Incorporating Network Effects in Industry-Wide Platform Strategy Making

Jerome Chih-Lung Chou
Hwa-Hsia Technology Institute
Assistant Professor
jerome@cc.hwh.edu.tw

Chia-Liang Hung
National Chi-Nan University
Assistant Professor
clhung@ncnu.edu.tw

Abstract

The logic of platform strategy that contributed to the success of walkman or power tool needs to be reexamined when the product is modularized, interfaces are standardized, and derived products are developed and marketed by a group of technologically and commercially interdependent firms. In such context, direct and indirect network effects affect the success of industry-wide platform strategy. In this article, the authors try to synthesize theories of modularity and network economics to provide a theoretical framework to rejuvenate the platform strategy in the modern network context.

Keywords: modularity, network effects, platform strategy

1. Introduction

Platform strategy is a product strategy whereby the firm uses common modules as the base to develop a stream of derivative products to target multiple market segments efficiently (Meyer & Lehnerd, 1997; Meyer & Seliger, 1998; Meyer & Mugge, 2001; McGrath, 2001). The blueprint of governing common modules is the so-called platform and obtained from system modularization. Controlling an industry-wide platform is taking the position of driving and channeling the industry's innovations, and more and more firms want to become platform leaders (Gawer & Cusumano, 2002). It is obvious that several companies are more successfully providing industry-wide platforms such as Microsoft and Intel. Baldwin & Clark (2000) proposed six modular operators to characterize basic patterns of system modularization, and they interpreted competition among hidden modules by competition in applying substituting, one of the six operators. Modular operators also seem to be a useful conceptual tool for the firm to formulate strategies for developing system architecture, that is, platforms. The purpose of this article is to develop a theoretical framework of successful platform strategy based on modularization actions in the network context. The authors reviewed literature of modularity and adopted a value-creating perspective to view system modularity in the context of value network (Christensen & Rosenbloom, 1995) wherein connections between modules are of both technological and transactional nature. More than enhancing product design efficiency, the platform must be designed to facilitate the realization of desirable transactions and thereby to incorporate the force of network effects into the strategy. By applying positive feedback loops of direct and indirect network effects as a framework, the authors demonstrate that modularization and demodularization actions can be arranged to help the firm seek sustainable industry-wide platform leadership.

2. Modularity and the platform

Modularity is a general system concept. It is a continuum indicating the degree to which a system's components can be taken apart and recombined efficiently. High degree of modularity refers to both the loose coupling among components and more freedom of mixing and matching of components for which the system architecture allowed.

Modularity is indeed a solution for human to conquer complex design tasks that mentioned by one of Nobel Prize winners, Herbert Simon. When the complexity of some part of the system crosses a certain threshold, defining a separate abstraction that has a simple interface would isolate that complexity. That is, that part of the system is eligible to be designed as a module for designers to tackle the difficulties of fixing and configuration. While tackling complexity, the designer usually designs a module as a subsystem for performing a reduced set of functions in the system. Ulrich (1995) defined modular architecture as one-to-one mapping of function and physical element, and the interface must be decoupled. The objective of modularization can be extended to meet engineering and commercial requirements, such as fulfilling needs of certain market segments or reducing production costs, or more. The one-to-one mapping principle enhances commercial design tasks as well as to reduces technological complexity of the system product. Sanchez & Collins (2001) defined modularity with more commercial considerations. They argued that the power of modularity is also applicable when a system is modular that the interfaces between functional elements allow variations of components substituted within the architecture, and the interfaces keep unchanged for inviting innovations during the "commercial lifetime" of the product.

Sanderson & Uzumeri (1995) suggested that the commercial success of Sony Walkman was because it could more easily provide component-based functions, features, and performance levels of products to saturate the market. Meyer & Lehnerd (1997) made similar arguments by analyzing the case of Black & Decker power tool. Earlier, Garud & Kumaraswamy (1995) researched the cost-side benefit of modularity. They proposed that if the organizational design can support modular product design, benefits of substitution could be derived from the reduced performance slippage, amortization of initial design cost over generations of products, and reduced incorporation and searching costs for technology.

In these literatures, modularity is a way to enhance the firm's productivity in design. To reap benefit of modularity in design, the firm must plan its product platforms. The conventional wisdom is that with fine product platforms, the firm can introduce products covering multiple market segments more speedily and economically. Moreover, with the ability of introducing new products, the firm can also learn the market needs more speedily and economically to achieve a competitive advantage. However, nowadays many system products are not designed by a single firm and those platform literatures were before the emerging of broad commercial applications of the Internet. Because of the reduced coordination cost due to modularity enabled by interface standards and the efficient communications enabled by the Internet, a constellation of individuals, firms, alliances, and consortia could collaboratively develop system products with decentralized decisions but on the same rhythm. To reap benefit of modularity in design in such context, the strategic logic of product platforms seems insufficient because the technology decisions for developing derivate products are not necessarily made by the platform module providers centrally. The economies of scope due to reuse of platform modules, the benefit of learning, and the benefit of flexibility are no longer easily realized in such context.

In this article, the authors use the term "module" to refer to a subsystem designed by a specific organization. On a system diagram, a module is a group of design tasks that are densely interrelated within the group that could be done by one organization with only loosely connections to the other parts of design tasks. The connections between modules link organizations not only in technological aspect but also by transactions. As the number and complexity of such connections decreases, the degree of modularity of the system increases. Sometimes, an organization may provide multiple modules for a system product especially when the interrelated design tasks call for a strategic hierarchical control to coordinate.

3. Manipulation of modularity: modularization and demodularization

Baldwin & Clark (2000) have identified six modular operators that characterize modularization. These operators are splitting, substituting, augmenting, excluding, inverting, and porting. Definitions of these operators list in Table 1.

By definition, modularization partitions erstwhile interconnected network of tasks into discrete sub-networks, called modules, which can still function together because they jointly recognize and follow a set of design rules. The settlement of module interfacing rules for various applications becomes the platform.

Table 1. Definitions of Modular Operators

Operator	Definition	Example
Splitting	Separating systems into components which interact across defined interfaces	Interchangeable drives, keyboards, mice, monitors, and printers
Substituting	Switching between components which perform the same function	Replacing a Pentium CPU with a Centrino CPU
Augmenting	Adding a module to increase the functions of a system	Attaching a Web camera
Excluding	Removing a module to reduce the functions the system can perform	Removing a floppy disk drive
Inverting	Making an imbedded function into a stand alone module and setting the module's interfaces	Separating the operating system from DEC's system to create Unix
Porting	Moving a module from one system to another	Using a Mac printer on a PC network by adding a translator

Source: Summarized and revised from Baldwin & Clark (2000)

Modular operators represent patterns of modularization processes. Among them, splitting and inverting will create new interfaces. Substituting does not result in new interfaces, but it improves the system's local performance in a predictable way. Substituting preserves some reused modules, realizing economies of scope. Augmenting and excluding are paired operators to add and subtract modules on the defined interfaces to search ways to create value. The porting operant links two previously separate systems by a translator, redistributing add-on value of modules. Especially when the targeted porting module is also a platform, new opportunities may emerge by interactions of the two platforms.

However, the degree of modularity will not increase forever. If the product performance achieved satisfactory mixing and matching, integrators will aggregate the platform with some other modules to produce complete system products without changing the interface. That is, integrators keep the degree of system modularity but increase the variety of derived system products. In contrast, if the performance is unsatisfactory due to the technological constraints of the interface standard, component providers or integrators may try to integrate some modules by changing existed interfaces, and, somehow, lowering down the degree of modularity, to explore technological and commercial potential of new system products or new components. New platforms may emerge in this way. Some inverting actions integrate previously separate modules to provide more functions, higher performance, or even hiding the information. Schilling (2000) argued that the inter-firm product modularity decreases when component specialization achieves insufficient system product's functionality, or when customers face difficulties in assessing the quality and in assembling of components. In other words, knowing conditions of the network, the platform module provider may integrate strategically among platform's functions to be less dependent on outside components as well as to make the system less modular purposefully. In this way, inverting can also be regarded as an operator of demodularization.

The successful system product evolves usually through module providers' manipulation of modularization and demodularization actions, which are guided by competitive motives in order to construct a favorable value network around the module, and to seek platform leadership in the modular cluster. That requires a strategy, just as a product plan, a platform design, and an industrial ploy.

4. Developing an industry-wide platform strategy

Recent cases have shown that companies providing modules instead of providing complete system products are usually more successful. These more successful companies such as Intel and Microsoft became industry-wide platform providers. They substitute their platforms frequently and make decisions of interface compatibility strategically to keep their leadership in the system evolution. Many but small complementary component designers and some integrators form value networks centered on their platform modules.

The first purpose of launching a platform strategy is to deploy and leverage necessary outside resources. The platform provider must design modular compatibility and transaction mechanisms to encourage complementors to explore the system product's potential by designing various components, or to lower the total cost of the system product by designing cheap components. Surprising functions boosting the market demand may emerge during the exploration, and the lowered price enabled by the lowered total system cost helps the system product diffuse to the mass market quickly.

The second purpose of platform strategy is to establish a modular cluster among complements. When common agreed modular interfaces divided and coordinated innovation, opportunities and risks are emerged and redistributed across modules. Opportunity realization and risk management are value-creating activities driven by module providers' value-seeking incentives that are also the micro foundation for explaining and predicting modularization or demodularization. By splitting the system to set architecture and inverting certain module to set new design rules, the capable system developer can position itself as a platform provider, and share incentives to other designers through the spillover opportunities resulted from the platform's unrealized technological and market potential. Other conforming developers may take these opportunities according to their respective visions, capabilities, and resources.

The third purpose of platform strategy is to manage the self-reinforcing mechanism. In a modular cluster, the platform provider leverages resources of the external network to create value and appropriate rent. Other module providers and integrators also take opportunities by applying operators of substituting, augmenting, porting, and excluding, conforming to the platform module's design rules to form a value network around the platform. Therefore, many transactions of technology among suppliers as well as of end product/service between suppliers and consumers create value aggregately. The platform becomes more valuable as long as some portion of the value created from spillover opportunities can return to the platform, and as more conforming design actions and more transactions take place. Moreover, the adoption of the platform provider's proprietary technology may strategically lock in the conforming designers. Besides, if the demand-side economies of scale of the system product exist, the network of end-users may lock in themselves due to the large utilities of network effect (Katz & Shapiro, 1986). Therefore, the ability of capturing value from the released opportunities back to the platform provider and the lock-in effect among platform adopters are the reasons why an industry-wide platform status may lead to sustainable competitive advantage.

5. Applying modular operators in platform strategy

The conventional wisdom of continuous renewing and cannibalizing the platform or the architecture needs to be broadened (Morris & Ferguson, 1993; Meyer & Lehnerd, 1997; McGrath, 2001). As Table 1 shows, more modular operators than just substituting are available.

Firstly, the platform provider must consider inverting and porting because the two actions determine the platform's compatibility and hence the target market and competitive conditions. The platform provider can continuously substitute the platform in order to make conditions for direct network effects to rise, to improve the platform performance, to keep controlling the interfaces in order to defend away rival platforms, or to create more innovation opportunities for complementors. The platform provider can apply porting to interlink multiple platforms to bring more opportunities because the redefined system scope enables opportunities from multiple platforms to interact. Provision of fine translator module, that is, porting, will reduce complementary design costs, and thereby increase the value of opportunities to complementors.

Secondly, Taking opportunities brought by the growth of the installed base of the platform, complementors or the platform provider can employ augmenting, substituting, and porting to make various complementary components in the following stages. Augmenting adds components of new functions; substituting adds components with the same function but various levels of performance; porting adds components from other systems erstwhile. Hence, basing on a large variety of complementary modules, the system integrator, or the end-user can assemble, integrate, and configure more variety of derived systems to search and meet customer needs. They can also select new components to plug into the system product, or to exclude some components from the system product.

To pursue direct network effects, the platform module provider can invert or augment components with certain functions into the platform. Such functions must enable end-users to exchange or save information of specific formats.

In the network context, modular operators can be applied to help start and reinforce positive-feedback cycles to trigger network effects. Modular operators that can be applied along the system dynamic cycles are summarized in Figure 1.

As shown in Figure 1, if there is an opportunity to make the system product or service with demand-side economies of scale, the platform provider should take this opportunity by altering the platform's function to approximate a complete product with this attribute. In so doing, the platform provider can internalize the positive network externalities. If such opportunity is not obvious or does not exist, the platform upgrades can be planned to pull in resources for complementary innovations to search and expand customer needs, and to reduce the total cost of the system product.

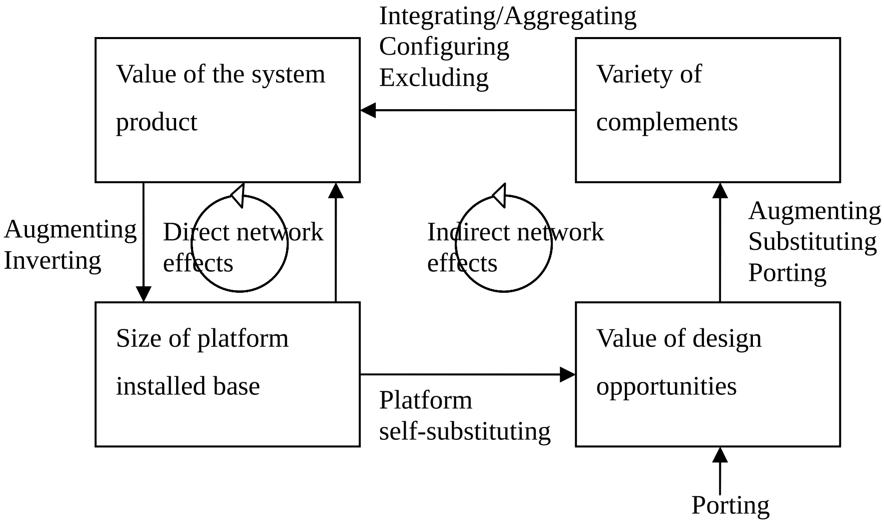

Figure 1. Applications of Modular Operators in Platform Strategy

Source: This Research

While pursuing indirect network effects, in order to channel the direction of complementary innovations, the platform provider can elicit complementors' substituting, porting, and augmenting actions, and elicit integrators' assembling, integrating, excluding, and configuring actions by providing exemplifying actions. On the other hand, in order to signal to complementors that their innovation opportunities are ensured, the platform provider may deliberately not to augment certain complementary modules to encourage complementors. To keep complementors unthreatening to the platform status, the platform provider should foster multiple complementors and confine their contributions to the adoption of the system product relatively small. The development of complementary modules that will significantly boost the market demand or entail the end-user switching cost should be internalized. Complementors in a big amount should be fostered and supported well to sustain the platform's central role. As the variety of complementary design soars, the platform provider should provide tools for integrators to overcome transaction difficulties and fully take advantage of such component variety to roll out various products.

In addition to enlarging the installed base, the platform module provider can interlink the platform with other platforms (that is, porting the platform onto other platforms) to let opportunities from multiple platforms interact and increase the value of spillover opportunities. In the network context, modular operators can be applied to improve contextual conditions for the module providers to seek sustainable platform leadership.

6. Conclusion

Modularity permits system developers to hide technology cmplexity inside modules, and the modular interface enables them to divide innovation tasks and facilites transactions of technologies encapsulated in modules. The substantive modular compatibility supports the formation of transactional relations among system developers and end-users that weave into a value network. In the network context, more successful developers are those who can make their modules common cores needed by many other suppliers, and system products based on their core modules are needed by many end-users. In other words, more successful developers are positioned at centers of value networks, and they provide platforms.

In the modern network context, modularization and demodularization decisions should be made according to the contextual conditions. These conditions can be lined along two causal paths, namely, the positive-feedback cycles of direct and indirect network effects. Platform strategy in the network context can be understood as applying modular operators in conjunction with integrating, assembling, and configuring to adjust these conditions to complete positive-feedback cycles for the platfom provider to create value and appropriate rent.

Complementary modules with demand-side economies of scale should be inverted and integrated into the platform module. Because such modules may boost the market demand and entail high switching cost to end-users, once the collective switching cost gets high, the complementary module provider may gain power over the platform provider and may become a platform competitor. The platform provider should demodularize the system by integrating modules with such potential. While pursuing indirect network effects, the platform provider should confine complementors to contribute to the adoption of the system product only through the increase of component variety. With the authors' framework, product designers and strategy makers can incorporate the force of network effects into their platform strategies.

References

[1]Baldwin, C.Y. and K. B. Clark (2000), *Design Rules Volume I*, The MIT Press, Cambridge, MA.

[2]Chapman, M. R. (2003), *In Search of Stupidity:Over 20 Years of High-Tech Marketing Disasters*, Apress, Berkeley, CA.

[3] Christensen, C. M. and R. S. Rosenbloom (1995), "Explaining Attacker's Advantage: Technological Paradigms, Organizational Dynamics, and the Value Network," *Research Policy* 24: 233-257.

[4]Gandal, N., S. Greenstein, and D. Salant (1999), "Adoptions and Orphans in the Early Microcomputer Market," *Journal of Industrial Economics*, XLVII: 87-105.

[5]Gandal, N., M. Kende, and R. Rob (2000), "The Dynamics of Technological Adoption in Hardware/Software Systems: The Case of Compact Disc Players," *Rand Journal of Economics*, 31: 43-61.

[6]Garud, R. and A. Kumaraswamy (1995), "Technological and Organizational Designs for Realizing Economies of Substitution," *Strategic Management Journal*, 16: 93-109.

[7]Gawer, A. and M. A. Cusumano (2002), *Platform Leadership: How Intel, Microsoft, and Cisco Drive Industry Innovation*, Harvard Business School Press, Boston, MA.

[8]Katz, M. and C. Shapiro (1986), "Technology Adoption in the Presence of Network Externalities," *Journal of Political Economy*, vol. 96(4):822-841.

[9]Katz, M. and C. Shapiro (1994), "Systems Competition and Network Effects," *Journal of Economic Perspectives*, Vol.8, No.2, Spring: 93-115.

[10]McGrath, M. E. (2001), *Product Strategy for High-Technology Companies: Accelerating Your Business to Web Speed*, Second Edition, McGraw-Hill.

[11]Meyer, M. H. and A. P. Lehnerd (1997), *The Power of Product Platforms: Building Value and Cost Leadership*, The Free Press, New York.

[12]Meyer, M. H. and R. Seliger (1998), "Product Platforms in Software Development," *Sloan Management Review*, Fall, 61-74.

[13]Meyer, M. H. and P. C. Mugge (2001), "Make Platform Innovation Drive Enterprise Growth," *Research Technology Management*, Jan.-Feb., 25-39.

[14]Morris, C. R. and C. H. Ferguson (1993), "How Architecture Wins Technology Wars," *Harvard Business Review*, Mar.-Apr.

[15]Rohlfs, J. H. (2001), *Bandwagon Effects in High-Technology Industries*, The MIT Press, Cambridge, MA.

[16]Sanchez, R. and R. P. Collins (2001), "Competing - and learning - in modular markets," *Long range planning*, 34(6), 645 - 667.

[17]Sanderson, S. and M. Uzumeri (1995), "Managing Product Families: The Case of the Sony Walkman," *Research Policy*, 24(5): 761-782.

[18]Shapiro, C. and H. R. Varian (1998), *Information Rules: A Strategic Guide to the Network Economy*, Harvard Business School Press, Boston MA.

[19]Schilling, M. A. (2000), "Toward a General Modular Systems Theory and Its Application to Interfirm Product Modularity," *Academy of Management Review*, 25: 2 (2000): 312-334.

[20]Shy, O. (2001), *The Economics of Network Industries*, Cambridge University Press.

[21]Ulrich, K. T. (1995). "The Role of Product Architecture in the Manufacturing Firm," *Research Policy*, 24: 419-440.

[22]Uzumeri, M. and S. Sanderson (1995), "A Framework for Model and Product Family Competition," *Research Policy*, 24: 583-607.