Center for
eBusiness@MIT
http://ebusiness.mit.edu

A research and education initiative at the MIT Sloan School of Management

Driving High-Tech Innovation:
The Four Levers of Platform Leadership

Paper 152

Michael Cusumano
Annabelle Gawer

October 2001

For more information,
please visit our website at **http://ebusiness.mit.edu**
or contact the Center directly at **ebusiness@mit.edu or 617-253-7054**

MITSloan

Driving High-Tech Innovation:
The Four Levers of Platform Leadership

Michael A. Cusumano (MIT Sloan School of Management)

and Annabelle Gawer (INSEAD)

Draft:
October 18, 2001

Contact Information:

Cusumano@mit.edu

Annabelle.GAWER@insead.fr

> [W]e are tied to innovations by others to make our innovation valuable. If we do innovation in the processor and Microsoft or independent software parties don't do a corresponding innovation, our innovation will be worthless. So, it really is a *desperate situation* for us.[1]

This is a quote from David Johnson, one of the directors of the Intel Architecture Lab (IAL). IAL is a facility of some 700 people that Intel founded in 1991 to develop interface standards and technologies to advance the personal computer. He was talking in a recent interview with us about the microprocessor, the "computer on a chip" that was Intel's main product and the core hardware component of the personal computer. In 2001, Intel provided nearly 85 percent of the microprocessors that went into personal computers as well as an increasing number of microprocessors for other programmable devices. Since almost everything was becoming programmable, it seemed like Intel sat atop the high-tech equivalent of a gold mine. Indeed, in fiscal year 2000, before the downturn in the high-tech market, Intel had revenues of nearly $34 billion and net profits of more than $10.5 billion.

The reality for Intel, though, as this manager put it, was a "desperate situation." The reason is that Intel microprocessors could do little or nothing useful by themselves. The microprocessor was a key component in a broader platform or "system" – the personal computer. The value of the PC system, and core components such as the Intel microprocessor, depended on products that other firms have to create – such as the software operating system, software applications, software development tools, and hardware components like monitors, keyboards, storage devices, and memory chips. These "complementary products" helped fuel the remarkable growth of the PC market and companies such as Intel. But there was no guarantee that complements producers would continue to produce these innovations at the same rate as, for example, Intel introduced new microprocessors.

Although several definitions exist for what a platform is, we define it as an evolving "system" made of interdependent pieces, each of which can be innovated upon separately. In the case of Intel, the target platform is the personal computer. Intel is a supplier of a key component – the microprocessor. Microsoft produces another key component – the software operating system. Neither the microprocessor nor the operating system, like the PC platform itself, has much value with other components of the platform that make up the PC system. Other common platform products with somewhat simpler designs include the video recorder and the compact disc player, neither of which has much value without prerecorded tapes and TV broadcast signals, or prerecorded music discs.

Producers of key components for a platform such as the PC exist in a state of continual desperation, despite their usually enviable positions as market leaders, because their future as well as their present market positions depends heavily on the decisions of other firms to invest in complementary innovations. For example, Intel spent around $2 billion developing and preparing to manufacture a new 64-bit microprocessor (the Pentium 4), which it introduced in mid-2000. This new product doubled the information-processing capacity of the current generation of 32-bit microprocessors, which handled

information in batches of only 32 bits, and truly brought mainframe-computer power to the desktop PC. Yet Intel's latest innovation provided little benefit to users unless hundreds of companies, beginning with giants like Microsoft, Dell, and Compaq, redesigned their hardware and software offerings to take advantage of the new 64-bit architecture. Another manager in the Intel Architecture Lab, Bala Cadambi, used an analogy from the automobile industry to explain Intel's predicament and goals as a company that is trying to advance the PC as a platform:

> Intel is in the business of providing the engine for the PC, just like Honda would be in the business of providing the engine for the automobile. That engine is doubling in capacity every 18 to 24 months (that's Moore's Law). It increases the capacity in terms of efficiency, scalability, power, and the kind of things it can do with multimedia. What we really want is to ensure that the rest of the platform goes with it. This means that, if the engine gets better, the tires get better, the chassis gets better, the roads get better, you get better gas mileage. You can have navigation systems that are scaleable. Everything that goes with having a better experience. *The platform around the engine limits the engine. So we want the platform — which is everything that's around the microprocessor — to be keeping pace and improving and scaling, such that the microprocessor can deliver its potential*.[2]

We consider both Intel and Microsoft "platform leaders" for the personal computer because of their influence over the PC system architecture as well as over the network of firms that produce needed complements. But platform leaders such as Intel and Microsoft face at least three problems. One issue is how to maintain the integrity of the platform (i.e., compatibility with complementary products) in the face of continuous technological innovation and the independent product strategies of outside firms, whichcan threaten the internal coherence of the platform and its interfaces. Another, related problem is platform evolution: Platforms have to evolve technologically or they will become obsolete. How to do this and maintain compatibility with past complements is often a vexing technical challenge. A third problem is how to preserve or achieve market leadership in platform environments – how to get to be a platform leader. This article offers a framework to help platform leaders and leader-wannabes think about these issues. Our observations center around the Intel case, with some comparisons to Microsoft, Cisco, Palm, and NTT DoCoMo.[3] Though beyond the scope of this article, we also believe that understanding how these companies work can provide useful insights for the thousands of firms that fall into the category of providing complementary products for the dominant platform in their industry.

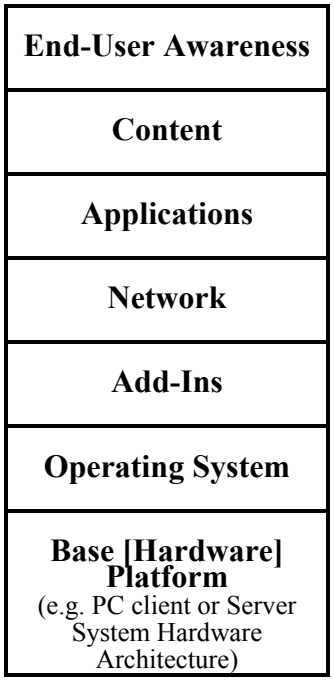
The Essence of Platform Leadership

Most platform leaders do not have the capabilities or resources to create all complements – complete systems – themselves. As a result, they usually need to work closely with other firms. The combined efforts of platform leaders and complementary innovators, together, can increase the potential size of the pie for everyone. For example, to deliver PC capabilities to the end-user required coordination between different actors in the industry. In fact, most Intel managers we interviewed referred to the need for some sort of coordination between firms in order for their innovations to be valuable to

consumers. Not surprisingly, Intel managers thought a lot about how to direct this coordination toward improving the PC platform as a system. Dave Ryan, director of technology marketing in the Intel Architecture Lab, highlighted the importance of coordination in the industry:

> This picture of a stack [see Figure] represents the industry layers with which we need to interact in order to deliver a complete capability to computer users. We frequently find that, for a technology advance to be delivered to the computer user, or for the computer user to have a new capability that may be unrelated to a particular hardware advance, changes in many of these layers need to occur accordingly. For the user actually to see that new capability, some companies need to make new hardware. Some companies need to make changes in the operating system. There need to be new types of networking or new products developed, or new applications developed and Internet changes that may be needed to take advantage of this new opportunity to deliver content. So, to get a new use or new capability to occur, we have to work simultaneously across all these layers. The players in the layers are different. These are really different industries and very different companies, with different marketing requirements in each of those areas.[4]

Figure: Industry Capability Stack



Source: Intel in-house materials. Used with permission.

Platform leadership is the ability of a firm to drive innovation around a particular platform technology at the broad industry level. Leadership is important because, in many industries, the value of the platform system (it is often a product itself) increases when there are more complements available (such as more VHS tapes, or more Windows

applications). The economics and strategy literature has discussed this dynamic for years,[5] using terms such as "network externalities" and "bandwagon" or "positive feedback" effects. The more people who use these kinds of platform products, the more incentives there are for complement producers to introduce more complementary products, which then stimulate more people to buy or use these products, stimulating more innovation, ad infinitum. It is, therefore, in the interest of a platform leader to make sure there is lots of innovation on complementary products.

The game is complex, though, because platform leaders can fail to get other firms to cooperate and innovate, and strategic failure can lead to greatly reduced sales for everyone tied to the same platform. Since platforms are made of components that interact following standard interfaces, standards wars, with winners and losers, are usually part of platform strategies. Examples from recent years include Sony and the Betamax standard, which failed to dominate the VCR market after an early lead, and the Apple Macintosh line of personal computers, which has found some resurgence in recent years but has not unseated Windows machines as the mass-market standard. The Navigator browser is another platform-type product that declined sharply in market share as a result of Netscape's inability to keep this product as the industry standard. (It once had a market share of around 90 percent.)

Framework: The Four Levers of Platform Leadership

Based on our analysis of Intel as well as Microsoft, Cisco, and NTT DoCoMo, we developed a framework consisting of four tools or levers (see inset). Managers can use these levers to design a strategy for platform leadership (or to work more effectively with platform leaders). The four levers cover both strategy formulation but also implementation. They are distinct but closely related; therefore, managers need to make choices on these dimensions in a coherent, systematic fashion, as we illustrate in the following discussion.

The Four Levers of Platform Leadership

1. ***Scope of the firm***: This lever deals with what the firm does inside and what it encourages others to do outside. In particular, managers of platform leaders and wannabes need to ask if it is better for them to develop an extensive in-house capability to create their own complements or let the market produce complements. Or they might consider if there is a "happy medium" in between these two extremes, and when that is the best approach.

2. ***Product technology (architecture, interfaces, intellectual property)***: This lever deals with decisions that platform leaders and wannabes need to make with regard to the architecture of their product and the broader platform, if the two are not the same. In particular, they need to make decisions about the degree of modularity, the degree of openness of the interfaces to the platform, and how much information about the platform and its interfaces to disclose to outside firms – potential complementors[6] who may also become competitors.[7]

3. ***Relationships with external complementors***: This lever centers on determining how collaborative versus competitive should the relationship be between the platform leader or wannabe and complementors. Platform leaders and wannabes also need to worry about how to create a sense of consensus, and how to deal with potential conflicts of interest, such as when the platform leader decides to enter complementary markets directly and turn former partners into competitors.

4. ***Internal organization***: This lever allows platform leaders and wannabes to use their internal organizational structure to manage external and internal conflicts of interest more effectively. Options include keeping groups with similar goals under one executive, or separating groups into distinct departments if they have potentially conflicting goals or outside constituencies. The issue of culture and process comes in as well: we found that since innovative, modular industries are often ambiguous environments, where a complementor today can become a competitor tomorrow, an internal atmosphere that encourages internal debate or at least tolerates ambiguity -- in times of decision-making -- accelerate the strategy re-formulations that are sometimes necessary. At the same time, efficient internal communications of corporate strategy, once a decision has been made at the top, facilitate the implementation of strategic re-orientations.

Lever 1: Scope of the Firm
(Make Complements In-House or Not)

A goal of platform leaders is to occupy a central position within a thriving ecosystem or network of innovation that allows them and customers to benefit from complements that make their platform products more valuable. Therefore, determining the scope of the firm – i.e., what complements to make inside and what to leave to external firms – is probably the most important decision that platform leaders have to make. It is also important to recognize that this is usually not a one-time decision; there was almost always the opportunity to do at least part of the innovations in-house and add new functionality to their core products. Another decision is how to stimulate external innovation.

Companies that want to become platform leaders first need to assess how dependent they are on complements. Then they need to determine how to increase demand for their platform. Windows PCs, for example, have tens of thousands of software applications that helped make Windows and the Intel microprocessor standards for the PC platform. Palm stimulated the external development of over 8,500 applications for the Palm OS. NTT DoCoMo encouraged the creation of 40,000 official and unofficial web sites for its services in Japan. Cisco made dozens of acquisitions each year of complementary firms as well as potential substitute technologies that it could tie into its Internet router, a specialized computer for sending packets of information across the Internet.

It makes no sense for a platform leader or wannabe to develop its own complements if it does not have the technical and organizational skills or the financial capacity to compete in the relevant markets. Microsoft had the skills to develop the most popular applications for Windows by itself, even though this decision damaged the businesses of many prior complementors, including Netscape, Novell, WordPerfect, and Lotus. Intel, though it has thousands of software engineers who program microprocessors and do research in complementary technologies, would probably have a very difficult time competing with the likes of Microsoft in making commercial operating systems or software applications. Nor is Intel a consumer electronics company, which suggests that it will have difficulty competing in consumer mass markets, such as with MPEG music players or digital cameras, even though it has entered these businesses, at least to seed them.

Microsoft clearly had the technical skills to move from building PC programming languages and operating systems to software applications. Still, Microsoft generally stayed with "horizontal" applications that fit well with its platform technology, such as Office or broadly used database management programs. Microsoft did well in video game markets for the PC, but we expected the company to have difficulty competing in consumer hardware products such as video game boxes, even though it recently entered this market with its xBox game player.

Cisco provided perhaps the most complex example of utilizing our first lever. It seemed willing to invest in any technology that could work with Internet routers. It did not build many end-user applications itself, but rather expanded through acquisition into different infrastructure products that enabled various types of networking applications. Cisco did not simply rely on the market to provide these different technologies and

applications. Some of these technologies (such as intelligent switches) were also potential substitutes for the router, though Cisco and its customers often used them as complements, tied together by software in hybrid networks.

Another critical issue is whether an actual or potential complements market – a group of firms able to build complements – already exists. If there is no market, then a platform leader or wannabe might well decide to build complements itself. Palm had to make sure that some applications came with its Pilot or the product would have failed immediately. NTT DoCoMo's i-mode service was likely to fail in the United States unless it could line up distinctive compelling content.

When platform leaders or wannabes chose to work with outside firms, they can exert great influence over the design and production of complements by acting on the incentives and capabilities of external developers. For example, platform leaders or wannabes can share technical information about their own products and platform interfaces as well as dispatch engineers to help complementors build compatible products. Or platform leaders can use what Intel called a "rabbit" strategy. This approach targets a promising developer of complements and helps the company in a very visible way so that other firms feel encouraged to follow. The rabbit strategy draws attention from investors and complementors to a potentially lucrative new market and signals to the world that the platform leader aims to stay out of this complementary market while encouraging competition in that market.

The platform leader or wannabe can also develop, as Intel did in its Intel Architecture Labs, enabling technologies such as application programming interfaces and software development kits, and share them either free of charge or for very low royalties, to stimulate the development of complements. In addition, platform leaders or wannabes can share market information and provide marketing funds to help complementors in their efforts.

Venture investments as well as mergers and acquisitions also affect the scope of the firm and its ability to influence complements production. We have seen several platform leaders or wannabes, including Intel, Microsoft, Cisco, and Palm, take equity positions in complementors. Intel, Microsoft, and Cisco have also made many acquisitions. Acquiring a complementor, however, can produce conflicts. An acquisition blurs the line between platform and complement. Therefore, the platform leader becomes a competitor of former partners making the same complements. This situation might lead to reduced incentives for other firms to position themselves in the complementary market, resulting in less competition – and possibly less complementary innovation.

In short, there is no simple answer regarding whether to make complements in-house or not. We do know that platform products need complements. Consequently, there seem to be lots of advantages for a platform leader or wannabe to have at least some in-house complements capability. This capability can be useful to develop products directly or, at a minimum, to "seed" the market and provide direction and competition for third parties.

Lever 2: Product Technology

(Architecture, Interfaces, Intellectual Property)

Product architecture – including the high-level design of the platform product as well as the interface designs that determine how components or subsystems work together – can have a profound and lasting impact on the structure of an industry as well as the nature of innovation. *Architecture can determine who does what type of innovation as well as how much investment in complements occurs outside the platform leader.* Whether a firm chooses a modular architecture or not (made of easily separable components with open interfaces) is particularly important. Modular designs can reduce the costs of innovation for outside firms and encourage the emergence of specialized companies that may invest heavily and creatively in complementary products and services. We have seen this phenomenon operate in the case of the highly modular personal computer, for example, both with hardware complementors and software complementors.

Intel, Palm, and NTT DoCoMo all took very explicit positions regarding the modular architectures of the platforms they were building. Even the operating systems sold by Microsoft and Cisco, despite their somewhat haphazard evolutions (except for Windows NT/2000), also had modular characteristics that allowed many firms to create complementary products in parallel and independently.

Modular architectures are particularly useful when the interfaces are "open" – that is, when the platform leader specifies publicly how to connect components to its platform. Such open disclosure could facilitate the work of competitors that want to understand at a deep level how a competing product works. We saw Intel, for example, jealously keep the internal architecture of its microprocessors secret, even though it was very open about the specification of interfaces such as the PCI (Peripheral Component Interconnect) bus, or the USB (Universal Serial Bus) interface that linked a computer to its peripherals. Similarly, Microsoft did not give away Windows source code, which would reveal the internal structure of the software platform, even though it revealed detailed specifications on the Windows programming interfaces.

Intel and Microsoft did this over and over again. Intel microprocessors once encountered threats from different architectures, such as ultra-fast reduced instruction set designs used in high-powered workstations, or the superior graphical capabilities of Motorola chips used in Apple Macintosh computers. These products were no longer a major threat to Intel because it evolved its microprocessors to compete more effectively in terms of speed, processing power, graphics, and other dimensions. Microsoft also continually evolved Windows as well as built Windows NT/2000, a high-end operating system with a modern and professional architecture, to compete more effectively with Unix and Linux in the corporate server market. The challenge for Palm and NTT DoCoMo is to keep evolving their platforms.

Cisco's platform is essentially an operating system called IOS (Internetworking Operating System), based on open Internet communications and networking standards – standards that Cisco alone did not define. Therefore, the company had to make its software and hardware products compatible with any new communications technology that emerged, leaving it vulnerable to substitutes competition as well as to specialized niche players. In 2001, it was not clear that Cisco would be able to keep up with all the innovations that happened externally and maintain its historically high growth rates.

We should also note that defining the architecture of a system product could be a powerful way to raise barriers to entry for potential competitors that might consider offering a competing architecture with different interfaces. A potential competitor to Intel, for example, would not only have to invent a microprocessor with a better price/performance ratio; it would also have to rally complementary firms and original equipment manufacturers to change their designs to adapt to this component. This change could incur huge switching costs for the complementary developers. Having a coalition of other firms with their interests aligned with yours through common interfaces is a powerful deterrent for potential entrants. Platform-specific complements can then create, as described in the Microsoft antitrust trial, an "applications barrier to entry" for firms that would like to compete with alternative platforms.

A carefully chosen strategy for intellectual property can also help achieve platform leadership. When a platform leader or wannabe exchanges interface technical specifications with third parties, it stimulates the development of complementary products. Absent this interface information, third parties find it is very difficult — if not impossible — to design complements. Hence, a platform leader or wannabe's decision to make the specification of its interfaces available to the public ("open") lowers third parties' costs to innovate. In contrast, a platform leader or wannabe's decision to withhold the specification ("closed") raises these costs, sometimes dramatically.

In the cases we looked at, Cisco relied primarily on open standards, DoCoMo was pushing for the adoption of W-CDMA (Wideband Code Division Multiple Access), also an open data-transmission standard, and Intel adopted a free and open intellectual property policy on PCI, USB, and AGP (Advanced Graphics Port) interfaces. Palm chose to license its Palm OS to complementors but also competitors such as Handspring. We believe that disclosing information about interfaces is a powerful way to encourage external innovation. But as we said before, disclosing too much can be dangerous: It may facilitate imitation if the information disclosed allows external firms to understand the internal architecture and inner workings of the product.

This observation points to the *fundamental trade-off between secrecy and disclosure* (when one aims to stimulate innovation), and allows us to see the different consequences in terms of innovation of these two very different strategies for dealing with intellectual property. Secrecy protects the incentives of inventors. It guarantees they will reap financial rewards because it reduces the possibility of copying. So, *secrecy is good to block substitute innovation* and to encourage profit-seeking entrepreneurs to innovate on a stand-alone product. Patents are good for temporarily blocking substitute innovation as well. But *disclosure is best suited for encouraging complementary innovation* because it allows the involvement of many inventors, even if they do not have the ability to profit financially from their work.

In short, decisions about product technology – architecture, interfaces, and intellectual property – are critical to platform leadership. Successful firms need to build modular architectures and openly disclose external interfaces necessary to create complementary products and services, while still protecting their competitive advantage. Spending resources on design issues such as platform architecture and interfaces, or on activities to promote industry consensus around interface standards, are not wastes of money. They give the platform leader or wannabe an opportunity to shape the industry and the ecosystem in which it exists.

Lever 3: External Relationships (Consensus and Control, Collaboration and Competition)

To be effective over the long term, platform leaders need to pursue at least two objectives simultaneously. First, they must try to *obtain consensus* among key complementors with regard to the technical specifications and standards that make their platforms work with other products. Second, they must try to *maintain control* over critical design decisions at other firms that affect how well the platform and complements continue to work together through new product generations. Attempting to control what other firms do is usually not a good way to get them to be partners. Consequently, pursuing consensus and control at the same time can often be difficult to achieve.

Our research strongly suggests that consensus among industry players is probably impossible to achieve without at least one firm driving the process that leads to consensus. One firm must also exert some degree of control (such as over interfaces between components and layers between the hardware platform and the software operating system), that is, control over the premises of choice rather than forcing a choice itself. We called this kind of control "ecological control". Platform leaders or wannabes do not care which exact complementary products can be made, as long as they are compatible with the platform. The flip side, of course, also seems true: Control is probably impossible to achieve without some degree of consensus. In other words, industries that have platform products seem to require at least one firm to act as the technical leader to maintain compatible standards, but this kind of leadership is possible only when firms agree to follow a leader.

In the Intel case, the company had to cultivate the technical capability to design interface standards that defined how the microprocessor "talks" to other components. Intel also cultivated the organizational capability to gain the support of other firms to design their own products around these interfaces. But it was difficult for Intel to influence the complete set of PC interfaces because some were not part of the microprocessor itself, though they were part of the PC system. If a critical mass of key players does not agree on interface specifications for the whole product, then "the industry" will not develop *complementary* and *compatible* products, or "the industry" will innovate in these kinds of products only very slowly – perhaps too slowly to match the pace of demand that the platform leader wants.

Specific management processes can help a platform leader achieve consensus and maintain control at the same time. In particular, one key to Intel's success was in the way the company managed relationships with developers of complementary products: Intel engaged in a carefully thought out balancing act of *collaboration and competition* with other firms. To a large extent, this recognized a relationship of mutual dependency. Both Intel's future, and the future of many complements producers, depended on how well these firms cooperated.

The balancing act requires firms to *trust* the platform producer as a partner. But maintaining trust is difficult because the relationships can be ambiguous. It is not always clear if another firm is a supplier, competitor, or complementor, or if a supplier or

complementor today will become a competitor tomorrow. Some firms also play multiple roles. For example, IBM bought lots of Intel microprocessors but it also made its own microprocessors, such as the PowerPC, which competed with Intel products.

Then there is a real threat to complement producers that "dance with the elephant." Platform producers usually do not move into the markets of their partners very often, but they do it frequently enough to make outside firms wary of them as well as anxious to keep innovating in ways that benefit the platform producers. A platform producer may be less likely to intrude into the turf of a complementor if that firm can innovate in ways that the platform producer cannot.

Platform leaders should always try to create "win-win" situations and take on the role of "industry enablers." Enabling the industry to innovate in ever better ways around a platform sometimes requires the platform leader to sacrifice its own private interest, or at least its short-term terms, in favor of the common good. This was Intel's attitude when it invested in interface standards and relinquished royalty rights for technologies that facilitated evolution of the PC as a system. Intel gave away enabling technologies to encourage innovation and competition in complementary markets and coordinated the efforts of hundreds of engineers in developers' forums and what Intel called "plugfest" compliance workshops to make sure that various peripherals and other complementary products worked properly with Intel microprocessors and with each other.

We also believe that platform leaders and wannabes should build reputations as organizations that do not impulsively or carelessly step out of their product boundaries into the territories of their complementors. For example, Intel had a reputation as being careful not to destroy the business models of its complementors (usually). Not all platform leaders had such good reputations, however. Microsoft often crushed complementors that it saw as competitors. Cisco was likely to acquire firms that were key complementors or that posed a threat, if the price was affordable. Palm and its software licensee, Handspring, also had a complex relationship that was likely to become more complex in the future. Handspring was both a complementor to Palm (since its Visor product worked on the Palm OS) and a competitor in the PDA device market, with a rapidly rising share and architectural leadership ambitions, at least in the hardware space. Handspring also had the ability to switch operating system vendors should Palm fall behind in software technology or try to overcharge Handspring for the software platform.

Platform leaders have to find ways to manage these kinds of external tensions. Intel found that the best way to proceed was to adopt a gradual, low-key approach when pushing a particular agenda because this allows input from collaborating firms – and permits both sides to "test the waters." For example, when Intel failed to take a careful approach with its first foray in videoconferencing and tried to impose a new standard while there were strong incumbents such as PictureTel that followed different technical standards, the effort failed. Intel came close to destroying the ability of complementors to make a profit by giving away a cheaper alternative technology. But Intel gradually learned to push an agenda more subtly and successfully with other standards. Intel managers also found it important to assure complementors that critical technical information will remain open and that there will be adequate protection of proprietary intellectual property.

Platform leaders need to establish credibility in technical areas where they want to influence future designs or standards and allow potential complementors to feel

comfortable that the platform leader is acting on behalf of the whole industry or ecosystem, and not just for itself. In drafting interface specifications, for example, Intel did not insist on complete ownership of all related intellectual property. It also encouraged technical input from select firms and only later — once the specifications were almost stable — opened up the process of standards setting to a broader set of firms.

In short, platform leaders have to pursue several roles at once: collaborating with external complementors while championing the public interest, but also competing with external complementors when this proved necessary to stimulate a new complements market. Platform leaders and complementors may be "friend" or "foe," and the ambiguity of many relationships generated tensions and conflicts of interest that both sides had to deal with for the relationship to prosper.

Lever 4: Internal Organization (Structure, Process, Culture)

It is possible for a platform leader or wannabe to use specific organizational approaches to manage external relationships with complementors more effectively. Again, the problem is usually how to compete and collaborate simultaneously. These goals are difficult to balance because some groups within a platform producer might compete with complementors, while other groups might try to get those same complementors to cooperate and adopt the platform's technical standards.

We believe that these internal tensions must, and can, be managed. Intel's solution was to have some groups within the company worry more about competition with other firms, while other groups worried more about cooperation and building consensus with partners. Intel managers acknowledged the necessity to pursue those conflicting goals – which they referred to as "Job 1" versus "Job 2" or "Job 3." Job 1 was to sell more microprocessors, which included encouraging external, demand-enhancing innovation on complementary products. Job 2 referred to competing directly in complementary markets, while Job 3 involved building new businesses that were potentially unrelated to the core microprocessor business.

Intel's top management openly acknowledged that there were conflicts among these goals. Entering complementary markets brought Intel into direct competition with partners, but management sometimes decided that these moves were necessary. Investing heavily in new business development also sometimes distracted groups from focusing on the core businesses, but sometimes this seemed necessary as well to help the firm diversify. In addition to corporate recognition that the company had to pursue multiple goals, platform leaders should communicate clearly to the whole corporation what these goals are, and create a process to acknowledge and resolve the conflicts that may – and are likely to – arise.

Intel chose to clearly communicate its different corporate goals through ubiquitous "Objectives" posters. It also kept internal groups with different goals separate so that outside firms could more easily trust and exchange confidential information with Intel people. Microsoft did something similar, with separations among its operating systems and applications groups that helped it deal with competitors who were also

complementors such as IBM/Lotus, Netscape, Intuit, and Oracle. Cisco also kept its product units relatively independent, which enabled them to work with outside firms that competed with other Cisco groups.

In general, we saw a specific internal division of labor within some platform producers – somewhat like a "Chinese wall" — where different groups play different roles vis-à-vis third parties. But organizational design is usually not enough. Intel people relied heavily on internal processes such as formal planning and off-site meetings and reliance on senior executives to fill an arbiter role in order to resolve conflicts that might arise among the various units within the company. In addition, it seemed important for top management to foster an organizational culture that encouraged debate and tolerated ambiguity, such as permitting different groups to pursue sometimes conflicting goals.

In short, we believe that platform leaders have a better chance of success if they cultivate the organizational capabilities to view the platform as a "system" and have some neutral vision for how to evolve that system and involve external partners in its development. The Intel Architecture Lab embodied these capabilities. NTT also had large R&D resources that it devoted to studying technologies of general utility to the wireless industry. Microsoft, Cisco, and Palm demonstrated a clear recognition as well that their platforms contributed to particular systems, whether it be the PC, Internet-based networks, or personal digital assistants and handheld computers. But it was not clear that these firms had the same ability or willingness to act as neutral industry brokers as did Intel.

Final Thoughts

Even successful platform leaders can fall prey to problems that arise from too much of a platform-centric mentality. There are other ways to compete, such as a niche player maintaining superior quality or service. Not every company can be the platform leader.

Platform leaders can also become so tied to certain technologies that they find it difficult to evolve their platforms. Intel, for example, remained closely tied to the x86 microprocessor family. It was unlikely to move to radically new types of computer architectures. Microsoft continued to have a very "Windows-centric" view of the Internet. It might never take full advantage of the benefits of open standards or the open software movement. Cisco depended heavily its ability weave multiple technologies together through its IOS software. But IOS was a patchwork of code and standards that would someday outlive its usefulness. Palm was quickly becoming a hostage to the internal architecture and the external interfaces that defined the Palm OS. Even NTT DoCoMo had to live with the consequences of its decisions as to what standards to adopt for wireless data transmission and content rendering.

We could also see some platform leaders struggling with platform evolution. For some Intel groups, the platform was becoming the Internet and different kinds of servers and devices that ran Internet software, rather than use Windows and the x86 microprocessor technology. Microsoft was struggling with how to reconcile the Internet as a computing platform to deliver applications ("Web services") versus Windows and traditional "shrink-wrapped" applications. Cisco found itself moving beyond the Internet router as a platform to software that linked various types of networking equipment that communicated through Internet protocols and might someday make basic routers obsolete.

In conclusion, we can say that platform leaders need to have a vision that extends well beyond their current business operations or the technical specifications of one product or one component. They need a vision that says the whole of the ecosystem can be greater than the sum of its parts, if firms work together and follow a leader, and create new futures together. Producers of complementary products need to understand the vision of the platform leaders in their industries, and make some bets on what this vision means for their future. But it is the platform leaders, with the decisions they make and do not make, that probably has the most influence over the degree and kind of innovations that complementary producers create. Platform leadership and complementary innovation by outside firms are not things that happen spontaneously in an industry. Managers in both platform leaders and complements producers make them happen – if they know what to do and how to do it.

Endnotes

[1] Interview with David B. Johnson, director of the Media Architecture Lab, Intel Architecture Lab, on November 11, 1997, at the Intel Architecture Lab, Hillsboro, Oregon.
[2] Interview with Bala S. Cadambi, department manager, Peripherals & Interconnect Technology, Intel Architecture Labs, August 4, 1998, Hillsboro, Oregon. Emphasis added.
[3] This article is based on a forthcoming book: Annabelle Gawer and Michael A. Cusumano, "Platform Leadership: How Intel, Microsoft, and Cisco Drive Industry Innovation," Boston, Harvard Business School Press, 2002. The research consisted of approximately 80 interviews with Intel managers and engineers between 1997 and 2000, as well as various interviews and analyses of publicly available materials on Intel and other companies.
[4] Interview with Dave Ryan, Director of technology marketing, Intel Architecture Lab, Hillsboro, Oregon, August 4, 1998.
[5] Some references we found useful: Michael Katz and Carl Shapiro, "Product Introduction with Network Externalities", *Journal of Industrial Economics* 40(1): 55-83, 1992; M. Cusumano, Y. Mylonadis, and R. Rosenbloom (1992), "Strategic Maneuvering and Mass-Market Dynamics: The Triumph of VHS Over Beta", *Business History Review*, Spring 1992; Richard Langlois, "External Economies and Economic Progress: The Case of the Microcomputer Industry", *Business History Review* 66(1): 1-51, 1992; Richard Langlois and P. L. Robertson, "Networks and Innovation in a Modular System: Lessons from the Microcomputer and Stereo Component Industries", *Research Policy* 21: 297-313, 1992; Carl Shapiro and Hal Varian, "Information Rules: A Strategic Guide to the Network Economy," Boston, Mass.: Harvard Business School Press, 1999; J. Farrell and G. Saloner, "Installed Base and Compatibility: Innovation, Product Preannouncements, and Predation", *American Economic Review* 76(5): 940-955, 1996; J. Farrell and G. Saloner, "Dynamic Competition with Switching Costs", *RAND Journal of Economics* 21(2): 275-292, 1998; J. Farrell, H. K. Monroe, and G. Saloner, "The Vertical Organization of Industry: Systems Competition versus Component Competition," *Journal of Economics and Management Strategy* 7(2): 143-182, 1998; T. Bresnahan, and S. Greenstein, "Technological Competition and the Structure of the Computer Industry", *Journal of Industrial Economics* 47(1): 1-40, 1999.
[6] We use the term "complementors" instead of the longer phrase "developers of complementary products." See Adam Brandenburger and Barry Nalebuff, "Co-opetition: A Revolutionary Mindset that Combines Competition and Cooperation. The Game Theory Strategy That's Changing the Game of Business," New York: Currency Doubleday, 1997.
[7] Carlis Baldwin and Kim Clark, "Design Rules: The Power of Modularity," Cambridge, Mass: MIT Press, 2000), was particularly helpful in thinking about the problem of modularity.