality advantages are beyond the control of the ESS firms except for two strategic actions: offer a larger set of components or increase their attractiveness through alliances (Chellappa and Saraf 2003).

Ultimately, the unique nature of the ESS market is such that while component markets are separate, business consumers that buy these products need the components to work together. Product compatibility is achieved through a complex alliance network. Furthermore, a firm’s prominence in this network drives its performance because it corresponds to the overall market’s perception of its compatibility with all others in the market, and thus to the ability to extract
externality benefits from consumers. Not surprisingly then, the rivalry between firms is not a consideration in the formation of two-firm partnerships. Firms do not avoid rivals when joining an alliance because the potential for extracting indirect externalities (being attractive to a rival’s user base) outweighs the possibility that one’s own user base might be threatened by the rivals. The actual formation of the alliances is informed by two factors: (1) there are no distinctly superior partners such that only aligning with them will suffice, hence a firm’s partnership considerations cannot be limited to market leaders or leading coalitions; and (2) if business consumers adopt a “best of breed” approach rather than a uniform corporate standard, the only way ESS firms can make their non-best of breed components attractive is to ensure that the components are compatible with the best of breed ones (Chellappa and Saraf 2003).

The alliance formation of ESS firms is emblematic of another major characteristic of the computer industry—though platform competition remains fundamental, coopetition has become the trademark of firm behavior. While coopetition is an embedded and almost mundane feature of firms with established technologies, emerging technologies offer interesting insights into the effects of coopetition on the innovative and competitive dynamics in this industry. Namely, coopetition in emerging technologies is motivated by the coopetition among “networks of innovators.” This network is defined as a group of firms, consisting of competitors as well as suppliers that decide to cooperate because they are in competition with other networks of firms in developing new technologies or setting new standards. The decision to cooperate with competitors is affected by the firms’ desire to access and exchange new technologies and complementary knowledge, stimulate or enter new markets, and influence or control technological standards (Garraffo 2000).

In the smart card industry, cooperation is designed to serve all parties in building and sustaining competitive advantage and bundling resources and competencies. At the same time,
the smart card industry is characterized by technological change, price wars, standards conflicts, and competition for market share. Therefore, interactions between smart card actors in both vertical and horizontal dimensions combine cooperative and competitive attitudes. Contrary to intuition, increasing technological, financial, and market barriers discourage players to compete with each other and encourage them to work cooperatively to accomplish their shared goals (M’chirgui 2004).

Other research in the structure of the computer industry has focused on specific segments, the advent of the client-server platform, and start-up companies. Currently, the client-server setup takes full advantage of vertical disintegration in the PC market. Until standards are clearly defined for this setup, threats by buyers to use different combinations of components to form client-server arrangements may provide a competitive threat for existing combinations (Greenstein 1998). De Figueiredo and Kyle conducted a survey of firms that supply laser printers and found that the dominant firms of the market are very selective about the markets they pursue. They often stay out of the most innovative markets and choose to enter markets well behind the innovative frontier set by fringe firms. Contrary to theory, however, entry of the dominant firm is followed by a wave of entry by other fringe firms. The exit rate increases, but this increase is offset by an infusion of entrants that come in after the dominant firm (De Figueiredo and Kyle 2004).

In the computer industry, the most common type of entrant is the start-up, which has been largely studied in the context of intellectual property rights (IPR). Early work has suggested that environments in which IPR are weak, a dominant incumbent would prefer to take advantage of the R&D productivity of smaller firms. Therefore, established firms may be motivated to develop a reputation for “non-expropriation” in order to provide incentives for innovation and cooperation by start-ups (Gawer 2000). Recent studies of start-up behavior reinforce this finding.
Specifically, if patents are effective in protecting IPR, firms face high relative investment costs, and brokers are available to facilitate trade, then start-up innovators tend to earn their returns from innovation as an upstream supplier of “technology” rather than as a horizontal innovation-oriented competitor. However, when investment costs for the entrant are relatively low and technological innovation is not protected by patents, then start-up innovators are more likely to commercialize their innovations through product market competition (Gans, Hsu, and Stern 2001).

3. Historical Overview

When IBM first introduced the PC, there were only two display adapters available—the IBM Monochrome Display and Printer Adapter (MDPA) and the IBM Color/Graphics Adapter (CGA). The monochrome adapter was intended for professional use. It produces a clear, easy-to-read text display but no graphics. On the other hand, CGA displays graphics but does not have as fine a resolution as the monochrome adapter.²

In 1982, the Hercules Graphics Card (HGC), which has the ability to produce the same text display as the Monochrome Adapter as well as high-resolution monochrome graphics, was released onto the market.³ The HGC garnered the early support of important business applications, the most notable of which was the spreadsheet Lotus 1-2-3. Because this application demanded both clear text display (for the spreadsheet itself) and graphics ability for its business charting, the HGC was an ideal graphics card to support its display needs. Due largely to the success of Lotus 1-2-3, the HGC quickly became a video adapter standard, but one that IBM refused to accept.⁴ Nevertheless, the IBM PC’s open architecture made it possible for consumers to use whichever adapter they preferred.⁵
The existence of CGA emulation drivers, which allowed Hercules users to run programs written for the CGA card’s monochrome graphics modes, may also have been a contributing factor to the success of the HGC, especially so because programming for the Hercules’ card’s native graphics mode was somewhat hindered by the fact that there was no standardization from IBM.⁶

The competition between IBM and Hercules intensified when IBM released the Enhanced Graphics Adapter (EGA) in 1984. The EGA was intended to add new video modes and to replace the MDPA, the CGA, and the HGC. Like its predecessors, it was not compatible with Hercules graphics. In 1986, Hercules responded with the HGC Plus, an updated version of its original card. The efforts of Hercules and IBM to capture the adapter market by this pattern of technological one-upmanship continued until Hercules was bought by Guillemot Corporation, a French-based graphics card producer, in 1998.⁷

Throughout their rivalry, IBM was not able to shut Hercules out of the adapter market using its considerable market power as a PC manufacturer and graphics card producer. Industry observers and video display experts readily argue that this inability was due directly to the additional functions of Hercules cards over IBM cards and indirectly to IBM’s open architecture, which made it much easier despite lack of programming compatibility for consumers to use either company’s graphics card with an IBM PC.⁸ The availability of emulator technology also facilitated the use of the HGC with IBM hardware.⁹ Another source of success, largely overshadowed by analyses that focus on IBM-Hercules, was the support Hercules received from Lotus 1-2-3, the leading spreadsheet application at the time the HGC was first developed. Namely, the HGC’s capabilities served as a better complement to Lotus than those of IBM’s adapters. Therefore, despite the fact that Lotus would likely run on an IBM PC, it was displayed on a monitor using an HGC, thereby expanding the demand for the HGC through the broad
consumer base for Lotus. As was noted above, the resulting competition between Hercules and IBM resulted in a healthy stream of innovation with neither company able to stifle the other’s incentives to improve existing technology.

In the case of Intel, Microsoft, and Sun Microsystems, the contrary applied. Early 1995, Intel attempted to establish itself in the nascent MM market by developing a new technology called Native Signal Processing (NSP). NSP enabled real-time MM tasks, such as audio signal processing or video file decompression, to be executed on the Intel Pentium processor. The NSP reference platform specification sets a baseline for a PC that can handle the demands of current and future MM and communications applications. Examples of such software technologies include 3DR, Indeo video, DCI, Native Audio and IA-SPOX, DMI, Power Management, and Instant On.

Intel had initially hoped to release NSP at the same time Microsoft released Windows 95. However, Microsoft refused, citing technical problems with NSP that could not be resolved in a timely manner and its fear that NSP could potentially serve as a system software platform in competition with some of Windows’ MM functions (such as Direct Sound). Intel decided not to pursue the issue further for the time being in order to preserve Microsoft’s support of Intel’s MMx technology, which allows for higher speed MM operations. After Windows 95 was released, Intel tried to get Original Equipment Manufacturers (OEMs) to sell NSP separate from Windows and started to develop more advanced versions of the MPEG specification.

By 1996, Microsoft, no longer distracted by the release of Windows 95, decided to actively pursue new MM technology. Its first innovation, Talisman, a graphics architecture, was doomed from the start because third parties believed Intel’s MMx2 and AGP (Accelerated Graphics Port, designed to reduce overall cost of creating high-end graphics subsystems by using existing system memory) were faster and offered better graphics. Makers of popular video games
also had similar cutting-edge technology. As such, Microsoft finally expressed a desire to work with Intel on developing MM, namely, defining media class libraries.\(^\text{18}\)

However, it was also around this time that Java emerged as both a credible Internet and MM technology. Intel immediately expressed interest in the potential benefits to its position in MM, as well as the chip market, of partnering with Sun Microsystems to further improve the MM capabilities of Java. Thus, in 1997, Intel began to collaborate with Sun on the development of the MPEG4 spec and Java MM Application Programming Interfaces (APIs).\(^\text{19}\) Microsoft, believing that this partnership was a threat to their own MM efforts and the attractiveness of the Windows platform, immediately responded with the following demands: (1) Intel must drop the MPEG4 spec (which targets the augmented/interactive video market) because it was a competing standard to Microsoft’s Dynamic HTML and DirectX foundation. (2) Intel must cease helping Sun create Java MM APIs, especially ones that run well on Windows. (3) Finally, Intel must work to augment DirectX, not compete with it.\(^\text{20}\)

Although Microsoft initially sought to compromise by offering to ship Intel MM components as part of Windows, the final outcome of the situation was not as conciliatory.\(^\text{21}\) AMD, a competitor to Intel, would eventually ask Microsoft to support its 3DX (for games). Microsoft agreed to not do so if Intel ceased to support Java API development for MM. In the end, Microsoft incorporated into Windows any MM interfaces that Intel agreed to not help Sun incorporate into the Java class libraries. Over time, Intel altogether stopped aiding Sun’s development of class libraries that offered cutting-edge MM support. Thus, not only had NSP failed to gain the position in the market that Intel wanted, but the development of MM technologies that Intel and Sun had hoped to develop and market together was aborted.\(^\text{22}\)

Unlike the Hercules case, the interactions between Intel, Microsoft, and Sun did not encourage innovation but stifle it. Note that in the former, the companies were concerned largely
with creating the best adapter, not with the effects their innovations would have on their relationships with each other. Unlike the formidable partnership between Intel and Microsoft, Hercules and IBM had no existing relationship that both parties wanted to preserve. IBM’s open architecture also contributed to this independence. Arguably, the relationships between Intel, Microsoft, and Sun before the MM ventures were an additional influence on the incentives of each company to react to and conduct certain strategies concerning each other—an influence that Hercules, IBM, and Lotus did not have to deal with.

First, Intel, Microsoft, and Sun all knew that the MM market would yield high future growth, but the method for tapping into this growth was different for each company. Intel, while part of the “Wintel” monopoly, was the less powerful member of the partnership because it had no distribution channel independent of Microsoft. While the success of Intel chips depended on the popularity of Windows, the opposite was not true, as was made clear recently when Microsoft switched to AMD as its primary chip supplier. Becoming a major player in a new market has substantial benefits on its own. Though Intel never expressly stated so, it is reasonable to believe that an added benefit to MM was the potential to gain leverage in its relationship with Microsoft by developing a technology that Windows would come to be dependent upon for success.

To be sure, both Intel and Microsoft agreed that NSP was riddled with technical problems when Intel first approached Microsoft. As well, while rushing to release Windows 95, fixing these problems would not be a high priority for Microsoft. Nevertheless, Microsoft had other reasons of a more strategic nature to block Intel’s MM efforts. Like Intel, Microsoft recognized the leverage that NSP would afford Intel in their relationship should it be incorporated into Windows. Additionally, Microsoft had MM plans of its own. Software such as Dynamic HTML and DirectX were the beginnings of what Microsoft hoped to be a full-fledged
proprietary MM platform. It saw Intel’s NSP as a competitor to this platform rather than as a complement to Windows. A few years later, as Sun sought to carve a lucrative niche for its Java platform through extensive MM technology, the same issue of a firm serving as both a complementor and competitor would arise.

The addition of Sun as a player in the development of MM introduced a new dimension to the existing competition: whereas the firms’ strategies mainly focused before on whose technology was better, they now also took into account whose partnership was more valuable. Since Microsoft had by far the most dominant platform of all three companies in terms of user base, it is critical that technologies be allowed onto the Windows platform in order to reach widespread distribution. At the same time, Microsoft cannot simply shut out all potentially competing technologies unless it can ensure that no innovation by any other company in any sector is preferred to its own innovations. In other words, Microsoft must make itself compatible to complementary technology. Despite the weight of the second consideration, Microsoft was able to shut out the MM innovations of Intel and break up its partnership with Sun even though industry observers and users believed the Windows MM functions were no better and arguably less cutting-edge than those offered by the other two companies.

The interactions between Intel, Microsoft, and Sun once Sun entered into the picture are entirely plausible given the dominance of the Windows platform, but they are nevertheless counterintuitive in that small entrants to markets (Sun was not yet in consumer software) should not be considered a threat to dominant firms, especially if the former are creating a complementary product. Additionally, Intel’s decision to partner with Sun instead of continuing to develop MM with Microsoft remains a puzzle. After all, even if Microsoft would insist on an MM technology based largely on its own vision, Intel would benefit from Windows’ huge distribution channel more than it would from Sun’s. The IBM/Hercules/Lotus case, on the other
hand, is the more typical case in that the complementors (Lotus and IBM as well as Lotus and Hercules) did not compete, but relied, on each other to increase their user bases. The control and use of distribution channels for technologies also follow easily from conventional wisdom: Hercules used the ubiquitous IBM PC as its distribution channel and neither Lotus nor Hercules was shut out by IBM because both companies’ innovations helped sell IBM PCs.

The model that follows seeks to solve these puzzles by explaining what tips the balance between treating another firm as a competitor and as a complementor. In doing so, the model will then be able to offer insights about how firms select and control their distribution channels. Finally, the model seeks to provide a framework by which to understand the effects of the strategic interactions between these three firms (Microsoft, Intel, and Sun and to a looser extent Hercules, IBM, and Lotus) on the incentive to innovate.

4. Theory

A, a developer of primarily hardware technology, seeks to create a new technology that it would like B, its partner and a software provider, to incorporate into its own widely-distributed software package. B has a virtual monopoly in the software industry, and its partnership with A has proven largely profitable for both. Unlike B, however, A lacks a software distribution channel of its own and must rely on A or another software provider to include its technology in their products. At the same time that A is developing its technology, B is considering creating something similar. As such, B can potentially find it profitable to feature its own technology in its software instead of A’s. Should it agree to include A’s software, B is then able to expropriate value from A by threatening to substitute B’s own software for A’s.

In response to this threat, A tries to increase its leverage with B by finding an alternative distributor, C, of its software. C is an entrant to the software market with little chance of directly
challenging the market position of B. However, to be the first to distribute the technology that A is developing can potentially deny B a lucrative position in a new market sector and retard B’s overall growth. While A certainly improves its bargaining position with B, it must also take into account the likely event that B will retaliate if A partners with C—either through more aggressive competition in the market or a severance of their existing partnership. The former action will also harm C, not in the sense that C will be significantly worse off if they partner with A, but rather that their partnership will prove unprofitable. Finally, A is aware that the presence of C improves its strategic position in this game, but future gains are uncertain because a partnership with either company can result in being held-up and expropriated. A, however, does not have a reciprocal ability because once it gives its technology to B or C, it essentially surrenders its only asset in the game. The following models examine the potential outcomes of this game and what strategic variables and game structures determine those outcomes.

4.1 Two Parallel Games

The simplest game structure involving these three companies is that of two parallel games—one between A and B and another between A and C. That is, A decides before the game begins which company it would like to partner with to distribute this innovation. Should this company reject A’s offer, A will not go to the other company, choosing instead to not innovate.

If A chooses to offer the technology to B, then B can either reject A’s offer in favor of developing and distributing its own software or accept A’s technology and agree to incorporate it into their own. Given the possibility that B may reject A’s offer, A also has the option of not developing the new technology at all. The picture below illustrates this game.
Similarly, A can seek a partnership with C. The structure of the game is exactly the same as that between A and B; however, the payoffs to the parties may be different, which will be discussed later in the paper. Graphically:

**Figure 1**

Though it is unlikely that A would not take advantage of the gains from courting more than one potential distributor, these games serve as instructive points of comparison to the three-player game. What these games attempt to model, after all, is how the presence of an entrant affects the behavior of the parties to an existing partnership as well as the fate of the new technology. Thus, it is necessary to determine the outcome of a game between only two companies in order to discern the difference a third company makes.

**Figure 2**
4.2 Three-Player Game

Once A has developed its new technology, it can approach either B or C first. Since A has an existing partnership with B and B is by far the most powerful company in the industry, A will seek a collaboration with B first. B can then choose to accept A’s technology or reject it, instead creating similar software on its own. If B rejects A, then A can either extend the same proposal to C or not go to C at all. Despite the virtual uselessness of its technology when not paired with an existing software platform, A still finds the latter option viable in the event that B threatens severe retaliatory action for forming a partnership with C. Once C receives an offer from A, it too can either accept or reject it to develop its own technology. This game can be summarized in the following tree:

![Figure 3](image)

In addition to capturing the sequential, multi-player nature of the relationship between these firms, this game structure also includes the feature that at any decision node that involves only the interaction between A and B or A and C, the firm not directly involved can successfully influence the decision made by applying its own strategic pressures. How C acts is a function of how B has already acted and threatens to act should C’s decision be unprofitable for B; the same applies to B’s decision.
As the game is currently set up, most of the payoffs can be assumed to be determined exogenously by the market. Should A decide the market will not be able to provide sufficient incentives for B or C to accept, it can offer a payment to either firm in return for accepting. A can use this transfer of money as an additional tool to persuade B or C to accept. A brief and more specific description of how this transfer affects the game follows the discussion of the game’s outcomes.

4.3 Payoffs

4.3.1 Game 1

In the game involving A and B, there are three possible outcomes: A innovates and B rejects, A innovates and B accepts, and A does not innovate. If A innovates and B rejects, the payoffs are as follows:

\[ A: -c \]  
\[ B: (T_M)(Q_M)\delta_M-c \]  
\[ = \text{cashflow}_B-\text{costs} \]

where \( c \) represents the cost of developing the technology, which is assumed to be the same for both parties; \( T_M \) is the expansionary effect on B’s number of users due to the introduction of their own technology; and \( \delta_M \) is the per-unit price of B’s software. \( Q_M \) denotes the number of B’s users and is determined by the demand function:

\[ \frac{a + fN_M - \delta_M}{b} \]

where \( a \) and \( b \) are constants and \( f \) is the benefit to the marginal user of having \( N_M \) current users of B’s technology.\(^1\)

---

\(^1\) In the payoffs for all games in this model, variables of the same name represent the same value unless otherwise noted.
The intuition behind these payoff functions is that if B rejects A’s offer, then it will develop its own technology instead of using A’s. Therefore, B’s payoff is given by the increase in its revenues due to its new software net of the costs required to develop this software. Since A has already developed its technology but receives no benefits, its payoff is only the cost of innovating.

If A innovates and B accepts, then the payoffs to each player are:

\[
A: (T_I)(I_M)(Q_I)\delta_I-c-V_M[P(I_M^* T_I^* Q_I > T_I^* Q_I)]
\]

(4)

\[
B: (T_I)(Q_M)\delta_M + V_M[P(I_M^* T_I^* Q_I > T_I^* Q_I)]
\]

(5)

Both of these equations reduce down to

\[
A: (T_I)(I_M)(Q_I)\delta_I-c-V_M[P(I_M>1)]
\]

(6)

\[
=\text{cashflow}_A^B-\text{costs}-\text{transfer}_{B^A}
\]

\[
B: (T_I)(Q_M)\delta_M + V_M[P(I_M>1)]
\]

(7)

\[
=\text{cashflow}_B^A + \text{transfer}_{B^A}
\]

where \(T_I\) is the expansionary effect on the current number of users due to A’s new technology; \(I_M\) is the augmenting of A’s network due to the network effects from B’s existing network; \(\delta_I\) is the per-unit price of A’s new software, and \(V_M\) is the value B can expropriate from A resulting from its ability to hold A up once it becomes its primary channel of distribution. \(Q_I\) represents the number of users on A’s existing network and is given by the demand function:

\[
\frac{a + gN_I - \delta_M}{b}
\]

(8)

where \(g\) is the benefit to the marginal user to having \(N_I\) current users of A’s technology. As an aside, A’s network overlaps with B’s network such that \(Q_I\) can be expressed as:
where $\alpha$ is the share of A’s network that overlaps with B’s network.

Thus, A’s payoff consists of its increased profit from partnering with B and distributing its new technology as well as the expected amount B can expropriate from A from holding A up. The expected value of this expropriation is derived from the actual value of expropriation multiplied by the probability that B can collect this value. This probability is determined by the extent to which A must continue to rely on B as a partner in the future, which—in this game—is proxied by how much the parameter M expands A’s network. As long as B’s network effects further expand A’s network, thereby making B’s cooperation necessary to sustain A’s expanded network, A will find it more profitable to maintain the partnership with B than to leave. The result is that B gains extra value through expropriating A.

Should A choose not to innovate, then both A and B remain in the same state as before the game took place. Thus, their payoffs are (0, 0).

### 4.3.2 Game 2

The game involving A and C has, fundamentally, the same structure as that between A and B. The three possible outcomes are: A innovates and C rejects, A innovates and C accepts, and A does not innovate. If A innovates and C rejects, the payoffs are as follows:

$$A: -c$$  \hspace{1cm} (1)

$$B: (T_S/Q_S)\delta_S - c$$  \hspace{1cm} (10)

$$= \text{cashflow}_C - \text{costs}$$

where $\delta_S$ is the per-unit price of C’s software, and $T_S$ is the expansionary effect on the number of users in C’s network. $Q_S$ is C’s current network and can be expressed in the demand function:
where \( h \) is the benefit to the marginal user of having \( N_s \) current users. Since much of C’s software is used on A’s software platform (indeed C’s own platform can be used on top of A’s), C’s network of users can also be expressed as:

\[
\frac{a + hN_s - \delta s}{b} \beta + Q_s' \tag{12}
\]

where \( \beta \) is the share of C’s network that overlaps with B’s network, and \( Q_s' \) is the number of users that accrue exclusively to A.

Should C choose to reject A’s offer, it will receive the benefits, net of development costs, of creating its own technology and expanding its network through a more desirable software package. At the same time, C forgoes the possibility of benefiting from the network effects of A’s technology and existing network. A, however, gains nothing from being rejected by C, so its payoffs consist only of its cost of innovating.

If A were to innovate and C were to accept, then the following payoffs apply:

\[
A: (T_I)(Q_I)(S_I)\delta_I - c - V_S[P(S_I \times S_S > T_I \times Q_I)]
\]

\[
= (T_I)(Q_I)(S_I)\delta_I - c - V_S[P(S_I > 1)]
\]

\[
= \text{cashflow}_C - \text{costs} - \text{transfer}_{CA}
\]

\[
C: (T_I)(Q_S)(I_S)\delta_S + V_S[P(S_I > 1)]
\]

\[
= \text{cashflow}_{CA} + \text{transfer}_{CA}
\]

where \( S_I \) is the augmenting of A’s network due to the network effects from C’s existing network; \( I_S \) is the analogue of \( S_I \) for C’s network with respect to A, and \( V_S \) is the value that C can expropriate from A due to its ability to hold the latter up in the future. The payoffs under this outcome are supported by nearly the same intuition that underlies the (A: innovate, B: accept)
strategy in Game 1. Both A and B receive the net benefits from incorporating and selling A’s technology, and money is transferred from A to B due to hold-up. Though it may seem odd that C, an entrant, can hold-up A, an incumbent, it must be noted that the bargaining power of both companies once C has included A’s technology in its software is determined by who has less to lose from refusing to cooperate with the other in the future. Because A has already revealed its technology to C, A has lost its primary leverage against C. Additionally, C is now A’s only channel of distribution; therefore, A cannot credibly threaten to hurt C through not cooperating. Conversely, C’s position as A’s distributor and its ability to stop fulfilling this role in the future and substitute its own software instead ensures A’s cooperation. It is thus profitable for C to extract value from A during their partnership.

Unlike B, C benefits from the network effects of A when it incorporates its technology. This is due to the existing partnership between A and B, namely, A’s main business is to provide hardware for B’s software. In this role, A’s network is already largely subsumed by B’s such that if B receives extra network benefits from partnering with A in a new sector, these benefits are nominal at most. A and C, however, have not had a previous direct relationship, giving both parties the ability to tap into the non-overlapping portions of each other’s networks. Thus, the parameter $I_5$ appears in C’s payoffs under the strategy (A: innovate, C: accept), but not under B’s.

If A chooses not to innovate, then both A and C remain unaffected by this game. Their payoffs are thus (0, 0) in this outcome.

4.3.3. Game 3

In its most basic state, the game that includes A, B, and C plays out as follows: A decides whether to innovate or not. If it chooses the former, it will first approach B with an offer; B can then accept or reject. If B rejects, A can either go to C with the same offer or allow the game to
end. If A approaches C, C can either accept or reject. Though each of the direct interactions in this game involves only two companies, the corresponding payoffs are influenced by all three companies’ strategies.

The best case scenario for A is (A: innovate, B: accept). In this outcome, the payoffs are:

\[ A: (T_I)(I_M)(Q_I)\delta_I - c - V_M[P(I_M \geq 1)] \]
\[ = \text{cashflow}_A^B - \text{costs} - \text{transfer}_B^A \]

\[ B: (T_I)(Q_M)\delta_M + V_M[P(I_M \geq 1)] \]
\[ = \text{cashflow}_B^A + \text{transfer}_B^A \]

C: 0

Should B accept, then A has no reason to go to C, and thus the three-player game reduces down to the two-player game between A and B. Therefore, the payoffs to A and B for this outcome are the same as they are in Game 1.

Another plausible outcome is that A innovates, B rejects, and A does not approach C with an offer. The corresponding payoffs are:

\[ A: - c \]
\[ = \text{cashflow}_A^B - \text{costs} \]

\[ B: (T_M)(Q_M)\delta_M - c \]
\[ = \text{cashflow}_B^A - \text{costs} \]

\[ C: (T_S)(Q_S)\delta_S - c \]
\[ = \text{cashflow}_C^A - \text{costs} \]

That both B and C block A implies that the two companies plan to develop their own technologies similar to the one that A has already created. Thus, B and C gain the benefits of an expanded network due to the new software, net of costs of innovating. A does not strike a deal with either company and so gains no benefits but incurs the costs of developing the technology.
Though it is entirely possible for B to pay A to not go to C, thus increasing A’s payoff, I will assume for this game that B’s market position alone is enough to incentivize A to no offer a deal to C.

Interestingly enough, A can actually lose more than the costs of innovating if both B and C reject. If A innovates, B rejects, A goes to C, and C rejects, then the payoffs are:

\[ A: -c - V_{PA}[P(Q_M < (Q_M^* M^0 + T_M * Q_M))] \]
\[ = - \text{costs} - \text{partnership}_{AB} \]

\[ B: (T_M)(Q_M)\delta_M - c \]
\[ = \text{cashflow}_B - \text{costs} \]

\[ C: (T_S)(Q_S)\delta_S - c \]
\[ = \text{cashflow}_C - \text{costs} \]

where \( V_{PA} \) is the value of the current partnership between A and B from A’s perspective and \( M^0 \) is the share of B’s network attributable only to B’s technology. Since neither B nor C accept A’s offer, their payoffs only involve the profitability of their own software. By going to C, however, A draws the ire of B because B views C, buoyed by the new technology and network benefits of A, as a potential competitive threat, and would thus prefer that A not help the market position of C. B, because of its own market power and the plentiful supply of hardware providers, can sever the partnership with A at low cost. A, however, does not have the luxury of being able to build a partnership with a software supplier nearly as dominant as B; A will therefore almost certainly lose substantial profit from losing B as a partner. In short, B can credibly threaten to end its partnership with A should A engage in actions that decrease B’s profits—which in this case means trying to strike a bargain with C.

The expected value of this lost partnership to A is its actual value multiplied by the probability that B will cut its ties. The latter value can be proxied by the probability that the
current network without the benefits of A’s technology combined with the expansion of the network due to B’s own version of A’s new software is greater than B’s current network, which includes the effect of having A as a partner. Simply put, if B’s network does not suffer from ending its partnership with A, it will do so if A does not cooperate with B’s wishes.

The final outcome possible for this game is that A innovates, B rejects, A goes to C, and C accepts. The payoffs in this case are:

\[
A: \left[ P(\Theta = 1) \right] \cdot \left( T_I - \gamma \right) \cdot \left( Q_M \cdot \delta_M - c \right) - \left( V_M \cdot P(Q_M > k Q_I) - P(Q_M > k^* T_I^*) \right) - S_I^* Q_I^* - V_{PA} \left[ P(Q_M < (Q_M^* M^0 + T_M^* Q_M^*)) \right] \\
\]

\text{expropriation}_{B^A} - partnership_{AB}

\text{where } V_{MC} \text{ is the expected value of what B can expropriate from A due to their current partnership and } k \text{ represents the factor by which having B as a partner expands A’s network. } \gamma \text{ is the reduction in the growth of B’s network due to the introduction of software by A and C in a lucrative new sector, and } \Theta \text{ is a binary variable whose value is one if B will agree to make its software platform compatible with A and C’s new product and zero if B decides to remain incompatible.}

A partnership with C secured, A gains the net benefits to its network of its new technology and the spillover effects of C’s network. A also gains greater leverage against B because its profits, supported by the venture with C, are no longer as dependant upon its partnership with A. The decreased dependence is equal to the expected value of B’s
expropriation, which is in turn determined by the actual value of the expropriation weighted by the probability such activity will continue to occur. Before A’s partnership with C, this probability was proxied by B’s market power relative to A’s. Hence, I express this value in the payoffs as the probability that B’s network exceeds A’s by a constant that serves as the threshold at which the size of B’s network renders it much more powerful than A. Because of A’s increased network with its new technology, the probability that B can successfully expropriate has decreased somewhat, leaving B with a lower expected value of its expropriation of A. The difference between this value and the one that preceded A’s partnering with C shows up in A’s payoffs as a benefit from its new partnership.

Following the intuition of payoffs in previous outcomes, A must also incur the cost of being held up by C and of potentially losing its partnership with B.

B finds itself in an awkward position because it has a variable in its payoffs that is on some level controlled by the market but is more immediately determined by C. The term $\gamma$ captures the likely event that A and C’s succeeding in being the first to establish themselves in a new market sector reduces B’s chances of having equal success in the sector. Thus, the partnership between A and C does not diminish B’s current profitability. Indeed, it works to expand B’s network by introducing a software product that is a complement to B’s software platform, thereby making B’s platform more attractive to users. But in order to protect its dominance, B must continue to grow. Competing software firms that limit the number of ways in which B can grow pose a threat to B’s future. The extent to which A and C’s new software is a threat is thus indispensable to B’s payoff function and is represented by $\gamma$. The ways in which C can influence the value of $\gamma$ include the encouragement of other entrants to establish themselves in other sectors before B does and erecting barriers to the sector that A and C have opened up.
The success of these methods, while important, is exogenous to this game and will not be discussed in this paper.

It must be of comfort to B, however, that C’s situation in this game is relatively more precarious. Though C has a software platform of its own, it is much less profitable to have a new technology run on only its own platform instead of on B’s as well. In fact, it is crucial that A and C be able to run their program on B’s platform. If $\Theta=0$, then neither A nor C will find it profitable to innovate. Thus, any expansion that either A or C receives from their new software must be weighted by the probability that B will actually allow this software to run on their platform.

As in other outcomes in which C partners with B, C benefits from its ability to expropriate A.

4.4 Outcomes

4.4.1 Game 1

If

\[
\text{cashflow}_B - \text{costs} < \text{cashflow}_B^A + \text{transfer}_B^A, \tag{19}
\]

then B will accept A’s offer. Rearranging the equation, we get

\[
\text{cashflow}_B - \text{cashflow}_B^A - \text{costs} < \text{transfer}_B^A. \tag{20}
\]

Thus, if the advantage of innovating on its own is less than the value that can be extracted from A through hold-up, B will accept. If not, B will reject.

If B accepts, A should innovate if

\[
\text{cashflow}_A^B - \text{costs} - \text{transfer}_B^A > 0. \tag{21}
\]

Rearranging,

\[
\text{cashflow}_A^B > \text{costs} + \text{transfer}_B^A \tag{22}
\]
If the benefits to network expansion are greater than the costs of innovation and the expected value of expropriation by B, then A should innovate. If not, A should not innovate. Additionally, if B decides to reject, then A should not innovate to avoid incurring costs without receiving any benefits (—costs<0).

4.4.2 Game 2

C will accept A’s offer if

\[ \text{cashflow}_C - \text{cashflow}_A - \text{costs} < \text{transfer}_A \]  

(23)

Similar to Game 1, C will accept if the advantage to innovating on its own is less than the value it can expropriate from A. If C accepts, A should innovate if

\[ \text{cashflow}_A^C > \text{costs} + \text{transfer}_A \]  

(24)

Again, if the network expansion from innovating exceeds the costs as well as the value C can expropriate, A will innovate. If C rejects, then A should not innovate so as to avoid negative returns (—costs<0).

4.4.3 Game 3

There are four periods to this game corresponding to the following decision nodes:

- Period 4: C decides whether to accept A’s offer or not.
- Period 3: A decides whether to seek a partnership with C or not.
- Period 2: B decides whether to accept A’s offer or not.
- Period 1: A decides whether to innovate or not.

**Period 4**

C rejects if

\[ \text{cashflow}_C - \text{costs} > (\text{probability of compatibility}) \times \text{cashflow}_A + \text{transfer}_A \]  

(25)
Rearranging,

\[ \text{cashflow}_C - (\text{probability of compatibility}) \times \text{cashflow}_C^A - \text{costs} > \text{transfer}_C^A \]  \hspace{1cm} (26)

It is interesting to note that in this inequality, the higher the probability that B will make their platform compatible with A and C’s software, the less likely C will reject.

**Period 3**

If C rejects in period 4, then A will not approach C if

\[ \text{costs} > \text{costs} + \text{partnership}_{AB} \]  \hspace{1cm} (27)

Assuming that \( \text{partnership}_{AB} \) is greater than zero, A will not approach C if C will reject.

If C will accept, then A will approach C if

\[ \text{cashflow}_C^A - \text{costs} - \text{transfer}_C^A + \text{current expropriation}_B^A - \text{partnership}_{AB} \geq \text{costs} \]  \hspace{1cm} (28)

Canceling the costs from both sides of the inequality, it becomes clear that A will go to C only if C’s potential expropriation and the change in expected value of the partnership with B are less than the expanded network due to the new technology and the decrease in the expected value of B’s expropriation.

**Period 2**

If A will go to C and C will accept, then B will accept if

\[ \text{cashflow}_B^A + \text{transfer}_B^A > (\text{growth factor}) \times \text{cashflow}_B^A - \text{current expropriation}_B^A \]  \hspace{1cm} (29)

Rearranging,

\[ \text{cashflow}_B^A - (\text{growth factor}) \times \text{cashflow}_B^A > -\text{transfer}_B^A - \text{current expropriation}_B^A \]  \hspace{1cm} (30)

Thus, B will accept if the gain it receives from being able to further expropriate A is less than the retardation of the growth of its network due to the new technology.
If A will go to C and C will reject, then B will reject if

\[ \text{cashflow}_B - \text{costs} > \text{cashflow}_{B}^A + \text{transfer}_{B}^A \]  

(31)

Rearranging,

\[ \text{cashflow}_B - \text{cashflow}_{B}^A - \text{costs} > \text{transfer}_{B}^A \]  

(32)

The difference in the increase in B’s network due to the introduction of its own technology and due to A’s innovation net of the costs of innovating must be greater than the value B can expropriate from A in order to make rejecting optimal.

If A does not approach C, then B will reject if

\[ \text{cashflow}_B - \text{costs} > \text{cashflow}_{B}^A + \text{transfer}_{B}^A \]  

(31)

The outcome in this last case is the same as that in which C rejects A’s offer.

**Period 1**

If B will accept A’s offer, then A will innovate if

\[ \text{cashflow}_{B}^A - \text{costs} - \text{transfer}_{B}^A > 0 \]  

(32)

Rearranging,

\[ \text{cashflow}_{B}^A - \text{costs} > \text{transfer}_{B}^A \]  

(33)

If the net gain to innovating is greater than the costs of being expropriated, A will innovate.

If B will block due to the event that A has gone to C and C has blocked, then A will innovate if

\[ \text{cashflow}_{A}^C - \text{costs} - \text{transfer}_{C}^A + \text{current expropriation}_{B}^A - \text{partnership}_{AB} > 0 \]  

(34)

Rearranging,

\[ \text{cashflow}_{A}^C + \text{current expropriation}_{B}^A > \text{costs} + \text{transfer}_{C}^A + \text{partnership}_{AB} \]  

(35)
If the expansion of A’s network and the decreased expected expropriate from B are greater than the cost of innovating, the value of the partnership with B, and the expected expropriation by C, then A will innovate.

If both B and C will block, A should not innovate if

\[-costs - partnership_{AB} < 0\]  

(36)

Assuming both terms on the left of the inequality are positive, A should not innovate.

4.5 Bargaining

Depending on the bargaining power of A with regard to B and C, A can more effectively persuade both companies towards outcomes that favor A by offering a bargaining transfer of money, $t$, in return for accepting A’s offer. The specific amount of money that needs to be transferred depends on the other variables in the payoffs and the bargaining power of the parties involved that in turn can be used to determine a bargaining solution. Since $t$ can theoretically take on a near infinite amount of values, which lead to a correspondingly large number of outcomes, not all effects of adding $t$ to the payoffs will be explored in this paper. Instead, I will use an example to show how offering $t$ can help A achieve its desired outcomes.

Say B will not make its platform compatible with A and C’s software ($P(\Theta = 1) = 0$). Then C will accept A if

\[\text{cashflow}_C - costs - \text{transfer}_{CA} < t\]  

(37)

Rearranging,

\[\text{cashflow}_C - costs - \text{transfer}_{CA} < t\]  

(38)

Compared to the case without the transfer,

\[\text{cashflow}_C - costs - \text{transfer}_{CA} < 0\]  

(39)
C has a better chance of accepting even though B will block its software. Whereas without the transfer the value C can expropriate from A must be greater than the net benefits of the new technology, A can make up the difference between the two through $t$ if the former happens to not exceed the latter.

If C is less likely to reject A, then so is B. Let’s say B accepts A as a result of this. A thus gains:

With the transfer: $\text{cashflow}_A^B - \text{costs} - \text{transfer}_B^A - t \quad (40)$

Without the transfer: $-\text{costs} \quad (1)$

As long as $t$ does not exceed A’s gain from partnering with B, it is profitable for A to offer a transfer.

5. Model Results

The central question this model is trying to answer is: Does participating in Game 3 make A better off or worse off than simply playing Game 1 or Game 2? In the case that B accepts, A would be just as well off going to B only or approaching both B and C. It is important to note, however, that even though the payoffs will be the same, B will be more likely to accept in Game 3 because B’s payoff in this game depends on a variable—growth factor—whose value is controlled by C. Of course, even if C were to be given the option to exercise this control, it would not want to drive it too low because C and B are still complementors and B is the most expansive platform on which C’s technology can operate. However, C will also want to keep the growth factor low to benefit its own platform. Thus, if A is to benefit from C’s presence in Game 3, it will do so not through its own efforts, but the strategic interaction between B and C.

A is worse off going to Game 3 if:

\[ \text{cashflow}_B - \text{cashflow}_A^B - \text{costs} > \text{transfer}_B^A \quad (32) \]
AND either one of the following two conditions holds:

1. $\text{cashflow}_C - \text{costs} - (\text{probability of compatibility}) \times \text{cashflow}_C^A > \text{transfer}_C^A$ \hspace{1cm} (26)

   or $\text{cashflow}_C - \text{cashflow}_C^A - \text{costs} > \text{transfer}_C^A$ \hspace{1cm} (41)

2. $\text{cashflow}_B - \text{cashflow}_B^A - \text{costs} < \text{transfer}_B^A$ \hspace{1cm} (20)

   and $\text{cashflow}_B^A - (\text{growth factor}) \times \text{cashflow}_B < -\text{transfer}_B^A - \text{current expropriation}_B^A$ \hspace{1cm} (42)

A is indifferent between going to Game 3 and simply playing Game 1 or 2 if:

$\text{cashflow}_B - \text{cashflow}_B^A - \text{costs} < \text{transfer}_B^A$ \hspace{1cm} (43)

OR the following two conditions hold:

1. $\text{cashflow}_C - (\text{probability of compatibility}) \times \text{cashflow}_C^A - \text{costs} < \text{transfer}_C^A$ \hspace{1cm} (44)

   and $\text{costs} > \text{costs} + \text{partnership}_{AB}$ \hspace{1cm} (27)

2. $\text{cashflow}_B^A + \text{transfer}_B^A > (\text{growth factor}) \times \text{cashflow}_B - \text{current expropriation}_B^A$ \hspace{1cm} (29)

A is better off going to Game 3 if:

$\text{cashflow}_B - \text{cashflow}_B^A - \text{costs} > \text{transfer}_B^A$ \hspace{1cm} (32)

AND both of the following conditions hold:

1. $\text{cashflow}_C - (\text{probability of compatibility}) \times \text{cashflow}_C^A - \text{costs} < \text{transfer}_C^A$ \hspace{1cm} (44)

2. $(\text{probability of compatibility}) \times \text{cashflow}_A^C - \text{transfer}_C^A + \text{current expropriation}_B^A - \text{partnership}_{AB} > 0$ \hspace{1cm} (45)

Whether A is better off going to Game 3 is determined by whether B changes its strategy such that A receives a higher payoff than simply playing Games 1 or 2 due to the presence of C. Of course, if both B and C accept in Game 3, A has 2 distribution channels, but B’s strategy is
necessary and sufficient to determine whether A is better off. With this in mind, four possible scenarios exist for B’s strategies in Games 1 and 3:

- Accept in both games
- Accept in Game 1, but not Game 3
- Reject in Game 1, but not Game 3
- Reject in both games

A is worse off in the second scenario, better off in the third, and the same in the first and fourth scenarios. Given any parameters, one of these four situations is bound to come about, and so a unique set of the conditions listed above will be satisfied. It is important to note that these conditions are not simply restatements of the equilibrium of the game; they are determined by their underlying economic parameters, which by and large correspond to, but do not depend on the outcomes of the game.

In all of the conditions above, A’s welfare is determined by the values of the cash flows, transfers, and the probability of compatibility ($\Theta$) and growth factor ($\gamma$). The first two are largely determined by the market and by the firms’ abilities to bargain. Both of these determinants are exogenous to the game and the cash flows are not in the immediate control of the firms.

Bargaining power was touched upon earlier in the paper, but no further exploration will be done here as to how it would affect the values of the transfers. $\Theta$ and $\gamma$, however, are determined directly by B and C and are the operative tools that these firms can use to leverage the situation in their favor. The values $\Theta$ and $\gamma$ will take are endogenous to the game—to the extent that their values depend on the strategy of the firms controlling them. Thus, with respect to the confines of this game, what determines whether A will fare better in Game 3 as opposed to Games 1 and 2 is the interaction between B and C, namely how the two choose to balance their desire to further
their own platforms and the need to acknowledge their relationship as complementors. An examination of this phenomenon follows.

B will accept A’s offer if:

\[ \gamma > \frac{V_M[.] - V_{MC}[.]}{\delta u Q_M} \]  

(46)

That is, if the retardation of B’s growth from a successful partnership between A and C exceeds the ratio of the difference in B’s ability to expropriate A and the present cash value of B’s network, B will partner with A. Holding \( \gamma \) fixed, the higher B’s increased expropriation of A from the new technology is, the more likely the inequality above is going to hold true. Also, the higher the cash value of B’s current network, the less likely a partnership will be formed. The harder A and C work to set a high value of \( \gamma \), the more likely B will accept.

C will reject A’s offer if

\[ \frac{T_s}{T_i} - \frac{\cos ts + V_s[.]}{(T_i)(Q_s)\delta s} > P(\Theta = 1) \]  

(47)

The more C’s own technology is likely to expand its network relative to A’s technology, the more likely C will reject A. C will also likely reject if B sets \( P(\Theta=1) \) at a very low value.

It is important to note, however, that while \( \gamma \) and \( \Theta \) cannot be influenced by B and C, respectively, they cannot be entirely determined on an exogenous basis either. Both of these variables are dependant on the leveraging power of the companies that can influence them, and this power is proxied largely by network externalities. In the game, C’s network is comprised of a proportion of B’s network and a group of users that use C’s technology exclusively. It is this residual group of users and the possibility of their subscribing to A and C’s technology before B’s that gives rise to \( \gamma \). Though C can control the size of this group and their overall network
more effectively than any other company, there exist other determinants exogenous to the game that are wholly based on the market and the market structure.

On the other hand, \( \gamma \) is also dependant on a strategic interaction that is not captured by Game 3 but involves many variables that are endogenous to the game. Namely, the relative value of \( \gamma \) to \( V_M \) and \( V_{MC} \) gives B significant control over whether A and C can successfully set \( \gamma \) high enough to benefit their goals. At the same time, it is in C’s interest to increase \( \gamma \) because \( \gamma \) in and of itself is a proxy for its own growth relative to that of B, but it is also in C’s interest to not increase \( \gamma \) because doing so would increase B’s incentive to accept, shutting C out of A’s technology.

The same analysis can be applied to \( \Theta \). B would like to decrease \( \Theta \) to dissuade A from going to C. However, A and C’s technology makes B’s software platform more attractive, and so increasing \( \Theta \) also has positive effects. Additionally, the ability of \( \Theta \) to give B the outcome it wants is dependant on \( V_S \), which is wholly controlled by C, and other variables, such as \( T_i \), that are determined exogenously by the market. How A, B, and C will seek to set \( \gamma \) and \( \Theta \) is perhaps best captured by another game structure that involves exclusively the interaction between these two variables and the tools that other companies can use to determine their success.

6. Conclusion

Perhaps the most important point that Game 3 illustrates is that complementors can compete with each other for two reasons: 1) while the development of a new product by one firm complements and therefore makes more attractive the product of the other firm, the new product also deprives the non-innovating firm of the growth and revenues that come with being the first to introduce a new technology. Although the non-innovating firm can utilize resources to create
their own technology to compete, there are sizeable cost and network advantages to being the first mover. 2) If the innovating firm works with a partner of the non-innovating firm, which was the case in Game 3, an extra layer of tension is added to the latter partnership (in our notation, firms A and B). For the same reason that the innovating firm can be a competitor despite its direct role as a complementor, any firm that attaches itself to the complementary technology finds itself in the same situation. In A’s case, network expansion and impingement of the future potential growth of B are only one set of effects from its partnership with C. A also gains more leverage in its existing partnership with B because it is now not as reliant on B’s partnership for firm profits and a steady distribution channel. In sum, a complementor serves as a competitor by changing the nature of the partnerships between the firm whose product it complements and other firms, and by reducing the complementary firm’s potential for future technological development and growth.

To fairly assess the effects of C’s presence in Game 3, it is helpful to consider what actually happened when this game was played between Intel, MS, and Sun. First, Intel played Game 1 and MS rejected, but Intel innovated nevertheless. When C appeared as a potential distribution channel for Intel’s technology, A played Game 3. B then threatened to reduce the value of partnership$_{AB}$ such that A withdrew its offer to partner with C. In Game 3, the condition that had to be satisfied in order for A to approach C if C were to accept is given by

$$\text{cashflow}_C^A - \text{costs} - \text{transfer}_C^A + \text{current expropriation}_B^A - \text{partnership}_{AB} > -\text{costs}$$

(28)

The higher partnership$_{AB}$ is, the less likely this inequality will hold true. Thus, C can apply pressure in this game, but A will not find itself in a better position with B if the value of their partnership is high.

The presence of a third party is not enough to pressure the dominant member of a partnership to allow its partner to profit from innovating on its own. As long as the value of the
partnership is high enough, innovation will be stifled on some level. A natural extension of this project would be a study on the efficiency implications of this situation. At the surface, the strength of a partnership seems to diminish welfare. However, it may well be the case that the ability of a powerful partner to innovate at lower private and social costs results in greater efficiency gains than those which arise from several firms competing to be the first and dominant player in a given technology.

It is interesting to note the similarity between this problem and that of the relationship between market structure and innovation. Both stem from the question of whether a more concentrated or competitive market encourages greater innovation. Traditionally, attempts to answer this question have centered on the cost structures and competitive incentives of different sizes of firms. The same approach can be applied to the computer industry by assigning different relative values to the variable $c$ for each size of firm, examining the market risks each firm faces when innovating, calculating projected profits from the new technology, and comparing the resultant incentives to innovate. However, as this paper has shown, the strategic tensions between firms are also important to consider. For B, the cost of innovation matters much less than the potential loss of growth if it does not innovate. For A and C, the model predicts that neither should seek a partnership on a new technology even if it is competitively wise to do so. Though history offers a somewhat different picture, in equilibrium, no independent innovation will be undertaken by the partner of a dominant firm unless the latter allows it to occur. Again, the efficiency implications of this result have yet to be determined. It is clear, however, that any policy that seeks to increase the rate of innovation in the computer industry must take into account the potential of coopetition to break down and the decreased development of platform components by firms other than the one that controls the platform.

7. Endnotes


6 “Video Display Standards.” http://www.cknow.com/refs/Video Display


9 “Video Display Standards.” http://www.cknow.com/refs/Video Display


8. Reference List


