

MANIPULATING INTERFACE STANDARDS AS AN ANTI-COMPETITIVE STRATEGY

Jeffrey K. MacKie Mason^{*}

and

Janet S. Netz^{**}

August 2002

Abstract: The creation of interface standards enables competition at the level of components, rather than competition in complete systems. Consumers often benefit from component competition. However, the standard-setting process might be manipulated to achieve anticompetitive ends. We consider the conditions under which a standards consortium could impose anticompetitive burdens on the market, and several strategies such a consortium might employ to achieve anti-competitive objectives. We present a new strategy – one-way standards – and discuss the conditions under which it can be anticompetitive.

^{*} Arthur W. Burks Professor of Information and Computer Science, Professor of Economics and Public Policy, University of Michigan, Ann Arbor, MI 48109-1092, jmm@umich.edu.

^{**} ApplEcon LLC, Ann Arbor, MI 48104, netz@applecon.com.

MANIPULATING INTERFACE STANDARDS AS AN ANTI-COMPETITIVE STRATEGY

Jeffrey K. MacKie Mason

and

Janet S. Netz

August 2002

I. Introduction

For complementary devices in a complex technological system to interoperate successfully, they must communicate through interfaces. In communications and computing systems, interfaces are connections through which signals pass. The devices on both sides of an interface (*e.g.*, the microprocessors and a disk drive, or the PBX and the central office switch) must be designed so that they make the correct physical connection, send correct signals to each other, and correctly interpret signals received. We refer to the formal physical and signaling details as the *interface specification*.

An interface stands (physically or logically) between two (or more) separate components. Thus, for an interface specification to succeed, it must be adopted by at least one manufacturer of each of the components. When an interface specification is adopted and implemented by at least one different firm manufacturing each of the affected components, we refer to it as an *interface standard*.

Most specifications in telecommunications and computing are developed by firms participating in the industry. There are several different configurations of industry participants that might work together to create a specification. For example,

- A group composed of several manufacturers of each component. In some such cases, a relatively open process is used, in which a membership organization (with or without government sanctioning) accepts any qualified participant who manufacture either (or both) of the complementary

Manipulating Interface Standards

components, and through a formal process the organization jointly develops the specification.

- A group of firms that manufactures just the component on one side of the interface. For example, auto manufacturers might agree on a specification for attaching tires to wheels without the participation of tire manufacturers.
- In other situations, a single firm proposes a specification. For example, once required to do so by the FCC, AT&T announced the specifications for attaching customer premises equipment to its network.¹ Microsoft unilaterally announces the specifications of applications programming interfaces (APIs) for software programs to communicate with its operating systems.

Competition and consumers often – but not always – benefit when interface specifications are standardized and openly published benefit. Competing firms are enabled to design and manufacture system components that will correctly interoperate, and thus consumers (or systems integrators who then sell to consumers) can mix and match components from different manufacturers to get the set of components that offers the best combination of price and performance. However, there may be more incentive for innovation, or the adoption and rejection of new technologies may be more efficient when competition is for complete, incompatible systems.

It is conventional to assume that standards lower the barriers to entry in a market because potential entrants can design components that interoperating with existing complements if they adhere to the standard. However, the standard-setting process can be manipulated to *create or raise* barriers to entry. Just as with a price-setting consortium (a cartel), a standards consortium may be able to harm competition when its membership characteristics satisfy conditions for market power and barriers to entry.² Sufficient

¹ Rules requiring AT&T to permit others to attach customer premises equipment (CPE) to its network, and to publish the interface specifications necessary to do so, were developed by the courts and the FCC in a series of landmark decisions: the Hush-a-Phone and Carterfone cases (Hush-A-Phone Corp. v. United States, 238 F.2d 266, 268 (D.C. Cir. 1956), on remand, 22 F.C.C. 112 (1957); and, Use of the Carterfone Device in Message Toll Telephone Service, 13 F.C.C.2d 420, on reconsideration, 14 F.C.C.2d 571 (1968)), followed by the Part 68 Rules (47 C.F.R. § 68.1-.506) and the Second Computer Inquiry decisions (Amendment of Section 64.702 of the Commission's Rules and Regulations (Second Computer Inquiry), 77 F.C.C.2d 384, on reconsideration, 84 F.C.C.2d 50 (1980), on further reconsideration, 88 F.C.C.2d 512 (1981), *aff'd sub nom.* Computer & Communications Industry Ass'n v. FCC, 693 F.2d 198 (D.C.Cir. 1982), cert. denied, 461 U.S. 938 (1983), on second further reconsideration, FCC 84-190 (released May 4, 1984)).

² The European Commission (1987) recognized these characteristics in *X/Open Group*.

Manipulating Interface Standards

conditions to anticompetitively manipulate a standards process are (1) the consortium includes firms with sufficient market power to ensure industry adoption of the standard, and (2) membership and decision-making control are restricted in a manner that excludes viable potential competitors.³

When a standards consortium has the potential to exercise market power, various strategies may have anti-competitive consequences. These strategies include delaying publication of the standard to gain a first-mover advantage; creating standards that require other firms to use royalty-bearing intellectual property (e.g., a patent owned by a firm in the standards consortium); and creating one-way standards.

To the best of our knowledge, the last strategy – one-way standards – has not previously been described in the economics literature. In an industry with complementary system components that interoperate, component manufacturers on *both sides* of the interface require specifications for the physical and/or logical connections that enable the components to interoperate. In general it is necessary to publish the specification of both sides of the interface protocol for manufacturers on either side to use the standard. However, through creating a blind or cut-out – an extra technology layer -- a consortium can publish the information necessary to manufacture compliant components on one side of the interface without releasing information necessary to manufacture components on the other side. We call these “one-way interface standards”. They facilitate competition for one component, but harm competition for the other, complementary component.

Whether one-way interface standards harm consumers overall turns on the same issues well-known in the tradeoff between mix-and-match and systems competition. Our contribution is to show how an interface standards consortium can move the boundary that separates systems from mix-and-match competition.

In the penultimate section of this paper we present three detailed examples of standards consortia that apparently have employed these tactics to use standard-setting processes for anticompetitive gain. One is Intel’s specification of the AGP advanced graphics standard; another is the JEDEC consortium’s creation of a standard subject to

³ Many standards-setting groups have two levels or groups of membership. One group controls (sets) the standard, and the other group has an advisory and/or testing role. For example, the USB 2.0 Implementers Forum has Promoter Members, who are allowed to vote on decisions, and Participant Members, who are allowed to participate in the discussions but is not allowed to vote. See the group’s by-laws, www.usb.org/data/retail/usbif_bylaws.pdf, accessed 1 September 2002.

Manipulating Interface Standards

the patents of Rambus, which participated in JEDEC. The third is the development of the USB 2.0 and EHCI interface specifications to implement high-speed serial communications with desktop computer peripherals.

II. Competition and Interface Standards

When interface specifications are standardized and non-proprietary, component competition – that is, competition between multiple manufacturers of a given component in a system – can thrive. However, it is not given that component competition is necessarily superior to systems competition. We briefly describe the benefits and costs of component competition.

A. Benefits from component competition

There are a number of well-known benefits to consumers and society from component competition. We discuss several in this section. As an example keep in mind the markets for peripherals and components specific to Apple Macintosh hardware and for those compatible with Windows-capable PCs. Several of the interfaces in the Macintosh system are proprietary, and for those there are fewer firms, less product variety, and higher component prices than for PCs. For example, there have been tremendous advances in low-cost, high-performance audio and video hardware for PCs, largely driven by small new entrants who have found it worthwhile to compete because the interfaces were standardized.⁴

1. Competition on price and performance

Firms that are unable or unwilling to compete by designing and manufacturing entire systems of multiple components can enter the market for one (or more) of the components, and the greater entry results in more competition on price, performance, and quality of the component in question (Matutes and Regibeau 1988; Economides 1988).

Competition on price and performance is perhaps the most important pro-competitive benefit of component competition when interfaces are standardized and

⁴ PC Connection (a leading component retailer) lists 87 add-in video cards for Intel-based PCs. Mac Connection (owned by the same company) lists only 5 add-in video cards for Apple Macintosh computers. In addition, prices for the PC components are lower. For example, the ATI Tech Radeon 32MB DDR board for a PC is \$166 with an AGP interface (http://www.pcconnection.com/scripts/productdetail.asp?product_id=214468, accessed 28 July 2001). The same card for the Macintosh is \$209-\$240 with an AGP interface (http://www.macconnection.com/scripts/productdetail.asp?product_id=219741, accessed 28 July 2001).

Manipulating Interface Standards

open. Without standardization, competition is between incompatible systems, rather than between mix-and-match components. As a consequence, there is increased product differentiation among components of a particular type: they are compatible with different systems. If that differentiation does not serve substantially different consumer tastes – that is, if there is not much demand for that type of variety – but rather serves merely to divide the market, then this differentiation is costly to consumers. Firms with differentiated products are able to charge higher prices because they do not have as much competition from reasonably interchangeable substitutes. Thus, spurious differentiation can lead to higher prices and may not provide offsetting gain from variety (Farrell and Saloner 1986).

2. Scale efficiencies and lower production costs

By permitting firms to sell components to multiple buyers, standardized interfaces enable firms to realize efficient scale and learning economies (Hemenway 1975). For example, if interfaces between terminal equipment and key systems or PBX switches were standardized, phone system integrators could reduce their inventory costs because they would not have to stock as many different designs of handsets. The example given above of price differences for Macintosh and PC video cards might be explained by production scale efficiencies.⁵

OEMs (original equipment manufacturers) and other system assemblers can reduce their inventory costs because they do not have to stock multiple different designs of components (*e.g.*, motherboards, serial devices, printers, etc.) in order to put together systems that use different microprocessors.

3. Network externalities

For many products – of which desktop computers are one of the best known examples – consumers benefit the more other users there are of the same (or a compatible) product. More software will be available, more design errors will be corrected, learning curves will be shorter and the payoff to investing in learning will be greater (*e.g.*, in job mobility).

Several standards for mobile phones are in use. Phone companies in the U.S. largely adopted TDMA multiplexing; Europe and most of the rest of the world adopted

⁵ See footnote 4, above.

Manipulating Interface Standards

CDMA (specifically, the GSM flavor). Consumers with GSM phones benefit from being able to use their phones as they move from country to country.⁶ Some U.S. users are starting to benefit from this network externality, as AT&T deploys its new GSM network. However, to do so customers must purchase a more expensive “dual mode” phone to be able to make calls within the U.S. when roaming outside the rather limited footprint of AT&T’s GSM network.⁷

4. More innovation and variety for components

Similar to the consumer benefits from network externalities are the incentives for component manufacturers to invest in innovation and new product design (variety) when interfaces are standardized. When interfaces between complementary components are open and standardized, a firm making one component in a system faces a much larger potential market than in a market with closed, proprietary interfaces. If it develops a valuable innovation or new product design, the firm potentially can sell to all customers of the standardized, complementary products. If interfaces are proprietary, a component firm that innovates can only sell its component to the limited number of customers for the particular system with which it works. Thus, the incentives to innovate may be higher when the complements to the innovation are standardized. When the potential payoffs are much larger it is worthwhile for small, innovative new firms to incur the risks and costs of entry, enhancing competition.

5. Reduced risk of stranded investments

When interfaces are standardized, consumers will have confidence that they can buy upgraded components that will work with their systems, and that these components will continue to work if they purchase a new base system. For example, consumers can add larger and faster hard drives, improved monitors, scanners and other devices to their computers (Porter 1985).

B. Costs from component competition

There are also some potential costs to consumers from component competition

⁶ One of the authors observed Martin Cave, while in Australia, use his UK phone to call someone with an Australian phone who was sitting in a cubicle 10 feet away.

⁷ Many U.S. consumers have also purchased dual mode phones in order to roam between incompatible analog and digital networks within the U.S.

Manipulating Interface Standards

based on open standards. The costs we discuss in this section are not (necessarily) associated with anticompetitive behavior: they can occur in competitive markets. These costs are a consequence of the complementarities inherent in complex technological systems. With complementarities, consumers may be better off with production of *systems* consisting of components that connect through proprietary interfaces. In such cases, there may be sufficient benefits from competition between systems to outweigh the foregone benefits of component-wise (mix-and-match) competition.

1. Reduction in system design variety

Systems competition with the resulting differentiation between system architectures may provide benefits by increasing variety. When interfaces are proprietary, a firm that wishes to enter with a new, innovative design in one component may find it necessary to develop an entire system. The result may be an increase in variety. The entry of the NeXT computer in the late 1980s may be an example. NeXT introduced a new operating system that took greater advantage of the object-oriented programming model than any other desktop operating system. NeXT also produced its own hardware on which to run this operating system, introducing innovations in digital signal processing, raster-oriented (Display Postscript) screen output, mass storage (magneto-optical drives), and other features.

The innovative variety induced by systems competition must be balanced against the innovative variety from component competition. For example, in microprocessor markets there has been systems innovation as various manufacturers have attempted to compete against Microsoft and Intel by offering incompatible systems. This has led to the development of important microprocessors such as the PowerPC, RISC architectures, the MIPS, and others. However, except for the PowerPC used on Apple Macintosh and Sun's SPARC processors, most non-x86 chips have mostly failed in the market for desktop computers.⁸

Component competition has produced important variety as well. The hundreds of millions of users of Windows-based software demand systems that are *homogeneous* in terms of their compatibility with Windows. As a consequence, there has been substantial variety and innovation encouraged within components even when the interfaces are standardized. For example, while maintaining compatibility with the x86 interface

⁸ Nevertheless, Intel's Albert Yu (1998) describes several instances in which new processor ideas were developed in RISC chips that led to later improvements in Intel chips.

Manipulating Interface Standards

standards, firms other than Intel pioneered low-power microprocessors for mobile computers; Cyrix's MediaGX microprocessor spawned the sub-\$1000 PC market;⁹ AMD and Intel have been leapfrogging each other in a race for the fastest processors; and so forth.

2. Excess inertia

When network externalities are significant and there are already many users of a given standardized system, the incentives to innovate and develop a better system may be insufficient. Even if a firm does develop a better system, consumers may find it too costly to switch (in part because they do not believe that enough other users will switch). Thus some valuable new technologies may not be developed or adopted (Katz and Shapiro 1994; Farrell and Saloner 1986).

3. Insufficient friction

The opposite of excess inertia can also occur when network externalities are significant: if consumers become convinced that a new technology will succeed, the market may "tip" to that technology, even if consumers were better off with the old technology (Katz and Shapiro 1986, 1992).

The latter two problems – excess inertia and insufficient friction – are problems that lead to adopting too many or too few innovations. When network externalities are significant, these socially undesirable outcomes may occur in a market with open standards and component, rather than systems, competition. For example, in a market with competition among several incompatible systems entry by a new, innovative system may be easier than in a market with a single common set of standards (in which entry by a new component is easy), creating a potential excess inertia problem.

C. Summary: Systems versus mix-and-match competition

Manufacturers of complementary components need to know interface specifications in a system so that their components correctly connect and communicate. With open and standardized interface specifications, many firms can make compatible

⁹ The MediaGX combined a microprocessor, memory controller, graphics accelerator and PCI interface on a single chip. At the time competing offerings would have required at least a processor plus the north bridge of a chipset to match this functionality. *Microprocessor Report* (1997) attributes the MediaGX's success with driving Intel to finally breach the \$106 price floor it had long maintained for its mainstream processors.

Manipulating Interface Standards

components on both sides of the interface, and thus component competition will be viable. As described above, there are numerous benefits of component competition for consumers and, in some situations, there are offsetting costs. In some industries, these offsetting costs are sufficient that consumers are better served by systems competition, marked by proprietary interfaces and components that only work with specific matching complements.

For the most part, the history of the PC industry has been marked by component-based competition, and the availability of open standards has been credited with the high rate of innovation, the variety of low cost, high performance products, and the overwhelming success of the PC architecture against closed systems such as the Apple Macintosh and various RISC-based systems. Both systems and component competition have been dominant in different parts of the telecommunications industry.

Based on economic theory alone, we cannot conclude unambiguously that component-based competition and open interface standards are always best for consumers and the economy. We now turn to ways in which the interface standard setting process itself can be used to harm competition and consumers.

III. Anticompetitive Manipulation of Interface Standards

When a combination of firms together have power in at least one of the two markets for components on the two sides of an interface, they may be able to use the process of setting interface standards in order to increase or maintain market power, in the process harming consumers and society. However, to harm competition, a standards consortium must satisfy the two usual conditions: it must have market power and be protected by barriers to entry in order to successfully exercise that market power. In the case of standards, a consortium needs to include firms with sufficient market power in one or both of the component markets to ensure industry adoption of the standard. Second, to protect against competitive dilution, the consortium needs to restrict membership and decision-making control in a manner that excludes viable potential competitors. If not, *i.e.*, if any competent and interested firm could participate, and the decision process was not biased so that a subset of the members could exert effective control, then it would be hard for the consortium to implement anticompetitive strategies.¹⁰

¹⁰ Although restricted voting can enable a consortium to harm competition, it may not be necessary to force democratic participation and fair voting rules in order to protect competition. It may be sufficient to

Manipulating Interface Standards

The European Commission discussed precisely these conditions in its *X/Open Group* decision. It was concerned with market power because the case involved a standard-setting group of computer firms that were each of considerable size. The Commission also noted that it was possible for the members to exclude competing firms from membership. The Commission concluded that “an appreciable distortion of competition...may result from future decisions of the Group” (European Commission 1987 at ¶34).

The mere fact that a consortium of firms has formed to set standards does not imply that consumers and competition will be harmed. This is why agreements made by standards groups are generally not found to be *per se* illegal in the way that price fixing agreements are. We now describe some strategies such consortia might employ that do have anticompetitive effects.

A. Withholding or delaying information

A consortium may withhold necessary interface information from potential rivals for a short or long period. By withholding necessary information, the consortium effectively renders part or all of the interface specification temporarily or permanently proprietary (Farrell and Saloner 1992). Withholding necessary information raises rivals’ costs (thus raising the prices available to end users), and may deter entry (or hasten exit) altogether (Matutes and Regibeau 1996). In particular, if crucial interface information is withheld for long enough, a potential rival is forced to develop a complete system, in which it controls the interfaces, and then to compete on a systems basis. Developing and competing on a full systems basis may be technically infeasible or prohibitively costly.

B. Charging a toll

A second strategy through which an interface specification consortium can harm component competition is to design royalty-bearing intellectual property into the standard. Suppose one firm in the consortium holds a patent on a technology that is useful but not essential for the interface. That is, the interface could be designed without the patented technology and be equally efficient. The patent holder might induce the consortium to specify the interface so that it is necessary to use the patented technology and thus to pay royalties. For example, the patent holder might offer consortium

require that all information shared by consortium members be made simultaneously available to all other competent and interested firms.

Manipulating Interface Standards

members a reduced or zero royalty, with the result that their rivals' costs will be higher.

Sometimes a patent holder might be able unilaterally to deceive a consortium into including its patented technology in a specification. Often there are long delays between the date a firm files a claim and the grant of the patent. The consortium may not realize that a technology written into a specification is covered by such a "submarine" patent.¹¹ If the patent is granted after the specification is released and adopted as a standard by the industry, the patent holder may successfully raise its rivals costs through the royalties it demands.

C. One-Way Interface Standards

A standard can facilitate competition in components on one side of an interface, while restricting competition in components on the other side of the interface. We call this a one-way standard.

Implementing a one-way standard is not straightforward. Since a standard specifies both sides of an interface, it might seem that the consortium need merely withhold the specification information for one side of the interface. In fact, it is the nature of interface standards that manufacturers of components on both sides need all of the information about both sides of the interface. To understand this requires a bit more detail about interface standards.

Consider a simple interface: see figure 1. We have illustrated here the communications part of an interface standard, known as the *protocol*. For communications and computing systems, interfaces involve the sending of signals between components. The protocol specifies the language for the signaling, including a syntax and vocabulary. In the figure we show a piece of customer premises equipment (CPE), and a communications switch. The CPE sends queries (and directives) to the switch; the switch responds. Likewise, the switch queries the CPE, which in turn responds. The protocol specifies the permissible queries, and the responses that can be

¹¹ There are many cases in which patent claimants exploited Patent Office rules to intentionally delay the granting and publication of their patents. The Lemelson machine vision patents are a well-known example, in which delays were created by filing a series of continuation and divisional patent applications that claim priority from the initial patent application. The Federal Circuit recently ruled that a patent may be unenforceable if the patent applicant unreasonably delays prosecuting the patent (*Symbol Techs. Inc. v. Lemelson Med., Ed. & Research Found.*, 277 F.3d 1361 (Fed. Cir. 2002)). U.S. patent law was recently amended by the American Inventors Protection Act of 1999 to limit submarine patents. Claims filed after 29 November 2000 will automatically be published 18 months after they are filed, even if the review process is not complete.

Manipulating Interface Standards

generated to each query. If the CPE complies with the specification, it knows what

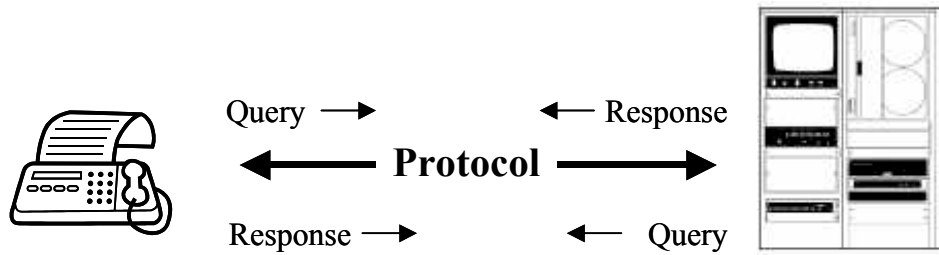


Figure 0: Telecom interface protocol

messages it can send, and it knows what responses the switch can give. If it didn't know the responses the switch could give, then it could not be programmed to make use of those responses. Likewise, a compliant CPE knows what queries it can receive, and what responses it is expected to give.

From the example it should be evident that the components on both sides of the interface must know the full specification. The CPE must know not only its permissible queries, and its responses to the switch; it must also know the switch's permissible responses, and the queries it can receive from the switch. It is not possible to publish only one side of the specification and design components on that side.

How, then, can a standards consortium design a one-way standard? The basic idea is to create a "drop-out", or a "blind": to insert an additional structure between the two components. We call this structure a *translator*. Now, the interface specification between one component and the translator can be published, but the interface specification between the translator and the other component is treated as proprietary and is not published. Manufacturers on the open side can manufacture compliant components that communicate with the translator, but non-member manufacturers on the other side cannot make their components communicate with the translator.

To illustrate, consider the following example of how the protocol in figure 1 would break down if both sides were not published. Imagine that both devices could be programmed to communicate in natural language, and that the CPE was programmed to communicate in English. However, the switch is programmed to communicate in another language, and the CPE does not know what language the switch is using, nor how to speak it. Clearly, the CPE and the switch cannot interoperate successfully.

Now introduce a translator (figure 2). The interface protocol between the CPE and the translator can specify that they speak English to each other. The language spoken

Manipulating Interface Standards

between the translator and the switch can be kept secret. Anyone can manufacture compliant CPE, but only those consortium members who know the secret language spoken by the translator can manufacture switches.



Figure 0: Telecom interface with translator

There is a simple and reasonably familiar way to implement a one-way standard, at least conceptually. The standard could specify that the two components communicate via public key cryptography (PKC). The standard would publish a “public key” and an algorithm that components on one side could use to encrypt messages sent to the other side, and to decrypt messages arriving from the other side. Components on the other side would need the corresponding “private key” to decrypt messages encrypted with the public key, and to encrypt messages that could be decrypted with the public key. As long as the component manufacturers on the protected side of the interface kept the private key secret, no other manufacturer could make a component that could communicate with the public side components.¹²

The effect of a one-way standard is to extend the boundary of systems competition. Continuing with the example, the switch in figure 1 is a system.¹³ To compete in switches, manufacturers need to implement all of the features that switch users expect (in particular, the ability to communicate with external components through specified interfaces). Thus, there is *systems competition* in switches. Suppose that when a

¹² It is unlikely that PKC would actually be used for this purpose for at least two reasons. First, the private key would need to be hard-coded into the physical components, and then it would likely be a straightforward matter for competing firms to discover it. Second, PKC imposes substantial computational overhead, and would not be practical for the many very fast, very short messages that communications and computing devices exchange. PKC was first proposed by Diffie and Hellman (1976); the most widely used implementation is RSA (Rivest, Shamir, Adelman 1978).

¹³ The switch surely is a system: that is, a set of complementary components that communicate with each other to collectively perform services for users.

Manipulating Interface Standards

one-way standard is imposed, as in figure 2, the switch is on the proprietary side. Now, a potential competitor who previously would have designed complete switch systems to compete, must design switch plus translator systems. That is, since the specification between the CPE and the translator is public, potential switch competitors can connect to CPE if they develop their own translators that conform to the public CPE-translator standard. The system boundary has expanded to include the translator device.¹⁴

Expanding the system boundary is a variation on raising rivals' costs. It may be possible to design and market expanded systems (that include proprietary translators), but it takes time and money to do so. If the translator design is sufficiently costly or time-consuming, then firms excluded from the standards consortium may find it very difficult to compete effectively.

D. Timing is critical

Timing is a crucial element in the above strategies. In the communications and computing industries, technological innovation is so constant and rapid that significant delays in time to market can mean the difference between vibrant, successful competition and a persistent pattern of dominance with minor fringe competition. Thus, none of the above strategies needs to be leak-proof or permanent. As long as the dominant firm can impose the above competitive disadvantages for as little as a few months or a year, the effects on competition can be devastating. This is particularly so because the ongoing cycle of innovation gives that the dominant firm the opportunity to put its competitors "on the treadmill". For example, with one-way standards, a dominant firm could introduce one translator after another, for each new or revised interface that arises. Potential competitors would bear an ongoing stream of higher costs and delays in getting to market.

The U.S. Federal Trade Commission (1999) makes this point quite forcefully in its analysis of Intel's conduct published along with the consent decree entered into by Intel and the FTC:

"The computer industry is characterized by short, dynamic product cycles, which are generally measured in months. Time to market is crucial.

¹⁴ Notice that this strategy is similar to tying as a foreclosure strategy: a firm with monopoly power over good A requires consumers to purchase good B if they want to get good A. If demand is sufficient for good A, this may harm competition in the market for good B. In order for a good B producer to effectively compete, it may have to develop its own version of good A, so that it can offer consumers a complete package of A and B.

Manipulating Interface Standards

Indeed, the denial of advance product information is virtually tantamount to a denial of actual parts, because an OEM customer lacking such information simply cannot design new computer systems on a competitive schedule with other OEMs. An OEM who [sic] suffers denial of such information over a period of months will lose much of the profits it might otherwise have earned even from a successful new computer model. Continued denial of advance technical information to an OEM by a dominant supplier can make a customer's very existence as an OEM untenable.”

The European Commission noted the same concern in the context of a standards consortium (1987 at ¶32):

“In an industry where lead time can be a factor of considerable importance, membership of the group may thus confer an appreciable competitive advantage on the members vis-à-vis their hardware and software competitors this advantage in lead time directly affects the market entry possibilities of non-members”

That is, it is not necessary for a standard-setting consortium to withhold the interface specification standard forever for competition to be harmed. If the member firms have advance knowledge of the standard they can bring compliant products to market before non-members, and in the communications and computing industries even a few months of lead-time can spell the difference between market success and failure.

IV. Examples of Possible Anticompetitive Interface Specifications

In this section, we examine three examples of possibly anticompetitive interface standards in the computer industry. In one example the consortium incorporated patented information in a memory standard; in another, a monopolist established a one-way standard for graphics processors; and in the third a consortium imposed a one-way standard and also gained competitive advantage by delaying the release of necessary specification details for a peripheral standard.

Before discussing examples of standards consortia in the computer industry, we briefly describe some relevant technological and economic characteristics of microprocessors. By themselves microprocessors have little or no value to end users. Microprocessors can process computational instructions, but they need software to deliver the instructions. They also need a variety of other devices that assist in performing the tasks that end users desire. For example, microprocessors need memory to hold data and instructions (which end users demand in a variety of configurations: DRAM, hard disks, floppy disks, CD-ROMs, etc.). Microprocessors need input and

Manipulating Interface Standards

output devices (keyboards, scanners, microphones, cameras, printers, monitors, voice and data network lines). For all of the above, the microprocessors need communications pathways and devices that manage the vast variety of complex and extremely high-speed signals flowing among all of the various devices. In short, end users demand computer systems, of which microprocessors are but one component. The systems, in turn, are comprised of numerous components. Between these components are a variety of interfaces.

In the microprocessor industry, many consortia exist to create standards for the interfaces between hardware devices that connect to a personal computer's microprocessor or to the microprocessor's associated chipset. Many, if not most, of these consortia have closed membership, and the members of the consortia both control the details of the interface standards and have advance knowledge of the interface details, enabling consortia members substantial lead-time in developing compatible products.

Both systems and component competition occur in the computer industry. When standards are proprietary, competition must take place on a systems basis. An example is the current technology for microprocessors and chipsets. In the mid-1990s Intel made the bus that connects its microprocessor to chipsets proprietary. Since then, Intel-compatible microprocessors and chipsets compete as a system against AMD-compatible microprocessors and chipsets.¹⁵

When interface specifications are open and standardized, it is possible for multiple firms to compete for the manufacture of a given component for use in the same system. This is known as component competition. An example is the competition among Maxtor, Seagate, IBM, Fujitsu and others to make and sell hard drives that are used in personal computers manufactured by Dell, Compaq, Vobis, Groupe Bull, and others.

A. Inserting Patents in Standards: Rambus

Rambus is a designer of memory for computers. It apparently engaged in a strategy of assisting the creation of a standard that contained patented Rambus technology.¹⁶ Rambus expected to collect royalties of \$1 to \$2 on every PC sold in recent years.

¹⁵ Intel making the bus proprietary and thus expanding the boundary of its microprocessor system to include chipsets is an example of a one-way standard.

¹⁶ The Rambus case is a bit unusual because it appears that Rambus did not actively participate in the standards consortium, although it was a member. It has been described as a "mole" that was watching to see which of its technologies were being adopted, and then going back to revise its pending patent applications to ensure they covered the new standard.

Manipulating Interface Standards

Rambus filed patent applications in 1990, then joined the JEDEC memory standards consortium in 1992. Rambus failed to disclose any of its pending patent applications, as required by JEDEC rules. JEDEC issued its specification for SDRAM memory in 1996. Rambus filed amended patent applications in 1997 to cover SDRAM technology; these patents were awarded in 1999 and 2000. Rambus then filed suits against most of the major memory makers alleging infringement (Infineon, Micron and Hyundai, which is now Hynix). A U.S. District Court ruled in May 2001 that Rambus committed fraud by failing to properly disclose the patent information to the JEDEC consortium while it was a member. In June 2002 the FTC filed an antitrust suit against Rambus (Kanellos 2002).

B. One-Way Standards: AGP

The Accelerated Graphics Port was developed by Intel. “The Accelerated Graphics Port (AGP) interface is a new platform bus specification that enables high performance graphics capabilities especially 3D, on PCs at mainstream price points.”¹⁷ In this case, the standard was unilaterally defined by Intel. There is now an AGP Forum, of which Intel is the sponsor. Other firms can become members for \$2,500 a year, with the benefits of “Participation in events and technical support subject to availability.” Intel has the right to limit the number of participants or to discontinue the program altogether.¹⁸ Intel maintains unilateral control over the standard. This is a case in which the standards consortium that implemented the one-way standard is essentially a single firm.

The AGP has electrical specifications on one side, between the AGP and the peripherals, and software specifications on the other side, between the AGP and the chipset. Intel has made the electrical specifications public, which means that firms can manufacture peripherals that will interoperate with AGP. However, chipset makers cannot manufacture on their side of the standard because the software specifications are not public.¹⁹ Chipset manufacturers may be able to “invent around” the software specifications.²⁰ However, because Intel has unilateral control over the standard, it can

¹⁷ From the AGP Forum Web page, <http://www.agpforum.org/>, accessed 1 September 2002.

¹⁸ See the membership application for the AGP Forum at <http://www.agpforum.org/membform.htm>, accessed 1 September 2002.

¹⁹ The Draft AGP V3.0 Interface Specifications, revision 0.95, May 2001, contains a chapter on the physical layer specification, but not the software layer. See http://www.intel.com/technology/agp/agp_draft9.htm, accessed 1 September 2002.

²⁰ AMD plans to include AGP on its new eighth-generation platform in 2003 (Techworthy.com 2002).

Manipulating Interface Standards

change it at will to keep other chipset manufacturers on the treadmill.

C. One-Way Standards and Publication Delay: USB 2.0

The Universal Serial Bus (USB) is a standard for the microprocessor to communicate with slow- and medium-speed peripherals such as mice, keyboards, printers, scanners, and digital cameras. It defines an interface between a host controller and the peripherals. The host controller, which can be an independent physical device or which can be integrated onto the chipset, speaks with the system software via the host controller interface. Thus, we have two interfaces working together: the USB interface between the peripherals and the host controller, and the host controller interface between the host controller and the system software.

USB 1.1 was originally developed in 1995. Two host controller interfaces, OHCI (Open Host Controller Interface) and UHCI (Universal Host Controller Interface), work with USB 1.1. UHCI is Intel's proprietary interface, and is available via a royalty-free, reciprocal license for adopters of USB. OHCI was jointly developed by Compaq, Microsoft, and National Semiconductor and is an open standard, available for download at <http://www.compaq.com/productinfo/development/openhci.html> (Compaq 1997).

USB 2.0 increases the speed of the peripheral to PC connection by forty times relative to USB 1.1. USB 2.0 was completed in April 2000.²¹ USB 2.0 is only compatible with a new host controller interface, EHCI (Enhanced Host Controller Interface). Version 0.95 of the EHCI was made public in November 2000, and Version 1.0 was released in April 2002, a full two years after the USB 2.0 specification was published. The EHCI interface is proprietary to Intel, licensed only in exchange for the grant of a royalty-free license to Intel on a firm's related intellectual property. Until May 2002, implementation of USB 2.0 required the use of a host controller, a separate add-on piece to the chipset. In May 2002, the first chipsets with integrated host controllers were announced.

In Table 1, we present a time-line of some of the more important developments in the history of USB 2.0.

²¹ A beta version was published in October 1999.

Manipulating Interface Standards

Table 1: Timeline for USB 2.0 and EHCI specifications

Date	Event
4/12/00	NEC (a consortium member) developed the world's first USB 2.0 host controller. Sampling to begin in May, volume production in September.
4/27/00	USB 2.0 specification released
8/21/00	Lucent (a consortium member) releases a USB host controller. Plans to release samples by October 00 and mass launch in Dec 00.
11/10/00	Intel publishes the EHCI 0.95 specification.
11/14/00	NEC's USB 2.0 host controller has passed compliance tests.
2/26/01	NEC begins volume production of the first USB 2.0 host controller. Intel EHCI compliant.
5/9/01	Philips (a consortium member) announces its host controller. Has two OHCI (for USB 1.1) and one EHCI (for USB 2.0). Sampling now, volume production for late fall 2001.
7/01	Intel announces it will have P4 chipsets with integrated USB 2.0 host controller in the first half of 2002.
12/01	Microsoft introduced a USB 2.0 driver for Windows XP and Windows 2000 that currently supports ONLY NEC's host controller.
4/17/02	Intel releases final version of EHCI 1.0 spec. Licensed on a reciprocal, royalty-free basis.
5/15/02	Via's (not a consortium member) P4X333 chipset integrates USB 2.0. Patent dispute with Intel over whether Via can produce P4 chipsets or not.
5/20/02	Intel (a consortium member) releases chipsets that integrate USB 2.0.

1. The Standard-Setting Consortium

Compaq, Intel, Microsoft, and NEC were the original firms behind USB 1.1. For USB 2.0, these four were joined by Hewlett-Packard, Lucent, and Philips. These collaborators created and control the interface specification standard for version 2.0 of the Universal Serial Bus (USB 2.0).²² Jason Ziller of Intel serves as Chairman and President of the Board of Directors of the USB Implementers Forum. The formation of the USB 2.0 Promoter Group was announced at the Intel Developer Forum in Spring 1999.

²² The by-laws are available from http://www.usb.org/data/retail/usbif_bylaws.pdf, accessed 20 July 2001.

Manipulating Interface Standards

The USB Implementers Forum has two membership classes: Promoter Members, who have voting rights, and Participant Members, who do not have voting rights. Promoter Members must be engaged in research and development of the USB specifications. The Board of Directors, made up of employees of Promoter Members, has sole discretion to accept or reject applications from other firms to become a voting member. To become a Promoter Member, one must receive unanimous approval of the Promoter Members; any individual Promoter Member has veto power over a Promoter Membership application. Thus, the consortium satisfies one of the criteria that enable a consortium to behave anti-competitively: membership is limited and current members control which firms can become a member.

The voting members are Intel, Compaq, Hewlett-Packard, Lucent, Microsoft, NEC, and Philips. Intel is the only voting member that manufactures microprocessors and chipsets. Microsoft is of course the dominant firm in operating systems and many applications. Compaq and Hewlett-Packard recently merged are now the largest PC OEM. Lucent, NEC, and Philips are firms that produce a variety of complementary components to microprocessors; these firms are competitors with each other.

As described above, this is a situation with two devices – peripherals and the microprocessor (and system software) – separated by a “translator”, the host controller (cf. figure 2). The consortium does not include any firms that produce chipsets or microprocessors except for Intel. Intel has an incentive, then, to manipulate the standard-setting process in such a way as to advantage itself against other microprocessor firms (chiefly AMD) and other chipset firms (e.g., VIA Technologies).

The second criterion for a standards consortium to have the potential to manipulate anticompetitively a standard is it include firms with sufficient influence ensure that the standard is adopted. In this case, Intel and Microsoft together have the ability to ensure industry-wide adoption of a standard.

2. Anti-Competitive Behavior?

The consortium has behaved in a way that is consistent with two of our anti-competitive strategies: implementing a one-way standard and delaying the release of information about the standard.

A one-way standard was implemented because the host controller interface,

Manipulating Interface Standards

EHCI, is proprietary to Intel, while the USB 2.0 interface is an open interface.²³ That is, the consortium has released the specification information necessary for makers of complementary peripherals to implement their side of the USB 2.0 interfaces. The EHCI is required to implement the chipset/motherboard side of the USB 2.0 interface. Thus, microprocessor and chipset manufacturers who are excluded from the consortium are unable to provide the necessary functions in their devices.

In addition to the one-way nature of the standard, consortium members had a competitive advantage through early access to the USB 2.0 and EHCI specifications. That is, any firm has been able to build a peripheral that is USB 2.0 compliant, but only consortium members have been able to build the system-side hardware. For example, NEC announced that it had developed the world's first USB 2.0 and EHCI-compliant host controller on 12 April 2000, 15 days before the USB 2.0 interface was released and six months before the EHCI interface was released. In August 2000, Lucent announced a host controller;²⁴ and in May 2001, Philips announced a host controller. Intel previewed a chipset with USB 2.0 in August 2001 and released core logic chipsets with integrated USB 2.0 20 May 2002. Not until two years later did a non-member of the consortium announce implementation of a USB 2.0 compliant chipset or host controller.²⁵

V. Conclusion

We have described the circumstances under which firms can use the standard-setting process in an anti-competitive manner. Anti-competitive strategies include manipulating standards to include a firm's patented IP; using information gained from within the consortium to gain a time-to-market advantage; and creating standards that are "one-way". This last we believe has not been discussed in the prior literature.

²³ Intel did not publish the EHCI specification until six months after the USB 2.0 specification was published. Further, although published, the EHCI specification is still proprietary and must be licensed from Intel. In exchange for using the specification, a firm must relinquish its related intellectual property to Intel with zero royalties.

²⁴ We believe that Lucent did not succeed in manufacturing this host controller until at least May 2001.

²⁵ In May 2002, VIA announced a chipset that integrates a USB 2.0 host controller. However, Intel is challenging VIA's legal right to produce a P4 chipset.

VI. References

- Compaq, 17 September 1997, "USB and the Difference Between OHCI and UHCI", <http://www.compaq.com/support/techpubs/whitepapers/ecg0480997.html>, accessed 1 September 2002.
- Diffie, W. and M. Hellman, New directions in cryptography, *IEEE Trans. Inform. Theory*, 22 (1976), pp. 644-654.
- Economides, Nicholas, December 1988, "Desirability of Compatibility in the Absence of Network Externalities," *American Economic Review*, vol. 79, pp. 1165-81.
- European Commission, *X/Open Group*, 1987, Commission Decision 87/69/EEC O.J.L. 35/36.
- Farrell, Joseph and Garth Saloner, March 1992, "Converters, Compatibility and Control of Interfaces," *Journal of Industrial Economics*, Vol. 40(1), pp. 9-35.
- Farrell, Joseph and Garth Saloner, 1986, "Economic Issues in Standardization," *Telecommunications and Equity: Policy Research Issues*, J. Miller, ed., (North-Holland: Amsterdam).
- Farrell, Joseph and Garth Saloner, December 1986, "Installed Base and Compatibility: Innovation, Product Preannouncements, and Predation," *American Economic Review*, Vol. 76, pp. 940-55.
- Hemenway, D., 1975, *Industrywide Voluntary Product Standards* (Ballinger: Lexington, MA).
- Kanellos, Michael, 19 June 2002, "Rambus target of FTC antitrust suit", CNET News.com, <http://news.com.com/2100-1001-937449.html>, accessed 31 August 2002.
- Katz, Michael and Carl Shapiro, March 1992, "Product Introduction with Network Externalities," *Journal of Industrial Economics*, Vol. 40, pp. 55-84.
- Katz, Michael and Carl Shapiro, Spring 1994, "Systems Competition and Network Effects," *Journal of Economic Perspectives*, Vol. 8(2), pp. 93-115.
- Katz, Michael and Carl Shapiro, August 1986, "Technology Adoption in the Presence of Network Externalities," *Journal of Political Economy*, Vol. 94, pp. 822-41.
- Matutes, Carmen and Pierre Regibeau, Summer 1988, "Mix and Match: Product Compatibility Without Network Externalities," *RAND Journal of Economics*, Vol. 19, pp. 221-34.
- Matutes, Carmen and Pierre Regibeau, 1996, "A Selective Review Of The Economics Of Standardization: Entry Deterrence, Technological Progress And International Competition", *European Journal of Political Economy*, Vol. 12, pp. 183-209.
- Microprocessor Report*, 10 March 1997, "MediaGX Targets Low-cost PCs".
- Microprocessor Report*, 14 July 1997, "Intel Fumbles MMX Transition".
- Porter, Michael, 1985, *Competitive Strategy* (Free Press: New York).

Manipulating Interface Standards

- Rivest, R., A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Comm. ACM*, 21 (1978).
- Techworthy.com, “AMD Demonstrates New Graphics Standard, AGP-8X, on AMD’s Future Eighth-Generation Processor, 4 June 2002, <http://bedford.nyws.com/NewsFull.asp?ID=277>, accessed 1 September 2002.
- U.S. Federal Trade Commission, “Analysis of Proposed Consent Order to Aid Public Comment”, FTC v. Intel, Docket No. 9288, available from <http://www.ftc.gov/os/1999/9903/d09288intelanalysis.htm>, accessed 23 July 2001.
- Whinston, Michael, 1990, “Tying, Foreclosure, and Exclusion,” *American Economic Review*, Vol. 80, pp. 837-859.
- White, Charlie, 21 May 2001, “1394 Trade Association Intros New 1394 Multimedia Standard”, *Digital Video Editing*, http://www.digitalvideoediting.com/cgi-bin/getframeletter.cgi?/2001/05_may/news/1394b_standard.htm (accessed 28 July 2001).
- Wilcox, Joe, 25 August 2000, “Toshiba's notebook sales slide”, CNET News.com, <http://news.cnet.com/news/0-1006-200-2613994.html?tag=rltdnws>, accessed 21 July 2001.
- Yu, Albert, 1998, *Creating the Digital Future: The Secrets of Consistent Innovation at Intel* (The Free Press: New York).