Mobile Computing: The Next Platform Rivalry

By TIMOTHY BRESNAHAN AND SHANE GREENSTEIN *

* CORRESPONDING AUTHOR: TIM BRESNAHAN, STANFORD UNIVERSITY, STANFORD, CA, 94305. TBRES@STANFORD.EDU..

All modern information and communications technology (ICT) industries use the platform organization. A platform in computing is a reconfigurable base of compatible components on which firms and users build applications. Applications share the general purpose components, which leads to the exploitation of increasing returns at an industry-wide level (Bresnahan and Trajtenberg, 1995).

Platforms compete for developers, who create applications which make the platform valuable for users. Distinct platforms serve different or/or overlapping customers. Platforms also compete in their governance structures, which determine what obligations a developer assumes, and what rights the platform leader reserves for itself. Governance serves a useful function, mediating the terms of transactions and assigning responsibilities to build complements.

We consider governance in mobile computing, specifically. The market involves many high-profile companies, such as Microsoft, Google, Apple, Nokia, and Research in Motion, who employ different approaches to platform governance. That variance frames a seemingly simple question: why doesn't one form of platform governance emerge as superior, dominating most markets in which platforms play an essential role?

Our essay will stress the reasons for differentiation, and we propose an argument that is missing from the platform literature, about changes over time. Platform leaders commit to their approach to governance, but the governance that can help at one moment can get in the way at a later time. That opens up opportunities for differentiated platforms.

I. Options for Platform Governance

Industry figures describe governance as "open" and "proprietary" to different degrees. We eschew these labels, as they come with considerable confusion about their economic meaning. We stress the "hierarchy" of the platform. For simplicity of discussion, we presume a single firm that manages the platform.

At one extreme is an entirely hierarchical platform, with vertical integration of all products. Not just a theoretical ideal, this was used by IBM at the beginning of its period of dominance and by many telephone companies through the period of providing voice services and into the early period of data service provision. In this fully hierarchical governance, all intermediate and final outputs are provided by the platform leader, and all but one firm are merely customers.

Slightly weaker coordination of the general purpose components of a platform arises without vertical integration but with explicit contracts for general components. Once again, not merely a theoretical archetype, this is the governance structure of the Windows platform since 1995. While Windows PCs involve supply of general purpose components by firms other than Microsoft, those firms, such as manufacturers of PCs, contract with a dominant firm who controls the platform.

The Windows platform also uses explicit contracts with software developers who build applications. In this sense the platform mixes vertical integration and contracts. Vertical

integration is present: Microsoft makes Office, for example. Hundreds of firms also work under the contractual obligations and restrictions of SDKs and related developer support systems. The contracts help a developer create value, but encumber them as well.

A related form of governance uses less explicit contracting. It employs implicit contracts between platform sponsors and applications software developers. This is the form most studied in the two-sided market literature (Rysman (2009).) The platform sponsor sets contract features, and these induce an optimal supply of applications products. The most studied problem is the pricing of applications products, but a few papers (Tirole and Weyl, 2010) also take up invention of new applications in response to the reward terms set by the platform sponsor.

This structure was used in the Apple iOS/iTunes store platform. The firm sought to be vertically integrated into all the general purpose technical components (iPhone, iPad, iOS operating system, assorted iWhatnots) and into the exclusive distribution channel for applications and content, the iTunes store.[1] It contracted for content and applications from others with the implicit promise that the core platform would continue to evolve.

The exclusive distribution channel matters. Access to its users raises the value of invention from a game developer or content developer. That gives Apple considerable negotiating leverage with its partners, for example, requiring specific conduct of developers.

One important departure from these models arises in the case of divided technical leadership of the platform. For example, this was observed for a considerable length of time in the PC industry. A number of different firms supplied platform technologies before Microsoft and Intel formed their partnership to act as a single platform sponsor, dividing responsibilities between hardware and software. In that era none of these firms had the ability to block the widespread distribution of platform improvements, not even IBM in the days of the "IBM PC" (Bresnahan and Greenstein (1999).)

In smart phones we observe a limited form of divided technical leadership, in which one firm continues to be the platform sponsor but cannot fully control applications developers. Consider the tensions between Apple and Google over the use of Maps in the iPhone. Apple could contract for use of Map, but could not get Google to be fully cooperative, particularly at intermediate stages of invention involving recombination of multiple inputs. Google had a direct connection to customers, reducing Apple's negotiating leverage.

Why does this matter? Divided technical leadership may interfere with creating and capturing value from improvements in selected general purpose components. A platform dominant firm may seek to use exclusive control over the component to price discriminate, for example, and thus may resist supply by others. Related, a dominant platform firm may view provision of a widely used component as a potential first step to future loss of control of the platform. And that, of course, can make divided technical leadership a very good thing from the perspective of users and developers, as the Apple Maps debacle shows.

Now consider a non-hierarchical platform, the other extreme. In every important historical example, such a platform lacks a profit-oriented platform leader. The Internet Engineering Task Force (IETF) and World Wide Web Consortium (W3C) are the archetypical version of these organizations today. The organizations exert little control over firms. Any and all participants have access to the information that these organizations disseminate, at final and often at intermediate stages. Firms use the technology as they please.

[1] Even this is a mixed control structure. Apple contracts with a large variety of mobile service providers, such as AT&T or Verizon, in much the same way as Microsoft contracts with PC manufacturers. So some general purpose components are coordinated by contract.

The result is, as a Microsoftie put it with a mixture of envy and fear, "a platform that nobody controls and anybody can enhance." This appears to be as far from a hierarchical governance design as feasible.[2]

II. Governance Goals

Governance can succeed or fail along many different dimensions. In this essay we principally focus on two aspects of performance, coordination and exploration.

Since both applications and platform (general purpose) components are needed, a platform industry has a need for coordination. Applications and platforms must work together, or services will not be delivered. It also arises during research and intermediate stages of product development, and during improvement of many different general purpose components.

Incomplete information about the best technologies and/or most important markets calls for exploration. Exploration aims to find improved combination of traits for services or products, prices and costs for operations, and volumes of demand to support both. This activity does involve search, discovery and finding of solutions, but typically not in a controlled environment. It usually takes place in a highly contingent market-oriented setting, involving multiple trials of market activities.

At an industry-wide level a single platform cannot simultaneously achieve both maximal coordination and exploration. Exploration calls for a wide range of initiatives – i.e., many trials with distinct combinations of costs and services. Coordination, by its very nature, calls for some narrowing of that range. The need for coordination makes wide exploration hard to achieve, and vice versa, and that defines a tradeoff.

Two mainstream economic approaches resolve the tradeoff, but they are not perfect. One solution fixes the design of platform components and leaves all the exploration to applications. The other solution invites competition between platforms, each one coordinating platform components with applications, in a (potentially differentiated) standards race. Neither of these works well if there is a single established platform with inertia around a less than optimal technology.

The trade-offs between coordination and exploration emerge in two types of market settings, greenfield and renewal markets. Greenfield markets involve no strong established platform. These are rare. In any industry with ongoing opportunities for innovation – and this includes all ICT industries – the question of renewal of exploration, coordination, and market selection arises more frequently. Renewal becomes possible when new technology, information and learning alter the potential value of different choices.

Generally speaking, renewal sub-goals include backward compatibility and breakthroughs. Achieving backward compatibility is costly, technically challenging in many contexts, and technical restrictive of frontier aspirations. Achieving breakthroughs is also challenging, requiring new combinations of pioneering operations and pioneering designs. Achieving compatibility and breakthroughs at the frontier can and do come into direct conflict.

That conflict can create tradeoffs, which begins to hint at why differentiation may play a role eventually. Changing market needs can create a dynamic tradeoff. For example, the possibility of the need for renewal can increase the value of platforms that support exploration, but at other times coordination may be valuable.

[2] There is some variance within the structure of these groups as well. Several open source organizations have adopted distinct policies for participation and for the decision rights. We leave a full discussion of this variance to aficionados of open source.

III. Benefits and Costs of Hierarchy

In this section we argue that more hierarchical systems increase the degree of coordination, while less hierarchical systems increase the extent of exploration. This tradeoff is sharpened in times of renewal. A hierarchical platform will typically involve intertemporal contracts or commitments, and thus cannot change quickly; changing market needs for coordination vs. exploration will often best be addressed through differentiated platform competition.

Hierarchy and contract help with coordination. This is a familiar argument: hierarchical platforms can coordinate the supply of complements (e.g., applications and general purpose components), raising the returns to services in which users employ both. Intertemporal contracts and commitment support coordination by providing *ex ante* incentives to invest in invention and ensuring that there will not be *ex post* opportunistic problems. When there are thousands of applications developers and a single provider of general purpose components, it is easy to see the point of the "two sided markets" literature, that hierarchical control by the platform provider can provide efficient incentives for such difficult coordination problems as platform balancing.

Why don't those factors always dominate? As already noted, technical or market change over time can create renewal opportunity with strong incentives for a change in firm or in technology, and that will hold irrespective how markets initially select a dominant platform. A new race to establish a new platform (and platform sponsor) may socially optimal and, from the private perspective of the existing dominant firm, problematic but inevitable.

To the familiar list of reasons why an incumbent firm may have difficulty responding to change of this type, we add a few that are idiosyncratic to platform industries with hierarchical leadership. For example, there is an element of inertia around a successful platform. Inertia is great for incumbents as long as it lasts, but the build-up of technological and market opportunity that happens outside a given platform means that a renewal opportunity, when it finally arrives, will have radical efforts by competitors and thus can entail the need for radical innovation in response.

Second, the hierarchical governance structure itself can create difficulties in times of renewal, particularly if there are elements of commitment. Coordination of innovation and product design is costly; a profit maximizing platform sponsor will face limits on how many designs it will support. These limits are sharper if applications developers will sink costs only into platform designs to which the platform sponsor is committed.

Coordinating on only a few or even one options is not problematic when the direction of technical progress is largely known. However, renewal raises risks of coordinating on providing a service that many customers do not want any longer. The incumbent platform sponsor has an incentive to "experiment" with compatible extensions to the existing general purpose components. Yet, there may be considerable uncertainty about the optimal direction, so more experiments may be required. Related, the past coordination of developers to past users' needs may limit the range of available applications.

Exploration in non-hierarchical (market) structures also contains one advantage over hierarchical structures: there are fewer contracts to encumber developers. This can have considerable appeal to developers who want to explore new ways to generate value. This is not a new idea to economists. The idea that hierarchical contracting is always better than markets in reducing transaction costs has been discredited (Williamson, 1975, Gibbons, 2005). We say more below about such frictions below.

As illustration, consider what happened to Research in Motion (RIM) as the renewal opportunity emerged in smart phones. RIM made Blackberry, and it had been the dominant platform firm in

smart phones for business use, and RIM had done a terrific job of coordinating developers around technologies that appealed to enterprise customers. Apple emerged with a platform that generated value in the market place by offering a new kind of smartphone, one that appealed to consumers.

This was a renewal opportunity, with the overall market for smartphones about to grow significantly. Blackberry ultimately fell to the position of being a very small platform in a very large market. That outcome was not inevitable. In the short run, enterprise smartphones were highly differentiated from consumer smartphones, and the tip away from Blackberry was far enough in the future to permit taking up, rather than being swamped by, the new market conditions. Indeed, the firm offered new kinds of phones, new kinds of services, and opened an app store. Users immediately demonstrated interest.

The many differences between a consumer smartphone and an enterprise smartphone framed a challenge for RIM.[3] We do not want to diminish those challenges, but we want to stress that they were not technically insurmountable. Research in Motion did not lack executive talent with extensive experience in the market. No issues about costs of components or shortages of inputs played any significant role in market event. And, as we shall see, *de novo* entrants were able to take them up.

Governance did play a role. Like RIM, The leading mobile platforms in the early part of the millennium – Nokia, Microsoft, Apple – had taken explicit and implicit contractual approaches to managing their partnerships with developers and other business partners. As part of a competitive response, RIM had to change its governance to attract developers of the new, consumer oriented form. RIM's reputation for, and commitment to, supporting enterprise technologies made the move to new consumer technologies hard for RIM and for its cluster of developers. Such a reputation is a recipe for diseconomies of scope between the old, enterprise, business and the new, consumer one; in the language of Bresnahan, Greenstein, Henderson (2012), it is a "necessarily shared asset".

This example also illustrates the general argument, that a renewed round of exploration of platform technologies by a wide variety of potential platform sponsors is undesirable to the existing platform sponsor. Renewal raises the possibility of removal, i.e., competition for the position of platform leader.

IV. Frictions

Which transaction costs are most related to platform governance? We focus on factors that developers regard as frictions.

Uncertainty about the value of new apps may limit the ability of established platform sponsors to contract efficiently over time. Initial platform design may involve technical decisions (the APIs, the SDKs), governance decisions (the boundaries between components and real-time support) and shared assets (e.g., maps, fields for data input-output). If those plans aim in the direction that turns out to have less value to developers, then with commitment it is a difficult march back to square one.

Further, inter-temporal contracting can not necessarily take care of a platform design and governance arrangement that has lost its optimality. One might think that suppliers of complements need to write inter-temporal contracts in order to invest in innovation because, post investment,

[3] The Blackberry was far away from the features of the Apple iPhone in a variety of features: from keyboard to touch keying; from small screen to big color; from minimizing bandwidth to allowing high bandwidth applications; from minimal web presence to high web support; from data-center to cloud-based support; from deep carrier cooperation to tension with carriers; from protected email to data-plan for surfing. See Steve Sinofsky, http://blog.learningbyshipping.com/ 2013/10/03/disruption-and-woulda-coulda-shoulda/.

there could be any of a number of hold up problems, including double marginalization in pricing. If a contract protects platform leaders, developers worry that platform leaders will add new frictions later. Given that renewal opportunities arise when there is fundamental new information, no contract can prevent multiple holdup episodes across multiple rounds of renewal. That interferes with valuable incentives for application developers at early stages.

When there are a very large number of applications, of which only a few may be very valuable, it may be in the platform leader's interest to take control over the value of successful inventions *ex post*. This gives all potential app developers an Arrow problem. If they must contract with platform sponsor to enter, then they have disadvantaged themselves for the later stage at which their invention competes against the incumbent's imitation. This is a serious prospect, especially when the platform leader has the ability to be a "strong second."

Hierarchy has another and related drawback that economics has long stressed: platform sponsors are loath to cannibalize their own services and revenues. Rarely will a platform leader plan to support alternatives to its own services, or contractually allow its complements to threaten those services. As illustration, Apple has made it easy to support its music services, and requires use of iTunes. Android has not supported alternative mapping technologies, and embeds default search at the Google search engine. Neither has precluded users from going to web pages through a browser, but apps use less data and load more quickly, so the defaults have powerful positions in user experience, and developers respond in kind.

What was the role for differentiated platform governance? Apple entered with a user-friendly platform, one that was backward compatible with its music services on iTunes. Their governance choices reserved many rights for Apple, and pushed costs on to developers and content providers. Developers overtly complained on many web sites about the control Apple retained. . No developer can access the iTunes store without agreeing to revenue sharing with Apple. Apple has an approval process for all apps. From Apple's perspective, this is part of a platform-balancing strategy, tilting to the consumer/user with a platform wide promise of quality and safety. From a developer perspective, these are hierarchical frictions.

In contrast to first-mover Apple, Google entered with a very different platform strategy – platform differentiation at work in an entrant's strategy and in competition. The Android platform is, first off, governed in a very non-hierarchical way. Google maintains neither control of information about Android nor control of distribution of apps. The lack of control of information has led to some coordination failures, as different hardware vendors have taken different directions with Android and incompatibilities have arisen. On the other hand, it has allowed a customization of hardware and software by outsiders in a way that has had some tremendous market success (the Amazon Kindle, for example, is an Android device, and is so differentiated in software and distribution as almost to represent a second platform.) While most Android apps are distributed through a Google online store, the platform sponsor allows any application to sell anywhere, speeding time-to-market for developers. More importantly, it leaves Google committed to a strategy of not exploiting developers.

It is notable that the successful second entrant in this vibrant platform race entered not only with different technology, but also with a different platform tilt and with a different governance structures. The elements of platform differentiation are several. ,

Google entered with a purchase rather than its own development in a classic fast-follower strategy, with both defensive and offensive motives. Delay would have provided a longer opportunity to Apple, with the potential to gain such positive feedback that potentially no second entrant could succeed against. Delay also reduced the likelihood for success in competition with

any other new potential entrant, or newly changed platform from among one of several established firms with the resources, such as Nokia, RIM or Microsoft. As it has turned out, events suggest fast entry was as important as the right choices over dimensions of platform differentiation.

IV. Open Questions

The tradeoff between coordination and exploration in platform industries, especially renewed coordination or renewed exploration, turns on the platform governance structure. In the mobile computing arena, we have shown how use of a hierarchical governance structure impeded Blackberry's response to a renewal opportunity, and how a market governance structure provided valuable product differentiation for entrant Android competing against the hierarchical Apple governance structure. Those examples leave open the question of either socially or privately optimal governance structures. The Blackberry example shows how hierarchical control can sow the seeds of later competitive destruction; but renewal opportunities are rare, and the resulting differentiated competition is not costless or quick. The current platform race shows the potential benefits of differentiation in governance structures, but we would speculate that the incentives to differentiate along these lines need not align with those benefits.

REFERENCES

Bresnahan, Timothy and Manuel Trajtenberg, 1995. "General Purpose Technologies: 'Engines of Growth'?" Journal of Econometrics, Special Issue, January 1995, v 65, n 1, pp. 83-108.

Bresnahan, Timothy, Shane Greenstein and Rebecca Henderson 2012 "Schumpeterian competition and diseconomies of scope; illustrations from leading historical firms in computing," in the NBER 50th Anniversary Volume of the Rate and Direction of Technical Change, Josh Lerner and Scott Stern, eds., University of Chicago Press.

Bresnahan, T and S. Greenstein 1999. "Technological Competition and the Structure of the Computer Industry" in Journal of Industrial Economics, March 1999, v. 47n. 1, pp. 1.

Gibbons, Robert. 2005. "Incentives Between Firms (and Within)." Management Science 51: 2-17

Rysman, M. (2009) "The Economics of Two-Sided Markets," *Journal of Economic Perspectives,* 23(3): 125-143.

Shapiro, Carl and Hal R. Varian. Information Rules: A Strategic Guide to the Network Economy. Harvard Business School Press, 1998.

Slivka, Ben, 1995, "The Web is the Next Platform" GX 1016.

Tirole, Jean & Weyl, Glen, 2010. "Materialistic Genius and Market Power: Uncovering the best innovations," IDEI Working Papers 629, Institut d'Économie Industrielle (IDEI), Toulouse.

Williamson, Oliver. 1975. Markets and Hierarchies: Analysis and Antitrust Implications. New York, NY: Free Press.

Last, First, and First A. Last. 2003. "Article Title" In Publication Name, eds. First Last. 335–405. Use the style named References.