Innovation, Openness & Platform Control

Geoffrey Parker
Dartmouth College
gparker@dartmouth.edu

Marshall Van Alstyne
Boston University
mva@bu.edu

07-October-2016

Suppose that a firm in charge of a business ecosystem is a firm in charge of a microeconomy. To achieve the highest growth rate, how open should that economy be? To encourage third-party developers, how long should their intellectual property interests last? We develop a sequential innovation model that addresses the trade-offs inherent in these two decisions: (i) Closing the platform increases the sponsor's ability to charge for access, while opening the platform increases developer ability to build upon it. (ii) The longer third-party developers retain rights to their innovations, the higher the royalties they and the sponsor earn, but the sooner those developers rights expire, the sooner their innovations become a public good upon which other developers can build. Our model allows us to characterize the optimal levels of openness and of intellectual property (IP) duration in a platform ecosystem. We use standard Cobb-Douglas production technologies to derive our results. These findings can inform innovation strategy, choice of organizational form, IP non-compete decisions, and regulation policy.

Keywords: Open Innovation, Sequential Innovation, Platforms, R&D Spillovers, Intellectual Property, Bundling, Two-Sided Networks, Two-Sided Markets, Standard Setting Organizations.

Acknowledgments: This work has benefited from thoughtful comments by Susan Bergman, Marcel Canoy, Thomas Eisenmann, Jovan Grahovic, Josh Lerner, Sadao Nagaoka, Marc Rysman, Jens Prüfer, Tushar Shanker, Ekundayo Shittu, Tim Van Zandt, Patrick Waelbroeck, Thomas Weber, Jason Woodard, and Ken Younge. Xiaoyue Jiang provided key analytic simplifications and Nico Savva provided guidance for our model of technological uncertainty. Seminar participants from Cambridge University, IDEI / Bruegel, Imperial College, Kansai University, Law and Economics Consulting Group, MIT, Stanford Institute for Economic Policy Research, Tilburg Law & Economics Center, University of California-San Diego, University of Minnesota, University of Texas at Dallas, and University of Washington also helped shape this research. The National Science Foundation provided support via grant SES-0925004. We also thank executives from AT&T, Cisco Systems Inc., the member organizations of the International Post Corporation, Microsoft Corporation, SAP, and Thomson Reuters who generously gave of their time to help us understand key tradeoffs in platform business models.

1 Introduction

In 2015, three of the top five public firms by market capitalization used platform business models.[1] Of these firms, Apple nearly went bankrupt because it used closed technology (West et al., 2006), while Google only debuted as a public company in 2004. An open platform business model offers distinct economic advantages because it allows a firm to harness external innovation as a complement to internal innovation (Chesbrough, 2003). While prevalent in information intensive industries such as search (Google), operating systems (Microsoft), and video games (Sony), open platforms have emerged in aerospace (Lockheed Martin), food spices (McCormick), t-shirts (Threadless), 3D printing (MakerBot), and shoes (Nike).[2] Yet managing open innovation creates unique managerial challenges because outsiders respond differently than insiders. Despite considerable research on prices, quantities, incentives, contracts, and network effects, Yoo et al. (2010) note that little formal analysis investigates the ecosystem decisions of business platforms.

We address the gap in the literature by modeling innovation decisions that include a developer ecosystem. Essentially, we propose that a firm in charge of a business platform is a firm in charge of a microeconomy. Like a social planner, it can coax but it cannot coerce third parties into innovation behavior that enhances ecosystem welfare. Unlike a social planner, it favors profits it derives from the platform and it may not know all potential developers who could add value. Interpreting a firm in this way suggests at least three decisions beyond correctly setting prices. First, the firm can choose to give away intellectual property. This helps developers innovate but the firm must determine how much value to give away and how much it will get from developers in return. Second, the firm can choose to absorb developer intellectual property. This goes beyond taxing their sales to taking their ideas in order to disseminate them throughout the ecosystem. Absorbing an innovation reduces its value to the developer who made it but distributing it enhances its value to others. Third, beyond the granting and taking of IP, the firm conditions its optimal choices on

[1] Apple ($633B), Google ($465B), Microsoft ($380B), Exxon Mobile ($344B), Berkshire Hathaway ($330B) [Thomson Reuters data, accessed October 18, 2015]. The platform distinction is based on the presence or absence of a developer ecosystem.

[2] Based on the lead firm's appropriation and redistribution of third party technology, recipes, designs, blueprints, or fitness plans that users developed as complements to products or services provided by the lead firm. For details on the McCormick and Nike platforms see Wacksman and Stutzman (2014).

technical risk and IP reuse. Technical risk affects experimentation while IP reuse affects innovation. Interactions among these factors imply that terms of developer participation have a downstream effect on research and development (R&D) spillovers and these too drive future profits. Finally, a firm can only attract developers if it offers more value than alternatives such as open standards. Developers need not pay royalties or give up their IP in the latter case. We build and explore a formal model to examine these choices.

We analyze the opening of intellectual property as the degree to which the lead firm gives up platform value to third-party developers. On the one hand, these developers can extend the platform's utility to end-users and can create revenue streams that the lead firm can tax. On the other hand, loss of control over open technology sacrifices direct profits and creates the threat of more intense competition. Thus, sharing technology affects not only the innovative capacity of developers, but also the pricing power of the platform sponsor.

In addition, we analyze the optimal duration of developer property rights. Analysis proceeds by considering when a sponsor should absorb developer innovations into the platform and then push these features out to the entire market. There is strong precedent for platform sponsors to appropriate developer innovations, which has the effect of ending their IP exclusivity. Whether through internal development or acquisition, and whether coercively or not, platform sponsors such as Apple, Cisco, Facebook, Google, Intel, MakerBot, Microsoft, and SAP have routinely absorbed innovations developed by ecosystem partners. Cisco, for example, bundles new network features that have appeared among multiple developer products. "Developers don't like it but realize it's good for the ecosystem."[3] Facebook has copied features from multiple developers including Snapchat, Foursquare, and Groupon (Manjoo, 2012) which it then exposes through new application programming interfaces (APIs). Users of MakerBot developed a new design in order to fix a 3D printer flaw that could be printed on the faulty printer itself (Husney, 2014). After testing, MakerBot absorbed and made official the new design. Microsoft has absorbed innovations such as disk defragmentation, encryption, streaming media, and web browsing (Jackson, 1999) then opened APIs to allow access to these new layers. SAP publishes an 18–24 month roadmap alerting

[3] Interview with Guido Jouret CTO Emerging Markets Group, Cisco Systems Inc. 9-8-2006.

developers that they will not face competition or appropriation during this period. After that, any strategic complements may be absorbed to increase functionality of the core platform. On the one hand, a decision not to end developers' property rights can increase their profits, which the sponsor can tax. On the other hand, this prolongs monopoly distortions and prevents valuable new features from becoming community standards. The sponsor thus faces a choice: by extending developers' interests, the sponsor might enjoy higher rents on existing innovations but, by ending developer's interests, the sponsor might increase user adoption and R&D spillovers on future innovations.

A sponsor's decisions about how much to open the platform and when to absorb developers' innovations are critical parts of an ecosystem strategy. These decisions drive adoption and harness developers as an extension of the sponsor's own production function. Netscape founder and Silicon Valley investor Marc Andreessen (2007) describes the increased innovation that platform code sharing can facilitate:

> You can provide an open source ecosystem *within* your platform to let users freely share code with one another... You can, in essence, have your own open source dynamic within your [for profit] platform – in the best case allowing users to clone and modify one another's applications with a level of ease that the software industry has never seen. The rate of rapid evolutionary development that can result from this approach will, I think, be mind-boggling as it plays out.

Though competitors play a role, platform management decisions focus on end-users and third-party developers, who may or may not be known to the sponsor, and who can find it beneficial to participate in the platform. Developers often have ideas the sponsor has not considered and resources that the sponsor does not control. This issue is also described by Andreessen (2007).

> A "platform" is a system that can be programmed and therefore customized by outside developers – users – and in that way, adapted to countless needs and niches that the platform's original developers could not have possibly contemplated, much less had time to accommodate.

To gain access to outside developer resources, many platform sponsors have devised default contracts, enabling what is often referred to as "permissionless innovation" (Cerf, 2012) with appropriate incentives such that even developers not known to the sponsor respond by producing for the sponsor's platform.

As a motivating example, note that contrasting strategies appear to have played a role in the rise of Facebook and the demise of MySpace. In 2005, Facebook membership initially surged when the platform opened from the exclusive '.edu' to the '.com' domain, then it surged again in 2007 upon opening to third-party developers (Piskorski et al., 2012). Specifically, Facebook's strategy focused on creating a robust platform that allowed third-party developers to build new applications. In contrast, MySpace kept all development in-house, a decision that MySpace cofounder DeWolfe later acknowledged was ill-advised at best:

> "We tried to create every feature in the world and said, 'O.K., we can do it, why should we let a third party do it?' " says (MySpace cofounder) DeWolfe. "We should have picked 5 to 10 key features that we totally focused on and let other people innovate on everything else." (Gillette, 2011, p. 57)

The analysis below explores how a sponsor's decisions about how much to open a platform and how long to extend developer property rights move with exogenous factors such as production technology and code reusability. To compare the firm's profit optimization with the public's welfare optimization, we analyze the same choices from the perspective of a social planner – that is, in terms of creating the greatest good for the greatest number. We then extend the model to consider how technological uncertainty affects platform choices. Finally, from the developer perspective, we consider whether developers prefer to join a platform or an open standard. The former taxes and takes their innovations while the latter leaves their innovations unregulated. We conclude by making connections to other literatures, reviewing the theoretical contributions of our model, and outlining the strategic and policy implications of our findings.

To the best of our knowledge, this is the first paper to add a production function to a model that includes a platform, end-users, and developers. The state-of-the-art contributions in the two-sided literature (Parker and Van Alstyne, 2000b, 2005; Rochet and Tirole, 2003; Weyl, 2010) do not consider developer production functions. The most heavily cited papers in the sequential innovation literature – Green and Scotchmer (1995); Chang (1995); Bessen and Maskin (2009) – use only probabilistic innovation. The most heavily cited analytic models of optimal duration of intellectual property (IP) rights, e.g. Gilbert and Shapiro (1990); Klemperer (1990), do not treat sequential innovation or downstream reuse. Our model directly addresses the question of how to manage sequential innovation and by doing so finds, in contrast, that IP duration is *not* arbitrarily long. Our model also responds to the challenge that "although the era of open innovation has begun for many firms, we still lack a clear understanding of the mechanisms, inside and outside of the organization, when and how to fully profit from the concept" (Enkel et al., 2009, p. 312). This literature has, for the most part, avoided formal models presumably because of the complexity of the problem. For example, see the edited volume *Open Innovation* by West et al. (2006).

2 Literature

Before reviewing the literature on sequential innovation, openness, and private ordering, we first highlight papers that define platform ecosystems. Boudreau (2010) defines platforms as the components used in common across a product family. Platform functionality can be extended by third parties and are subject to network effects (Eisenmann et al., 2011; Evans et al., 2006; Parker and Van Alstyne, 2000a,b, 2005). Further, platforms are building blocks serving as a foundation for constructing complementary products and services (Gawer and Cusumano, 2002, 2008; Gawer and Henderson, 2007), or as systems for matching buyers and suppliers who transact with each other using system resources (Hagiu and Wright, 2011) or sales channels (Ceccagnoli et al., 2012). Tee and Woodard (2013) describe the effect of cross-layer interactions on platform governance.

We define a platform business model as an open standard together with a default contract. The standard provides the technological real estate upon which developers build. The contract provides the mechanism that motivates and controls developer behavior. Both are published in the sense that ex ante negotiation is unnecessary and developers need not disclose their identities or ideas before choosing to invest. Default contracts may, however, bind developer behavior as with Twitter's restrictions on in-app advertising or Apple's restrictions on off-platform purchases.

2.1 Sequential & Recombinant Innovation

Our formal analysis is rooted in the sequential innovation and idea recombination literature. The stacking and recombination of ideas can provide increasing returns and innovation spillovers (Weitzman, 1998). Hargadon (2003) argues that innovators rarely come up with completely novel ideas; instead, they recombine old ideas into new ones, adapting them from one context to another. Our model captures this process allowing a given developer to reuse earlier innovations of other developers. Chang (1995) and Green and Scotchmer (1995) find that, to increase innovation, a lead organization (the platform sponsor, in our context) should capture profits from follow-on innovators, and establish longer patents (the duration of intellectual property rights for third-party developers, in our context). Gilbert and Shapiro (1990) and Landes and Posner (2002) examine patent length and breadth as stimuli to innovation. They find that longer but narrower patents are superior to shorter but broader patents. We extend, but modify, the conclusions of this literature by finding that limited duration property rights are often better for both sponsor and developer. Similarly, Partha and David (1994) and Benkler (2002) find that property rights should have short or zero duration. However, we do not find zero duration patents to be optimal. To support our claims, we develop a two-stage model of sequential innovation and add a recursive production function to capture idea recombination. This model allows a firm to control downstream innovation through its choices. The model then gives the firm control over two key constructs: the level of platform openness and the duration of developer property rights.

2.2 Openness

Our understanding of the openness construct is informed by the following literature. A platform is more "open" to the extent that it places fewer restrictions on participation, development, or use across its distinct roles, whether for developer or end-user (Eisenmann et al., 2009). We conceive of complete openness, that is, the absence of control at the platform level – as a fully unrestricted open standard. Another factor distinguishing open from closed systems is the choice of governance model (Laffan, 2011), which we conceive of as the ability to bundle developer innovation (described below) and the decision whether to vertically integrate. Choosing the optimal level of openness is critical for firms that create and maintain platforms (Boudreau, 2010; Chesbrough, 2003; Eisenmann et al., 2009; Gawer and Cusumano, 2002; Gawer and Henderson, 2007; West, 2003). This decision entails a tradeoff between growth and appropriation (West, 2003). Opening a platform can spur growth by harnessing network effects, reducing end user fears of lock-in, and stimulating downstream production. At the same time, opening a platform typically reduces user switching costs, increases forking and competition, and reduces the sponsor's ability to capture rents. Empirical estimates of innovation based on level of openness exhibit an inverted-U shape (Boudreau, 2010; Laursen and Salter, 2005), suggesting that firms can optimize their levels of openness. Our choice to model openness as a continuum follows Valloppillil et al. (1998), Parker and Van Alstyne (2009), and Laffan (2011).

2.3 Private Ordering and Time as Limits on Developer Property Rights

Our second choice construct is the platform contract that controls the duration of developer property rights. The mechanism for such a contract is articulated in the law and economics literature on "private ordering," which is governance via private contract that seeks to achieve welfare gains higher than those provided by a system of public laws (Eisenberg, 1976). Because public laws do not account for information asymmetry and necessitate one-size-fits-all regulation, private ordering can yield better results than uniform law for both sponsor and developer (Williamson, 2002). Remarking on the arbitrariness of U.S. patent durations, Judge Giles Rich, of the U.S. Court of Customs and Patent Appeals, notes that these durations are grounded in social constructs from the time of Paul Revere (Schrage, 1991). In 1790, the term of apprenticeship lasted seven years, leading Congress to offer protection initially for two then later for three apprenticeships before compromising on seventeen years from the date of issue. The current twenty year term from the date of filing reconciles U.S. law with international law. Little evidence suggests that innovation should occur at the same pace in software, hardware, pharmaceuticals, foodstuffs, and sporting goods. Recombination of ideas might also occur at different rates. Contracts that use private ordering can limit developer property rights in order to make ideas available to other developers with the goal of fostering higher rates of innovation. A platform can bundle new features into the platform and expose new APIs. Whereas other studies analyze bundling for its ability to capture rents (Salinger, 1995; Bakos and Brynjolfsson, 1998; McAfee et al., 1989) or provide competitive advantage (Nalebuff, 2004; Eisenmann et al., 2011), our approach focuses on R&D spillovers. Prior literature characterizes R&D spillovers as knowledge externalities that increase the productive capacity of a region (Audretsch and Feldman, 1996) or increase the growth of whole economies (Edwards, 2001). In contrast, a platform-mediated spillover increases the productive capacity of microeconomy partners via a continuous process of innovation absorption and redistribution. Developers can then build on each other as well as on the platform. This term can efficiently adjust to such factors as the size of the developer pool, the production technology, and the extent to which one idea can be reused or recombined with another.

3 The Model

We develop a three-stage model of sequential innovation that includes a platform sponsor, third-party developers, and end consumers. In this economy, developers produce output using platform resources – open code, APIs, and system developer toolkits (SDKs). Conversely, end users consume both the platform and the developers' output.

In our model, the platform sponsor offers a one-time take-it-or-leave-it contract to developers whose only choice is to participate or not. In the first stage, the platform sponsor partially opens its technology, giving away IP, such that developers can innovate upon it. Extending this IP, developers sell to users and share revenues with the platform. At the end of this stage, the platform absorbs all IP extensions. In the second stage, the platform sponsor stimulates cumulative innovation further by giving away all new IP extensions from stage one. Developers again extend the platform and share revenues with the sponsor now for a second time. After revenues are realized in stage two, the sponsor absorbs all cumulative IP extensions and the model ends. For simplicity, all revenues are assumed to be collected at the beginning of each stage; stage one and stage two are assumed to have equal length t. Parameter definitions are listed in Table 1; Figure 1 illustrates model timing.

The central questions for the platform sponsor are (i) what proportion of the platform should the sponsor open to developers, represented by parameter $\sigma \in [0, \infty)$, and (ii) how long should the sponsor let developers keep their new IP, represented as $t \in [0, \infty)$. Note that a platform sponsor might reasonably choose $\sigma > 1$ in cases where the value of subsidizing developers exceeds the cost.[4]

[4] Choosing $\sigma > 1$ would require sponsors to either borrow internally or externally to support the additional subsidy.

Variable	*Paremeter Interpretation*
σ	Share of platform (%) opened to developers
t, δ	Time until exclusionary time expires (discount $\delta = e^{-rt}$)
α	Technology in Cobb Douglas production
F, c	Fixed and marginal costs
k	Coefficient of reuse
N	Numbers of developers
p	Price of individual developer applications $p = v(1 - \delta)$
v	Value, per unit, of developer output
V	Standalone value of sponsor's platform
y_i	Output of developers in stage i and input to developers in stage $i + 1$ with $y_i = k y_{i-1}^{\alpha}$ and $y_0 = S$
ω	Probability of success for a given innovation
π_{ps}	Platform sponsor profit function
π_d	Developer profit function

Table 1: Platform sponsor chooses σ and t (equivalently δ) to maximize π_{ps}

Developers can charge end consumers over one stage of duration t, that is, developers may charge users so long as their extensions are not "open." At the end of each stage, the sponsor absorbs all extensions into the core platform and opens them. Developers can build freely on each others' open code once the stage ends. Platform policy thus parallels a period of patent protection. We represent sequential innovation as two exclusive stages t_1 and t_2. If the platform sponsor chooses $t_1 = \infty$, and does not expire first stage output, then there is no new output in the second stage.

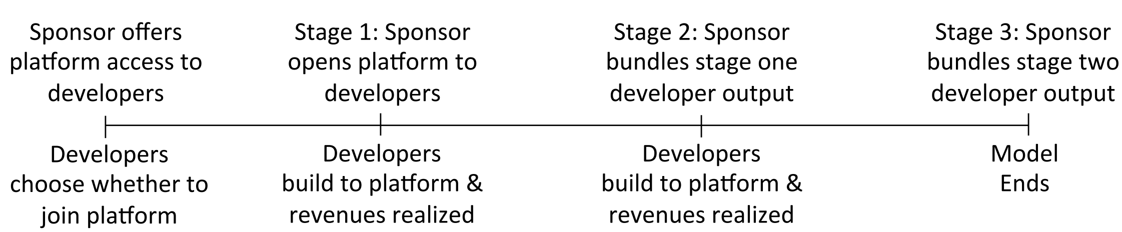

Figure 1: Platform Model Timing

We model a single developer and single end user, then add multiple developers in Section 4. As in Chang (1995), a user has uniform value v for each unit of developer output and a uniform value

V for the platform. The sponsor may sell the platform for V or share σV freely with developers, in which case platform sales fall to $(1-\sigma)V$. Open innovation implies that the sponsor forgoes sales of the "free" resource (West, 2003). In the two-sided literature, σV can be seen as a developer-side subsidy. A developer uses Cobb-Douglas production technology to produce output $y = kx^{\alpha}$. As in standard industrial organization models (Tirole, 1988), k represents real output per unit input and $\alpha \in (0, 1)$ represents diminishing returns technology. Initially, there are no costs of production. We introduce fixed F and variable $cy^{1/\alpha}$ costs in Section 3.2. In the first round, developers use the platform's open resource σV as input. Open code lasts only one round because of technological obsolescence. This prevents developers from reusing free code more than once, which would only strengthen the case for openness.

The output of stage one is $y_1 = k(\sigma V)^{\alpha}$ and that of stage two is $y_2 = k(y_1)^{\alpha} = k^{1+\alpha}(\sigma V)^{\alpha^2}$. Developer innovation is thus recursive. Although the production function stays constant, the effect of reuse rises from k to kk^{α} and the effect of technology changes from $(\sigma V)^{\alpha}$ to $((\sigma V)^{\alpha})^{\alpha}$. Depending on production function specifics – the interplay of reuse and technology – an input subsidy to y_1 has the potential to provide an even larger input subsidy to y_2. By contrast, if the sponsor chooses a closed platform, then developers produce nothing. We explore developer adoption of an open standard, thereby avoiding the platform, as an alternative in Section 4.

The decision to expire developers' IP rights limits their profits. Although consumers value each unit of output at v, they can also wait until a new innovation is bundled into the platform and becomes freely available. This implies that consumers are not willing to pay more than the difference between their maximum willingness to pay today, v, and the present value δv of the good that they will receive at price zero after time t. Thus, platform openness restricts price to $v - p \geq \delta v$, which implies developers may set a maximum price $p = v(1-\delta)$. If the IP rights never expire, developers may charge the monopoly price $p = v(1-0) = v$ but if expiration happens immediately, developers may charge only the competitive price $p = v(1-1) = 0$. To connect discount δ to duration t, note that $\delta = e^{-rt}$. Table 1 in the Appendix summarizes variable definitions.

Platform sponsors share in the innovation profits of third-party developers by imposing a royalty. As in Green and Scotchmer (1995), we simplify this sharing by using the Nash bargaining solution, giving each party 50%.[5] Summarizing, platform sponsor and developer profits are written as:

$$\pi_{ps} = V - \sigma V + \frac{1}{2}py_1 + \delta\frac{1}{2}py_2 \quad (1)$$

$$\pi_d = \frac{1}{2}py_1 + \delta\frac{1}{2}py_2, \quad (2)$$

where $\pi_d = \pi_{d1} + \delta\pi_{d2}$. Equation 1 says that platform profits are the sum of platform sales, first stage royalties, and discounted second stage royalties net of subsidy costs. Expressing platform sponsor profit in terms of model primitives yields

$$\pi_{ps} = V(1-\sigma) + \frac{1}{2}v(1-\delta)k(\sigma V)^{\alpha} + \delta\frac{1}{2}v(1-\delta)k^{1+\alpha}(\sigma V)^{\alpha^2}. \quad (3)$$

In contrast to prior literature, model innovations here include recursive production and resource spillovers. Section 4 explores the choice of organizational form.

3.1 Platform Sponsor Decisions

The platform sponsor faces two central tensions. First, closing the platform increases the sponsor's ability to charge for access while opening the platform increases developer ability to innovate. Second, the longer developers retain rights to their innovations, the higher the royalties they and the sponsor earn. In contrast, the sooner developers' rights expire, the sooner their innovations become a public good upon which other third party developers can build. The optimal contract is thus a pair $\langle\sigma, \delta\rangle$ (isomorphic to $\langle\sigma, t\rangle$) where choice parameter t represents the duration of exclusive control and choice parameter σ represents the level of openness. As we shall see, production in each stage, discount rate, code reuse, and value added by developers all govern a platform sponsor's choices.

[5] As of this writing, Amazon, Apple, Facebook, Microsoft and SalesForce charge 30%. Silicon Valley venture capitalist Bill Gurley identifies platform fees ranging from 1.9% to 70%. This suggests a 50% split is a reasonable approximation (see http://abovethecrowd.com/2013/04/18/a-rake-too-far-optimal-platformpricing-strategy/). Comparative statics are robust to choice of royalty.

3.1.1 Platform Sponsor Choice of δ and σ

Proposition 1 *Platform profits π_{ps} are well behaved and there exists a unique pair $\langle\sigma^*, \delta^*\rangle$ that maximizes π_{ps}.*

The optimal length of exclusionary stage δ^* has an interior solution and a corner solution, both governed by the ratio of first to second stage output. The two cases are then:

$$\delta^* = \begin{cases} \frac{1}{2}\left(1 - \frac{y_1}{y_2}\right) & \text{when } y_2 > y_1, \\ \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

There are three key results: (i) the interior solution occurs when second stage output exceeds first stage output, (ii) the condition for a finite developer property rights period is that first stage output must exceed the developer subsidy, and (iii) it is never profit maximizing to force immediate openness on developer applications. The boundary condition between an interior and corner solution for δ^* occurs at $y_1 = y_2$.

The optimal solution σ^* can be found implicitly when δ^* has an interior solution. σ^* has a closed form solution in the case of a corner solution where δ^*=0. The two cases are:

$$\sigma^* = \begin{cases} \frac{1}{V}(\eta_1\pi_{d1} + \delta\eta_2\pi_{d2}) & \text{when } \delta^* > 0, \\ \\ \frac{(\alpha v k/2)^{1/(1-\alpha)}}{V} & \text{when } \delta^* = 0. \end{cases} \tag{5}$$

Proof. *See Appendix.* ■

The results for δ^* have useful implications. If first stage output exceeds second stage output, the sponsor optimizes δ on $(1-\delta)y_1$, which binds at corner solution $\delta = 0$. If second stage output exceeds first stage output, the sponsor is optimizing on $\delta(1-\delta)y_2$ which solves to an interior solution nearer $\delta = \frac{1}{2}$. Intuition follows from either the sponsor's profit equation 1 or from the optimal discount equation 4. If first stage output matters more, the sponsor prefers near term royalties and lets developers raise prices so t rises to infinity. On the other hand, if second stage output matters more, the sponsor wants to reach stage two sooner yet still relies on developer contributions to get there, so t is finite.

This proposition provides what is, in effect, a choice of property rights period analogous to an industry specific patent, after which a sponsor can absorb innovations into the corpus of open innovation resources. In exchange for access to the platform and royalties on sales, the platform sponsor grants to developers a short term monopoly on their innovations.[6] Independent of the duration of protection that patent or copyright law might provide, a platform firm could then choose terms that adapt to the productivity conditions of its microeconomy.

Now consider the results for platform openness σ^*. First, it is important to note that σ^* can be greater than 1. The subsidy offered to developers can exceed the current value of the platform when the discounted future value is sufficiently great (Noe and Parker, 2005). Then the firm must finance investment through borrowing or venture capital. This can be observed in practice, especially for early stage platforms mobilizing their ecosystems. Twitter and Facebook, for example, both lost money before their initial public offerings, requiring millions of dollars of venture capital (Hof, 2013) and heavy spending to create developer economies. In 2014, Uber, with only hundreds of millions in revenue, received $1.2 billion in venture funding (Rusli and MacMillan, 2014) in part to finance the buildout of a developer ecosystem (Lawler, 2014).

When δ has an interior solution, we can express the relationship between openness σ and the elasticity of developer output in each stage as follows. Note that $\eta_i = \frac{\partial y_i}{\partial \sigma} \frac{\sigma}{y_i}$, $i = 1, 2$ and developer profit in each stage is denoted π_{di}. To see this, multiply both sides of the first case of Eqn 5 by V to get

$$\sigma V = (\eta_1 \pi_{d1} + \delta \eta_2 \pi_{d2}). \tag{6}$$

Intuitively, when the platform sponsor opens its core platform resources to third parties, the gain from sharing in developer profits must offset platform losses (forgone revenue σV). The elasticity term governs how sensitive developer output is to the level of platform openness, so that the optimal level of σ properly balances revenues lost and gained.

[6] In practice, the 18–24 month window used by Cisco and SAP seem much more realistic for technology and software than the 20 year period granted under U.S. patent law.

3.1.2 Effect of Model Primitives on Platform Sponsor Choices

The optimal proprietary period and optimal openness depend on industry specific factors inclusive of the platform's core value, developer unit value, and reuse technology. We summarize our findings in Corollary 1. Time t moves in the opposite direction from discount coefficient $\delta = e^{-rt}$.

Corollary 1 *Comparative Statics – Table 2 summarizes effects of model primitives on platform sponsor choices of optimal contract.*

Comparative Statics	σ^*	t^*
Platform value: V	-	0
Developer value: v	+	+
Reuse coefficient: k	+	0

Table 2: Openness rises in v and k but falls in V. Duration t rises in v.

Proof. *Derivations appear in the Appendix.* ■

Rising platform value V implies reducing platform openness ($\frac{\partial \sigma^*}{\partial V} < 0$). Equation 6 shows this directly for σ^* since V only appears as part of σV. A more valuable initial platform means that less of its value needs to be sacrificed to stimulate developer production. The initial value of the platform is unrelated to the duration of developer property rights until the sponsor absorbs innovations ($\frac{\partial t^*}{\partial V} = 0$), a reasonable assertion as V and v are not otherwise related.

In contrast, increasing the developer value, v, per unit produced has the effect of increasing the sponsor's willingness to open the platform. The sponsor rationally sacrifices direct platform profits in order to share in more valuable developer extensions. Likewise, an increase in the value of developer output leads a platform sponsor to offer developers a longer property rights duration t^*. Increased developer value in both stages has the effect of making the sponsor more patient, and more willing to postpone absorbing new features into the platform. The Atari 2600 provides a suggestive example of a platform that was too open relative to developer value-add. An initial wave of success followed from its high value joystick innovation but Atari did not lock out or quality review subsequent extensions. Advertisers such as Fox, CBS, Quaker Oats, and Chuck Wagon dog food then launched a large number of poorly executed titles. This significant reduction in value-add sparked the industry "crash of 1983" Kent (2001). After that, the Atari platform quickly became obsolete, such that development since that time has been limited to the tinkering of nostalgic hobbyists. Long-term success can be linked to a sponsor's ability to adjust the openness of its platform to developer value-add.

The successful F-16 military aircraft platform, now in its 40th year with over 4,500 aircraft produced has seen decades of technological innovation in mechanical and electronic design. The platform is now more open than it was in the 1970s when General Dynamics was the sponsor. Teece (1988) observed that "The trend in fighter plane subsystem costs has been away from air vehicle and propulsion and toward avionics, and this trend is likely to continue." Given the relatively larger fraction of value in add-ons to the airframe/propulsion platform, we conjecture that the current sponsor, Lockheed Martin, might profit from inviting more firms to take larger roles in upgrading and extending the F-16 while maintaining rights to critical complements to maintain platform control and the ability to share in external innovator profits.

Our model also predicts that higher developer values imply a longer intellectual property rights period for developers. We observe this in practice with SAP, which agreed to longer exclusivity for ADP, a major payroll processing player, in order to attract ADP to the SAP platform as it transitions from on-premise installations to a cloud-based solution.[7]

Reuse coefficient k has a different effect. As platform resources become more reusable, developer production increases. This dynamic implies that the sponsor should open the platform more but, surprisingly, does not alter the duration of the exclusionary period. In terms of openness, higher reuse implies higher value per unit of openness, leading the sponsor to open the platform more. For example, software tends to be more reusable than hardware, and tends to be given away more freely. In terms of developers' property rights duration, however, the effect of rising reusability is negligible. Given the same production technology, reusability increases developer output at the same rate in both stages such that, after discounting, the sponsor has no reason to favor first- or second-stage output. If technology *changed* between stages, better technology might correspond with shorter intellectual property rights protection.

[7] Interview with Thomas Spandl, SAP Vice President of Ecosystems, July 18 2011.

3.2 Welfare

We extend the model to include developer fixed costs F in each stage and increasing marginal costs, which we model as $c\,y^{1/\alpha}$. To avoid introducing an additional parameter, this formulation uses the same technology parameter α as in the production function. In the cost function $c\,y^{1/\alpha}$, $\alpha \in (0,1)$ serves to model convex increasing costs.[8] For simplicity, marginal cost remains small enough that $vy_2 \geq \frac{c}{\alpha} y_2^{1/\alpha}$. We continue to assume a convex region of interest, defined by a negative semidefinite matrix with respect to openness and time. These additions allow us to compare the choices for a welfare optimum from a social planner's point of view against those of a sponsor's maximum net profit. Adding fixed and marginal costs to Equation 1 provides the basis for comparison.

$$\pi^c_{ps} = (1-\sigma)V + \frac{1}{2}\left(py_1 - cy_1^{1/\alpha} - F\right) + \frac{\delta}{2}\left(py_2 - cy_2^{1/\alpha} - F\right) \tag{7}$$

Including consumer surplus, the following welfare equation then determines the social planner's optimization.

$$\arg\max_{\sigma,\delta} W = V + (vy_1 - cy_1^{1/\alpha} - F) + \delta(vy_2 - cy_2^{1/\alpha} - F) \tag{8}$$

This is subject to a developer participation constraint as follows:

$$\pi^c_d = \frac{1}{2}\left(py_1 - cy_1^{1/\alpha} - F\right) + \frac{\delta}{2}\left(py_2 - cy_2^{1/\alpha} - F\right) \geq 0. \tag{9}$$

A positive price, $p = v(1-\delta) > 0$, represents a wealth transfer from consumers, while the extent of platform openness σV represents a wealth transfer from the platform sponsor. Both are irrelevant to a social planner except to the degree that developers must cover development costs. Note that in the absence of costs, a social planner simply allocates all existing resources for innovation without delay and chooses $\langle \sigma_c^\dagger, t_c^\dagger \rangle = \langle 1, 0 \rangle$.

[8] This formulation includes the standard quadratic form cy^2 as a special case (i.e. $\alpha = \frac{1}{2}$) but allows cost to fall with improved technology. In this way, increasing (decreasing) α serves both to increase (decrease) output and reduce (increase) costs.

Proposition 2 *The social optimum is a contract* $\langle \sigma_c^{\dagger}, t_c^{\dagger} \rangle$ *with* $\sigma_c^{\dagger} > \sigma_c^{*}$ *and* $t_c^{\dagger} < t_c^{*}$ *. The social planner prefers a more open platform and a shorter proprietary duration* ($\delta_c^{\dagger} > \delta_c^{*}$) *for developer innovations than do platform sponsors.*

Proof. *See Appendix.* ■

We observe that the greater the share of downstream innovation captured by the platform sponsor, the greater is the incentive to open the platform. This finding parallels results elsewhere in the literature: internalizing downstream innovation causes the owner of an upstream innovation to behave more like a social planner. Interestingly, the proof shows that the converse is also true. Higher costs cause the social planner to behave more like a private firm.

3.3 Technological Uncertainty

Because innovation can involve risk, we analyze whether technological uncertainty influences the choice of platform openness and the duration of developer intellectual property rights time before bundling. Let the probability of technical success be given by ω. Further, to balance risk and reward, allow output from riskier innovations to rise conditional on their success. Then, first-stage production is given by the random variable

$$Y_1 = \begin{cases} \frac{k}{\omega}(\sigma V)^{\alpha} & \text{with probability } \omega, \\ \\ 0 & \text{with probability } 1-\omega. \end{cases} \tag{10}$$

This formulation assumes that in industries where technical success is difficult, i.e. ω is low, such success is highly rewarded.

Expected first-round innovation is given by $\mathbf{E}(Y_1) = k(\sigma V)^{\alpha}$ and variance is given by $Var(Y_1) = \left(\frac{1-\omega}{\omega}\right) k^2(\sigma V)^{2\alpha}$. Although the expected value of production is independent of technical risk, the variance of production increases with decreasing probability of technical success (Singh and Fleming, 2010). In the limit, as $\omega \to 1$, we revert to the original model with zero variance.

Similarly, provided that first stage innovation was technically successful, second-stage production is given by the random variable

$$Y_2 \mid \text{success in stage } 1 = \begin{cases} \frac{k}{\omega}(y_1)^{\alpha} & \text{with probability } \omega, \\ \\ 0 & \text{with probability } 1-\omega. \end{cases}$$

The unconditional, time zero, production in the second stage is given by:

$$Y_2 = \begin{cases} (\frac{k}{\omega})^{\alpha+1}(\sigma V)^{\alpha^2} & \text{with probability } \omega^2, \\ \\ 0 & \text{with probability } 1-\omega^2. \end{cases} \tag{11}$$

The unconditional expected value of second-stage production at time zero is $\mathbf{E}(Y_2) = \omega^{1-\alpha}k^{1+\alpha}(\sigma V)^{\alpha^2}$ with variance $Var(Y_2) = \left(\frac{1-\omega^2}{\omega^{2\alpha}}\right)k^{2+2\alpha}(\sigma V)^{2\alpha^2}$. Again, as $\omega \to 1$, we retrieve the original model with zero variance. Because $0 \leq \alpha \leq 1$, the value of the second-stage production is increasing in the likelihood of technical success ω and therefore decreasing in variance. Low likelihood of technical success (i.e. low ω) does not negatively affect the value of first-stage innovation because innovation is more valuable if it is difficult to achieve, but it does negatively affect the value of second stage innovation because, for a second stage to exist, the first stage must be successful.

With these definitions, the platform sponsor profit function becomes:

$$\mathbf{E}(\pi_{ps}) = V(1-\sigma) + \frac{1}{2}v(1-\delta)k(\sigma V)^{\alpha} + \delta\frac{1}{2}v(1-\delta)k^{1+a}(\sigma V)^{\alpha^2}\omega^{1-\alpha}. \tag{12}$$

Propositions 1 continues to hold but with y_1 and y_2 replaced by $\mathbf{E}(Y_1)$ and $\mathbf{E}(Y_2)$. We summarize these implications in the following result.

Proposition 3 *Holding all else constant, a higher likelihood of technological success increases platform openness and innovation, and decreases the time until the platform sponsor expires developer property rights. Increasing ω implies that σ^* and $\mathbf{E}(Y_2)$ rise, while t^* falls.*

Proof. *See Appendix.* ■

We can also conclude that a lower likelihood of technical success (i.e. decreased ω) decreases the optimal choice of how much to open the platform. Because subsequent innovation entails more risk, the sponsor prefers to collect royalties t^* longer rather than gamble on innovation from bundling sooner.

4 Open Standards – Cooperation in the Absence of Control

So far, our analysis has assumed an "open" platform in the sense of access yet it remains "closed" in the sense that a sponsor controls the IP contract. Developers cannot independently choose IP duration for their contributions. Is this the best way to organize for innovation? Might not participating in an open and unsponsored standard allow developers to choose their own durations, increase their individual profits, and thus grow industry profits? Here, we examine alternate organizational forms such as a developer's decision to cooperate with other developers rather than accede to the platform sponsor's terms. We find that a control mechanism that coordinates spillovers can raise welfare even for developers whose proprietary innovations become public. Past a threshold proportional to the size of the developer pool, increased profit in the second period exceeds lost profit in the first period.

Consider a two-stage game where two developers choose simultaneously in a given stage. For clarity, we illustrate with two players before proving the N player case. Each players knows the other's payoffs, but building on the unsponsored standard does not require them open their code. They may voluntarily cooperate or defect with the former interpreted as opening their code at the end of the stage in order to grow the pool, and the latter interpreted as keeping code closed at the end of the stage in order to charge full price. Figure 2 provides the extensive form of this game. Because each player moves simultaneously at each stage, there are five information sets for each player. In the figure, without loss of generality, we represent this by having Player 1 move and then Player 2 move without observing Player 1's action. Player 1's information sets are denoted I_1^j; Player 2's information sets are I_2^j. A complete strategy for each player lists their action at each information set. For example, a strategy for Player 1 is a quintuple, $s_1 = (s_1^1, s_1^{CC}, s_1^{CD}, s_1^{DC}, s_1^{DD})$.[9] Because each player has two possible actions (cooperate or defect) at each information set, each player has 2^5 possible strategies. Thus there are $32 \times 32 = 1024$ strategy pairs.[10] A pure strategy for player i can be written as $S_i = \{(s_i^1, s_i^{CC}, s_i^{CD}, s_i^{DC}, s_i^{DD}) : s \in C, D\}$.

[9] s_1^{CC} is the action that Player 1 chooses given that both players cooperated in stage 1. Similarly, s_1^{CD} is the action that Player 1 chooses given that Player 1 cooperated and Player 2 defected in stage 1. s_1^{DC}, s_1^{DD} have analogous meanings.

[10] See, e.g., Tadelis (2013) page 179.

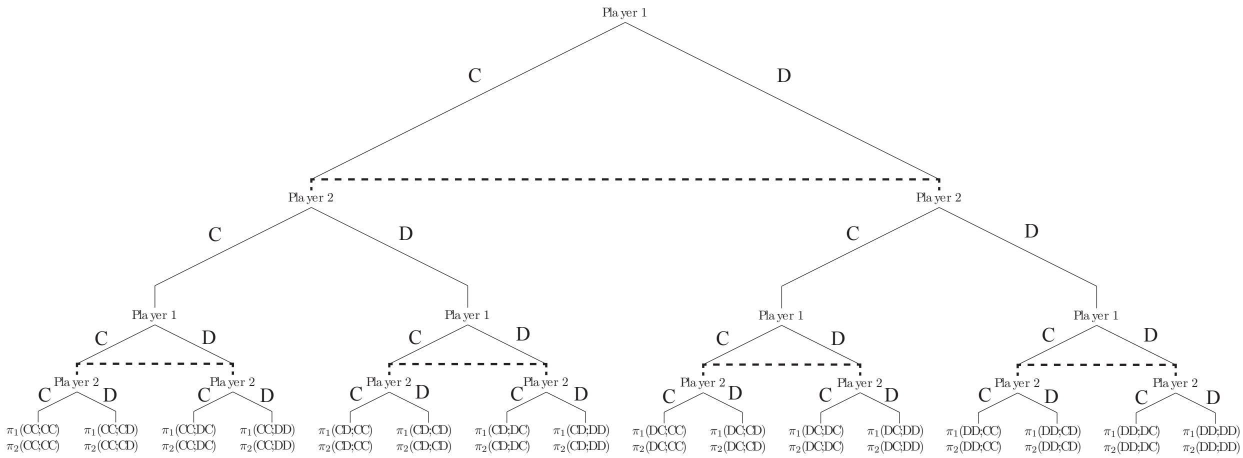

Figure 2: Two stage Prisoner's Dilemma. Actions are denoted $a_1(1)a_2(1); a_1(2)a_2(2)$ where $a_i(t)$ is player $i's$ action in stage t. E.g., DD;CD means that Player 1 and Player 2 defected in stage 1; Player 1 cooperated in stage two and Player 2 defected in stage two. Table **??** enumerates all payoffs $\pi_1(a_1(1)a_2(1); a_1(2)a_2(2))$ and $\pi_2(a_1(1)a_2(1); a_1(2)a_2(2))$.

We list the associated payoffs for each set of actions in Table 3 of the Appendix. Note that defecting in stage one benefits the defecting player in stage one by raising his price but then harms the other player in stage two by denying her access to a larger code base upon which to build new innovations. As explained in Section 3, the effect of cooperating (opening) in stage one is to limit price to the standard $p = (1 - \delta)\, v$ because the innovation becomes freely available. By contrast, the effect of defecting (closing) in stage one is to increase price to $p = v$ because the innovation is not otherwise freely available.

If the other player cooperated in the first stage, then there is extra open code in the second stage, which rises by a factor of 2^α. This can be seen by substituting both players' output $2y_1$ into the equation for y_2, which increases from $y_2 = k(y_1)^\alpha$ to $k(2y_1)^\alpha = 2^\alpha k(y_1)^\alpha$. By contrast, if the other player defected in stage one, then a developer can only build on her own code, which limits her second stage output to $k(y_1)^\alpha$.

This analysis extends naturally to the case of $N \geq 2$ developers. If a group of $n \leq N$ players cooperate in the first stage, then each player has access to extra open code, which increases by a factor of n^α. Substituting the n players' collective output into the case where $n = 2$ above, second period output y_2 rises to $n^\alpha k(y_1)^\alpha$. Cooperation at one stage increase open code available for innovation at the next stage. Interestingly, coalitions are unsustainable in any subgame without control. For the two-stage game, denote $n_i \in [0, N]$ as the number of developers who cooperate in stage $i \in \{1, 2\}$. Then, there is only one unique pure strategy equilibrium.

Proposition 4 *The unique pure strategy equilibrium is $n_i^* = 0$ for $i \in \{1, 2\}$. All players defect in both stages. Thus profit seeking developers close their code.*

Proof. *See Appendix.* ∎

When positive spillovers exist, self-interest of profit motivated developers leads them into an intertemporal prisoner's dilemma. Each prefers to close his code, even as he prefers that other developers open theirs. This implies that fully open standards are not socially optimal when developers can build on each others' collective output. This leads us to ask when a developer would prefer to submit to a contract that enforces cooperation. As set forth in Proposition 5, a straightforward solution finds the indifference point between the cooperative and the non-cooperative payoffs.

Proposition 5 *If a finite proprietary period $t < \infty$ maximizes profits, then N players will prefer a contract that forces cooperation when the number of developers exceeds a threshold bounded by:*

$$N \geq 2^{1/\alpha}.$$

The benefit of the cooperative solution relative to the non-cooperative solution also rises in the size of the developer pool. Then, as technology α improves, the group size threshold shrinks and the advantage of enforcing cooperation grows.

Proof. *See Appendix.* ∎

The implication is that every player would prefer to cooperate when the developer pool is sufficiently large. In the absence of a forcing contract, each player viewing her own choices privately and independently chooses to defect. But, each player, seeing the benefits of cooperation, prefers to enter into a contract forcing everyone to open his or her code at the end of the stage. The platform sponsor's optimal contract, which seems harsh in the one stage game, is helpful in the two stage game because broader reuse becomes valuable under recursive production. Interestingly, as technology improves, the threshold for cooperation falls, spillovers from cooperation rise, and value increases in group size. In the face of positive externalities, ecosystem governance becomes valuable and necessary.

A standard setting organization (SSO), analogous to a platform sponsor, might be able to achieve a similar effect. Proposition 5 shows that if innovation is cumulative, developers benefit substantially from recursive R&D spillovers. These benefits accrue *only* if the SSO binds developers to give up their first stage innovation in order that the ecosystem benefits from second stage innovation.[11]

If this binding contract occurs, then the SSO effectively behaves like a platform sponsor – each helps resolve a classic "collective action" problem (Baldwin and Woodard, 2009). In the absence of orchestrated governance, individual incentives to profit maximize lead to Pareto inferior welfare in terms of innovation and profits. Thus, some form of governance is necessary to effect control. As the comparative statics of Corollary 1 show, the optimal timing of property rights can also depend on industry specific factors such as v. If this is true, then an industry sponsor (or SSO) can craft more specific timing than a legislative regulator whose rules apply across industries. Relative to open standards and regulation, efficiency gains from platform sponsorship might therefore occur in coordination and in technology specificity. This efficiency allows innovation to adjust to the varying "clockspeeds" of different industries.

The sponsor's interest in efficient innovation has interesting real world application as a resolution to the problem of the "anticommons," identified as the hold-up that occurs when too many distinct parties can each block downstream innovation because each has a conflicting yet interlocking property right (Heller and Eisenberg, 1998). Under a platform model, the platform sponsor unblocks later innovation by making earlier innovation available to all developers on a non-discriminatory basis. The sponsor uses its property right in the platform to grant access to developers conditional on securing the ability to absorb enhancements into future versions of the platform. Proposition 5 shows that far from encouraging developers to avoid the platform, expiring their property rights can make them better off over multiple cycles of innovation. In the 1990s, more desktop OS developers

[11] We thank Jason Woodard for the insight that governing spillovers can apply also to SSOs.

added to Windows than to Linux (Jackson, 1999). In the 2000s, more mobile OS developers added to Android than to Ubuntu (?). Ease of development and expanded opportunity offer reasons why developers might have preferred the more controlled platform despite the higher risk of their innovations being absorbed by the platform sponsor. More control is not always better – indeed the model demonstrates interior solutions. Rather, a sponsor's self-interest in platform innovation can motivate it to shepherd the platform much as if it were a social planner. R&D spillovers are not simply an accident of proximity (Audretsch and Feldman, 1996; Edwards, 2001) but a controlled optimization of appropriation and dissemination that benefits the community.

5 Implications & Extensions

5.1 Managerial & Policy Implications

The preceding analysis offers a variety of managerial insights. First, it shows that opening the platform, i.e. giving away IP, is in fact profitable. A classic managerial mistake of Apple, MySpace, and others has been to hoard IP or to charge for it at levels that kept their respective ecosystems small. Our optimal openness result shows that a strategy of externalizing IP, and participating in royalties from open innovation, grows profits via an ecosystem as distinct from a strategy that emphasizes direct platform sales. Opening platform IP highlights the call for understanding the "inside-out" form of open innovation in which a firm "multiplies technology by transferring ideas to the outside environment" (Enkel et al., 2009, p. 312). Optimal openness rises with developer value-add and with resource reusability but falls with technical risk. Similarly, in our framework, the decision to absorb complements responds to the call for understanding the "outside-in" form of open innovation where a firm enriches its knowledge base by integrating external resources adapted to its needs (Enkel et al., 2009). We show how the optimal non-compete period, i.e. the time until the platform absorbs complementary innovations, rests on balancing current developer royalties against future innovation value. This result generalizes attributes of contracts used by Cisco and by SAP. The duration of the exclusionary period rises as developers add more value and technical risk rises.

Section 4 highlights a third form of open innovation that emphasizes "permissionless innovation." The platform firm solves a coordination problem by offering a simple default contract: the firm offers access to its IP, but in exchange gets access to developer IP. This exchange not only saves on the multi-party negotiation costs among developers, known in other industries as the "tragedy of the anti-commons" (Heller and Eisenberg, 1998), but it also means developers never have to disclose their ideas to the platform before implementing them, and thereby risk losing them (Cerf, 2012). The number of developers does not need to be very large in order that all developers prefer using platform IP to using an open standard. Knowing this, the manager's task is to design a default contract in such a way that developers choose the permissionless innovation option over their outside option, modeled here as an open standard. This is not obvious *ex ante* because developers who join the platform will not only pay royalties, but they will also eventually lose their innovations. In exchange, developers gain access to more innovation resources and more valuable sales. Therefore, by forcing openness on developers, the platform can effectively induce R&D spillovers that benefit everyone. In this sense, the focus of managerial attention shifts from maximizing individual firm profits to maximizing those of an ecosystem. Because of the emphasis on value creation outside the platform, the manager acts more like a social planner.

This has important implications for industry regulators as well as platform designers. Antitrust authorities might view appropriation of developer IP as evidence of coercive monopoly – unhealthy competition between platform and developer – especially as a platform becomes large. As noted in the introduction, three of the top five firms by market capitalization in 2015 were platforms. Such an antitrust view could overlook the important mechanism by which the platform *became* large. A condition of anti-competitiveness might not be the primary reason for such contracts but rather the secondary consequence. A useful test of welfare enhancing behavior is then whether IP appropriated by the platform is subsequently redistributed by the platform.

Consistent with Green and Scotchmer (1995), our analysis also implies that the duration of IP protection should favor the upstream party relative to the downstream party, in our case, IP protection should favor the platform relative to the application. This holds because the sponsor would lack the means to control developers in later periods if its own IP rights expired in earlier periods. Expiration of platform IP would effectively convert developer decisions to the condition of operating under an open standard as just analyzed. This IP view of platforms parallels the architectural view of platforms (Baldwin and Woodard, 2009), which holds that the platform should function as a stable and slow evolving set of modules relative to applications that function as a flexible and rapidly evolving set of modules.

5.2 Extensions

Analysis is relatively robust to changes in our assumptions. Major assumptions include (1) a point estimate of consumer value, (2) a Cobb-Douglas production model, (3) a one stage useful lifetime for open platform stock and developer applications, and (4) dynamics limited to two stages.

In keeping with other papers in the literature, we assume point mass consumer demand for tractability. Consumers as end users enjoy positive surplus in our model as a result of platform openness and finite property rights for developer output. Also, many information goods are sold in bundles, making a point mass estimate of average value a reasonable approximation. Bakos and Brynjolfsson (1998) show that the standard deviation of the item values in a bundle can be made arbitrarily small by aggregating additional goods into the bundle. Adding multiple features to a platform is easily interpreted using such an average value v. Interestingly, if we allow Hi and Lo consumer types, such that only Hi types buy from developers during the intellectual property rights time duration, then the platform could sell to Lo types after bundling. This would require the platform to be "closed," but it might allow the sponsor to extract additional rents from developer innovations.[12] In the current framework, all consumers have all apps so this is not feasible.

The common assumption of Cobb-Douglas production is, again, made for tractability and allows for simple results expressed in terms of constant elasticity of output with respect to changes in technology. Similar conclusions can be obtained with alternative formulations but results are particularly elegant with the current specification. Our model also introduces a novel choice parameter, contractual openness, which plays a central role.

Relaxing the assumption of a one stage lifetime for developer output would complicate the

[12] We thank a reviewer for this observation.

analysis but also strengthen results as increased code longevity would increase R&D spillovers. If open platform stock stimulates production beyond one stage, increasing developer output also increases willingness to open the platform. Similarly, extending the two stage model to multiple stages or to continuous time would not undermine the main results. In fact, we know from the Folk Theorem that the cooperative – here open – strategies become optimal in the infinitely repeated game. Here, the necessary and sufficient ingredient is the recursive production function where the output of one stage becomes input to the next stage. In contrast, reducing the model to one stage could change results as "reuse" could be lost. More stages preserve or amplify effects of reuse.

6 Discussion & Conclusions

The rapid growth of the platform business model suggests that firms must acquire new capabilities to orchestrate external partners if they wish to pursue such platform strategies. There are key decisions that firms must make in order to foster and benefit from external innovation. Even those firms that are aware of the decisions can benefit from a better understanding of the key tradeoffs.

Our contribution is to model two key decisions that firms must make in order to manage a platform micro economy. These decisions are (i) the amount of internal resources to expose to external innovators and (ii), the rate at which external innovation is folded back into the platform for all participants to build upon. A successful platform sponsor achieves a "private ordering" where R&D spillovers are available for participants to use. It acts as a self-interested social planner for its microeconomy, making choices that account for end user consumption and developer production through cycles of recombinant innovation. Several intuitions follow.

We show how platform sponsors can optimize openness. Firms in our model find it privately rational to stimulate third-party innovation even at the cost of sacrificing rents from direct platform sales. The rule for optimal openness is to give away enough free access that its value in the current stage is proportional to developer elasticity of output in the next stage. Optimal openness declines in response to a rise in intrinsic platform value, but rises in response to rising developer value, the sizes of developer and end user pools, and rising resource reuse. Interestingly, the level of openness and, equivalently, the size of the subsidy in our model can exceed the current value of the platform.

Second, analogous to the duration of patent protection, we identify conditions for a finite exclusionary time. In our model, this represents the time during which developers can charge for new applications before the sponsor folds their innovations into the open platform. Platform absorption should occur at the point at which second-stage developer output exceeds first-stage output. If second-stage output is smaller than first-stage output, then it is never optimal for the platform to expire developer's IP rights. The optimal exclusionary time increases in response to an increase in developer value, remains unaffected by changes in reuse, and decreases in response to rising technical risk.

We contribute to theory by limiting the earlier finding that optimal IP duration can be arbitrarily long (Gilbert and Shapiro, 1990; Landes and Posner, 2002). Earlier models do not account for recursive production, which have a significant effect on the optimal outcome. Our model can also allow IP duration to be infinitely long, but then developer output can not be increasing in later stages. If developer output is increasing, then finite durations are optimal. As a practical matter, earlier patent duration in the U.S. had been tied to the length of an apprenticeship (Schrage, 1991). Here, we have shown formally how IP duration can be tied to industry specific factors such as technical risk, developer number, and developer output.

Our analysis of developer participation shows that developers can prefer sponsored platforms even when platforms take developer IP. For this to happen, sponsors need longer duration property rights, in keeping with earlier findings that the time of protection should favor the upstream innovator relative to that of downstream innovators (Green and Scotchmer, 1995). As a contribution to practice, we find that platform sponsors should design contracts that reserve authority to take developers' innovations and they should share these innovations with the ecosystem to spur further innovation. This practice must be carefully managed. On one hand, developers can view a platform sponsor as acting too aggressively when taking IP. On the other hand, developers face monopoly distortions if they must buy each other's innovations while facing the increasingly complex task of integrating disparate applications themselves. Platform coordination solves this problem.

From a social welfare perspective, we show that a benevolent planner chooses to open a greater portion of the platform and to expire IP rights sooner than does a self-interested platform sponsor. However, increasing costs lead the choices of platform sponsors and social planners toward convergence.

Finally, we demonstrate a prisoners' dilemma where developers individually prefer to close their IP even though they prefer that every other developer open theirs. As a result, given a sufficiently large developer pool, all developers are better off if a strong platform sponsor compels developers to open their innovations so that each might build from ideas of the other. IP spillovers in a platform ecosystem resemble R&D spillovers in a geographic region, with the added benefit of being subject to optimization by contract. The platform sponsor must enforce such contracts not only for benefit of the platform and the users, but also for the developers themselves. These results matter both for platform designers as they seek to design optimal contracts for innovation ecosystems and for industry regulators as they seek to boost innovation competitiveness and social welfare. Temporal dynamics show a complex but important set of interactions among openness and control.

References

Andreessen, M. 2007. The three kinds of platforms you meet on the internet. URL http://pmarchive.com/three_kinds_of_platforms_you_meet_on_the_internet.html.

Audretsch, D.B., M.P. Feldman. 1996. R&d spillovers and the geography of innovation and production. *The American Economic Review* **86**(3) 630–640.

Bakos, Y., E. Brynjolfsson. 1998. Bundling information goods: Pricing, profits and efficiency. *Management Science* **45**(12) 1613–1630.

Baldwin, C., J. Woodard. 2009. The architecture of platforms: A unified view. A. Gawer, ed., *Platforms, Markets and Innovation*, chap. 2. Edward Elgar Publishing Limited, Cheltenham, UK, 131–162.

Benkler, Yochai. 2002. Coase's penguin, or, linux and" the nature of the firm". *Yale Law Journal* 369–446.

Bessen, J., E. Maskin. 2009. Sequential innovation, patents, and imitation. *The RAND Journal of Economics* **40**(4) 611–635.

Boudreau, K. 2010. Open platform strategies and innovation: Granting access versus devolving control. *Management Science* **56**(10) 1849–1872.

Ceccagnoli, M., C. Forman, P. Huang, DJ Wu. 2012. Cocreation of value in a platform ecosystem: The case of enterprise software. *MIS Quarterly-Management Information Systems* **36**(1) 263.

Cerf, Vint. 2012. Remarks at the digital broadband migration: The dynamics of disruptive innovation: Internet speculations. *J. on Telecomm. & High Tech. L.* **10** 21.

Chang, H.F. 1995. Patent scope, antitrust policy, and cumulative innovation. *The RAND Journal of Economics* **26**(1) 34–57.

Chesbrough, H.W. 2003. *Open innovation: The new imperative for creating and profiting from technology*. Harvard Business Press.

Edwards, S. 2001. Openness, productivity and growth: what do we really know? *The Economic Journal* **108**(447) 383–398.

Eisenberg, M.A. 1976. Private ordering through negotiation: Dispute-settlement and rulemaking. *Harvard Law Review* 637–681.

Eisenmann, T., G. Parker, M. Van Alstyne. 2009. Opening platforms: How, when and why? A. Gawer, ed., *Platforms, Markets and Innovation*, chap. 6. Edward Elgar Publishing Limited, Cheltenham, UK, 131–162.

Eisenmann, T., G. Parker, M. Van Alstyne. 2011. Platform envelopment. *Strategic Management Journal* **32**(12) 1270–1285.

Enkel, Ellen, Oliver Gassmann, Henry Chesbrough. 2009. Open R&D and open innovation: exploring the phenomenon. *R&D Management* **39**(4) 311–316.

Evans, D.S., A. Hagiu, R. Schmalensee. 2006. *Invisible engines: how software platforms drive innovation and transform industries*. The MIT Press.

Gawer, A., M.A. Cusumano. 2002. *Platform leadership: How Intel, Microsoft, and Cisco drive industry innovation*. Harvard Business Press.

Gawer, A., M.A. Cusumano. 2008. How companies become platform leaders. *MIT Sloan management review* **49**(2) 28–35.

Gawer, A., R. Henderson. 2007. Platform owner entry and innovation in complementary markets: Evidence from intel. *Journal of Economics & Management Strategy* **16**(1) 1–34.

Gilbert, R., C. Shapiro. 1990. Optimal patent length and breadth. *The RAND Journal of Economics* **21**(1) 106–112.

Gillette, F. 2011. The rise and inglorious fall of MySpace. *Business Week* **June 27** 52–59.

Green, J.R., S. Scotchmer. 1995. On the division of profit in sequential innovation. *The RAND Journal of Economics* **26**(1) 20–33.

Hagiu, A., J. Wright. 2011. Multi-sided platforms. Tech. Rep. Working Paper 12-024, Harvard Business School.

Hargadon, Andrew. 2003. *How breakthroughs happen: The surprising truth about how companies innovate*. Harvard Business Press.

Heller, M.A., R.S. Eisenberg. 1998. Can patents deter innovation? the anticommons in biomedical research. *Science* **280**(5364) 698.

Hof, R. 2013. Ipo filing: Twitter still losing big money even as 2012 revenues tripled. *Forbes* .

Husney, Jordan. 2014. Crowd tough. URL http://www.contagious.com/blogs/news-and-views/12029081-opinion-crowd-tough.

Jackson, TP. 1999. Us v. microsoft: Findings of fact. Tech. rep., Civial Action No. 98-1232.

Kent, S.L. 2001. *The Ultimate History of Video Games*. Three Rivers Press.

Klemperer, P. 1990. How broad should the scope of patent protection be? *The RAND Journal of Economics* **21**(1) 113–130.

Laffan, L. 2011. A new way of measuring openness, from android to webkit: The open governance index. *VisionMobile.com* .

Landes, W., R. Posner. 2002. Indefinitely renewable copyright. Tech. rep., University of Chicago Law & Economics, Olin Working Paper No. 154.

Laursen, K., A. Salter. 2005. Open for innovation: the role of openness in explaining innovation performance among uk manufacturing firms. *Strategic Management Journal* **27**(2) 131–150.

Lawler, Ryan. 2014. Uber opens its api with 11 launch partners. *TechCrunch* **Aug. 20**.

Manjoo, F. 2012. Is its new chat app a sign that facebook's falling behind? *Slate* .

McAfee, R.P., J. McMillan, M.D. Whinston. 1989. Multiproduct monopoly, commodity bundling, and correlation of values. *The Quarterly Journal of Economics* **104**(2) 371–383.

Nalebuff, B. 2004. Bundling as an entry barrier. *The Quarterly Journal of Economics* **119**(1) 159–187.

Noe, T., G. Parker. 2005. Winner take all: Competition, strategy, and the structure of returns in the internet economy. *Journal of Economics and Management Strategy* **14**(1) 141–164.

Parker, G., M. Van Alstyne. 2000a. Information complements, substitutes, and strategic product design. *Proceedings of the twenty first international conference on Information systems*. Association for Information Systems, 13–15.

Parker, G., M. Van Alstyne. 2000b. Internetwork externalities and free information goods. *Proceedings of the Second ACM conference on Electronic Commerce*. Association for Computing Machinery, 107–116.

Parker, G., M. Van Alstyne. 2005. Two-sided network effects: A theory of information product design. *Management Science* **51**(10) 1494–1504.

Parker, G., M. Van Alstyne. 2009. Six challenges in platform licensing and open innovation. *Communications & Strategies* **Q2**(74) 17–36.

Partha, Dasgupta, Paul A David. 1994. Toward a new economics of science. *Research policy* **23**(5) 487–521.

Piskorski, J., T. Eisenmann, D. Chen, B. Feinstein. 2012. Facebook's platforms. *Harvard Business School Case* **9-808-128**.

Rochet, J.C., J. Tirole. 2003. Platform competition in two-sided markets. *Journal of the European Economic Association* **1**(4) 990–1029.

Rusli, Evelyn M., Douglas MacMillan. 2014. Uber gets an uber valuation. *Wall Street Journal* .

Salinger, M.A. 1995. A graphical analysis of bundling. *Journal of Business* 85–98.

Schrage, Michael. 1991. Archaic patent laws need to be rewritten. *Los Angeles Times* **Oct. 24**.

Singh, J., L. Fleming. 2010. Lone inventors as sources of breakthroughs: Myth or reality? *Management Science* **56**(1) 41–56.

Tadelis, Steven. 2013. *Game Theory: An Introduction*. Princeton University Press.

Tee, R., J. Woodard. 2013. Architectural control and value migration in layered ecosystems: The case of open-source cloud management platforms. *Mimeo* .

Teece, D.J. 1988. Capturing value from technological innovation: Integration, strategic partnering, and licensing decisions. *Interfaces* **18**(3) 46–61.

Tirole, Jean. 1988. *The Theory of Industrial Organization*. MIT press.

Valloppillil, Vinod, Josh Cohen, Eric S Raymond. 1998. The halloween documents. *Microsoft Corporation http://www. opensource. org/halloween* .

Wacksman, Barry, Chris Stutzman. 2014. *Connected by Design*. Jossey-Bass, San Francisco, CA.

Weitzman, Martin L. 1998. Recombinant growth. *Quarterly Journal of Economics* **113**(2) 331–360.

West, J. 2003. How open is open enough? melding proprietary and open source platform strategies. *Research Policy* **32**(7) 1259–1285.

West, J., W. Vanhaverbeke, H. Chesbrough. 2006. *Open Innovation: Researching a New Paradigm*. Oxford University Press, USA.

Weyl, E Glen. 2010. A price theory of multi-sided platforms. *The American Economic Review* **100**(4) 1642–1672.

Williamson, O.E. 2002. The lens of contract: private ordering. *American Economic Review* **92**(2) 438–443.

Yoo, Youngjin, Ola Henfridsson, Kalle Lyytinen. 2010. The new organizing logic of digital innovation: An agenda for information systems research. *Information Systems Research* **21**(4) 724–735.

7 Appendix of Proofs

Proof of Proposition 1 - Optimal solutions for δ and σ

Proof.

We first develop results for δ. Recall that (i) the interior solution occurs when second stage output exceeds first stage output, (ii) the condition for a finite developer property rights period is that first stage output must exceed the developer subsidy, and (iii) it is never profit maximizing to force immediate openness on developer applications.

Since δ terms do not appear in y_1 or y_2, we simplify by expressing profit in terms of output. Note that, given parameter restrictions, $\frac{\partial^2 \pi_{ps}}{\partial \delta^2} = -2vy_2 < 0$, which shows π_{ps} is concave in δ. Solving first order conditions provides a global maximum. Beginning from equation 1, calculate first-order conditions on platform profit with respect to δ as follows:

$$\frac{\partial \pi_{ps}}{\partial \delta} = -y_1 v + y_2 v(1-\delta) - \delta y_2 v = 0. \tag{13}$$

To establish result (i), rearrange terms in equation 13 to arrive at the first case of Eqn. 4 which is repeated here for reader convenience.

$$\delta^*_{interior} = \frac{1}{2}\left(1 - \frac{y_1}{y_2}\right)$$

This interior solution exists if and only if $y_2 > y_1$. For $y_2 \leq y_1$, the optimal δ is 0 (i.e. t is ∞), meaning that it is best never to bundle new applications into the platform. This is the corner solution, occurring when later output does not increase in the subsidy. To establish result (ii), substitute primitives in the inequality $y_2 > y_1$, giving $k^{1+\alpha}(\sigma V)^{\alpha^2} > k(\sigma V)^{\alpha}$. Raise both sides by $1/\alpha$ and reduce to see that equivalently $y_1 > \sigma V$. Finally, to establish result (iii), observe that $\delta^* \leq \frac{1}{2}$ always therefore t^* is bounded above zero always.

To facilitate our analysis of the platform sponsor's choice of σ, we first establish the existence and uniqueness of σ^*. To be proven: there exists a unique σ^* that maximizes platform profit. From Proposition 1, we see that are two cases to analyze. In case 1, δ^* has an interior solution such that $\delta^* \in (0,1)$. In case 2, $\delta^* = 0$. We analyze each case in turn. Recall that platform profit is

$$\pi_{ps} = V(1-\sigma) + \frac{1}{2}v(1-\delta)k(\sigma V)^{\alpha} + \delta \frac{1}{2}v(1-\delta)k^{1+\alpha}(\sigma V)^{\alpha^2}. \tag{14}$$

Case 1: interior $\delta^* \in (0,1)$

The first-order condition on platform profit with respect to σ is:

$$\frac{\partial \pi_{ps}}{\partial \sigma} = -V + \alpha \frac{1}{2}v(1-\delta)k\sigma^{\alpha-1}V^{\alpha} + \alpha^2 \frac{1}{2}\delta v(1-\delta)k^{1+\alpha}\sigma^{\alpha^2-1}V^{\alpha^2} = 0. \tag{15}$$

Before proceeding, we check the second order condition for concavity of the platform profit function in σ. We substitute $\delta^* = \frac{1}{2}\left(1 - \frac{y_1}{y_2}\right) = \frac{1}{2}\left(1 - \frac{k(\sigma V)^\alpha}{k^{1+\alpha}(\sigma V)^{\alpha^2}}\right)$ into the platform profit function and take the second derivative with respect to σ to get the following expression.[13]

$$\frac{\partial^2 \pi_{ps}}{\partial \sigma^2} = \frac{1}{8}\alpha k v \sigma^{-\alpha^2-2} V^{-\alpha^2}\left((\alpha-2)(\alpha-1)^2\sigma^{2\alpha}k^{-\alpha}V^{2\alpha} + \alpha\left(\alpha^2-1\right)\sigma^{2\alpha^2}k^\alpha V^{2\alpha^2} + 2(\alpha-1)\sigma^{\alpha(\alpha+1)}V^{\alpha(\alpha+1)}\right) \tag{16}$$

Given positive values for primitives, $\sigma \geq 0$, and $\alpha \in (0, 1)$, note that each additive term inside the parentheses is negative. We conclude that the second derivative is negative.

Returning to the first order condition, we multiply through by σ and rearrange terms to get the following expression

$$\sigma V = \frac{1}{2}\alpha k v(1-\delta)\left((\sigma V)^\alpha + \alpha k^\alpha \delta(\sigma V)^{\alpha^2}\right).$$

Divide through by σV and pull $(\sigma V)^{\alpha-1}$ out front to get

$$1 = \frac{1}{2}(\sigma V)^{\alpha-1}\alpha k v(1-\delta)\left(1 + k^\alpha \alpha\delta(\sigma V)^{\alpha^2-\alpha}\right).$$

Let $L = k(\sigma V)^{\alpha-1}$. Then we have

$$1 = \frac{\alpha v}{2}L(1-\delta)\left(1+\alpha\delta L^\alpha\right).$$

Since $y_1 = k(\sigma V)^\alpha$ and $\delta^* = \frac{1}{2}\left(1 - \frac{y_1}{y_2}\right)$, we get the following expression

$$\delta^* = \frac{1}{2}\left(1 - \frac{1}{L^\alpha}\right). \tag{17}$$

Thus

$$\begin{aligned} 1 &= \frac{\alpha v}{2}L(1-[\frac{1}{2}(1-\frac{1}{L^\alpha})])\left(1+\alpha[\frac{1}{2}(1-\frac{1}{L^\alpha})]L^\alpha\right), \\ 1 &= \frac{\alpha v}{4}\left(L + L^{1-\alpha}\right)\frac{1}{2}\left(2+\alpha L^\alpha - \alpha\right). \end{aligned}$$

Define

$$f(L) = 1 = \frac{\alpha v}{8}\left(L + L^{1-\alpha}\right)\left(2-\alpha+\alpha L^\alpha\right). \tag{18}$$

Given $\alpha \in (0, 1)$ and $L > 0$, then $f(L)$ increases monotonically in L. Since $f(0) \to 0, f(\infty) \to \infty$ there exists a unique $L^*(\alpha, v)$ that solves $f(L^*) = 1$. Given $L = k(\sigma V)^{\alpha-1}$, $\alpha < 1$ implies that L monotonically decreases in σ. Thus $f(L)$ can be expressed as $f(L(\sigma))$ and a unique L^* implies a unique σ^*. Importantly, σ is not bounded by 1, so solutions with $\sigma > 1$ are feasible, and as argued in the text above, likely to be observed in practice.

[13]This requires some tedious algebraic manipulation that can be carried out mechanically using software such as mathematica.

Optimal Solution for σ in Case 1: interior $\delta^* \in (0,1)$

Take the first order condition of platform profit in 3 with respect to σ. Then, multiply all terms by σ to raise the production exponent by +1 and reproduce developer output in its primitive form to get

$$\begin{aligned}\frac{\partial \pi_{ps}}{\partial \sigma} &= -V + \frac{1}{2}\alpha p k (\sigma V)^{\alpha-1} + \frac{1}{2}\alpha^2 p k^{1+\alpha} (\sigma V)^{\alpha^2-1} = 0, &(19)\\ &= -V\sigma + \frac{1}{2}\alpha p y_1 + \frac{1}{2}\alpha^2 p y_2 = 0. &(20)\end{aligned}$$

Add σV to both sides and substitute developer profit $\pi_{d1} = \frac{1}{2}py_1$ and $\pi_{d2} = \frac{1}{2}py_2$ in stages 1 and 2. Cobb-Douglas production yields, $\eta_1 = \alpha$ and $\eta_2 = \alpha^2$. Substituting η terms for α terms completes the derivation.

Optimal Solution for σ in Case 2: corner $\delta^* = 0$

Again, calculate the first-order condition on platform profit with respect to σ. However, in this case, $\delta^* = 0$ and $p = v(1-\delta)$ implies $p \to v$. The second stage term goes to zero and the expression reduces to:

$$\frac{\partial \pi_{ps}}{\partial \sigma} = -V + \frac{1}{2}\alpha v k \sigma^{\alpha-1} V^{\alpha} = 0. \tag{21}$$

To ensure concavity, we check the second derivative with respect to σ to get

$$\frac{\partial^2 \pi_{ps}}{\partial \sigma^2} = \frac{1}{2}(\alpha - 1)\alpha v k \sigma^{\alpha-2} V^{\alpha}. \tag{22}$$

The second derivative is clearly negative. Returning to the first order condition, note that the expression simplifies to

$$(\sigma V)^{1-\alpha} = \alpha v k/2. \tag{23}$$

Raise both sides to $1/(1-\alpha)$ and solve for σ to get the closed form solution

$$\sigma^* = \frac{(\alpha v k/2)^{1/(1-\alpha)}}{V}. \tag{24}$$

■

Derivation of Corollary 1 - Comparative Statics

Comparative statics for σ^*

Using the derivations developed in Proposition 1, we explore the behavior of the platform choice variables of openness and time to bundle developer innovations as a function of exogenous parameters. Note that the comparative statics for σ^* are the same for both cases 1 and 2.

$\frac{\partial \sigma^*}{\partial V} < 0$

Case 1, $\delta^* \in (0,1)$: Given $L = k(\sigma V)^{\alpha-1}$, σ^* must fall in V in order to maintain the equality in equation 18.

Case 2, $\delta^* = 0$: By equation 24, σ falls in V.

$\frac{\partial \sigma^*}{\partial v} > 0$

Case 1, $\delta^* \in (0,1)$: The right-hand-side of equation 18 increases in v. Thus L^* falls in v in order to maintain the equality. Therefore σ^* increases in v.

Case 2, $\delta^* = 0$: By equation 24, σ increases in v.

$\frac{\partial \sigma^*}{\partial k} > 0$

Case 1, $\delta^* \in (0,1)$: Equation 18 establishes that a unique solution exists in L that optimizes platform profit. Given $0 < \alpha < 1$ and $L = k(\sigma V)^{\alpha-1}$, we conclude that σ^* increases in k.

Case 2, $\delta^* = 0$: By equation 24, σ increases in k.

Comparative statics for δ^*

Note that comparative statics for δ^*only make sense in Case 1, $\delta^* \in (0,1)$. Therefore the derivations below refer only to this case.

$\frac{\partial \delta^*}{\partial V} = 0$

Equation 17 expresses δ in terms of L. By equation 18, L^* is constant with respect to V.

$\frac{\partial \delta^*}{\partial v} < 0$

By equation 17, δ^* increases in L^*. By equation 18, L^* falls in v. Therefore δ^* falls in v. This is consistent with the derivation above. By equation 4 (with primitives substituted for y terms), δ^* falls in σ and we showed earlier that σ increases in v; thus δ^* falls in v.

$\frac{\partial \delta^*}{\partial k} = 0$

Equation 17 expresses δ in terms of L. By equation 18, L^* is constant with respect to k.

Proof of Proposition 2 - Welfare

Proof. To establish the claim with respect to δ, solve the platform sponsor's maximization problem inclusive of cost. Taking the first order condition of platform profit π^c_{ps} w.r.t. δ leads the platform sponsor to choose

$$\delta^*_c = \frac{1}{2}\left(1 - \frac{y_1}{y_2} - \frac{c y_2^{1/\alpha} + F}{v y_2}\right). \tag{25}$$

The social planner chooses δ subject to the participation constraint $\pi_d^c \geq 0$ for cost recovery. Solving for δ produces two roots. Eliminate the negative root by choosing $c = F = 0$. In the absence of cost, the positive root reduces to $\delta = 1$. Hence, absent the need to recover cost, a social planner prefers to release developer additions immediately. Otherwise, the social planner chooses

$$\delta_c^\dagger = \frac{1}{2}\left(1 - \frac{y_1}{y_2} - \frac{cy_2^{1/\alpha} + F}{vy_2} + \Delta\right). \tag{26}$$

All terms except $\Delta = \frac{\sqrt{4vy_2(vy_1 - cy_1^{1/\alpha} - F) + ((vy_2 - cy_2^{1/\alpha} - F) - vy_1)^2}}{vy_2}$ are the same as those chosen by the platform sponsor. Observing that Δ is the positive root completes the claim. Also note that $\delta_c^\dagger > \delta_c^*$ implies that the developer constraint is always satisfied by the platform sponsor's choice.

To establish the claim with respect to σ, apply the steps used in Proposition 1 to the system of equations including costs to produce the following pair of implicit functions.

$$\sigma_c^\dagger \quad : \quad \alpha(vy_1 - \frac{1}{\alpha}cy_1^{1/\alpha}) + \delta_c^\dagger \alpha^2 (vy_2 - \frac{1}{\alpha}cy_2^{1/\alpha}) = 0 \tag{27}$$

$$\sigma_c^* \quad : \quad \alpha(py_1 - \frac{1}{\alpha}cy_1^{1/\alpha}) + \delta_c^* \alpha^2 (py_2 - \frac{1}{\alpha}cy_2^{1/\alpha}) = 2\sigma V \tag{28}$$

Transform the first by mapping $\delta_c^\dagger$ to δ_c^* and the second by mapping p to v. As second stage surplus is always non-negative, the welfare and profit constraints can be sorted.

$$\sigma_c^\dagger \quad : \quad \alpha(vy_1 - \frac{1}{\alpha}cy_1^{1/\alpha}) + \delta_c^* \alpha^2 (vy_2 - \frac{1}{\alpha}cy_2^{1/\alpha}) = -\kappa_1 < 0 \tag{29}$$

$$\sigma_c^* \quad : \quad \alpha(vy_1 - \frac{1}{\alpha}cy_1^{1/\alpha}) + \delta_c^* \alpha^2 (vy_2 - \frac{1}{\alpha}cy_2^{1/\alpha}) = \kappa_2 > 0 \tag{30}$$

Where $\kappa_1 = \alpha\Delta(\alpha vy_2 - cy_2^{\frac{1}{\alpha}}) > 0$ and $\kappa_2 = 2\sigma V + \alpha\delta vy_1 + \alpha^2\delta^2 vy_2 > 0$. Under model assumptions, the first constraint binds always to the left of the second. In this case, producing $\sigma_c^\dagger > \sigma_c^*$.
■

Proof of Proposition 3 - Technological Uncertainty

Proof. The result can be evaluated as follows. The effect of increasing technical success ω goes in the same direction as increasing output $\mathbf{E}(Y_2)$. Proposition 1 establishes that increasing $\mathbf{E}(Y_2)$ increases both σ^* and δ^*. ■

Proof of Proposition 4 - Prisoners Dilemma in Open Cooperation

Given the two stage structure of the game, the standard solution concept is subgame perfect Nash equilibrium (SPNE). Table 3 provides the payoffs. Because there are four possible Player 1 and Player 2 actions in the first stage (CC, CD, DC, DD), there are four subgames. In order to solve the game, we use backward induction.

Player Actions	Payoffs	
$a_1(1)a_2(1);a_1(2)a_2(2)$	$\pi_1(a_1(1)a_2(1);a_1(2)a_2(2))$	$\pi_2(a_1(1)a_2(1);a_1(2)a_2(2))$
Subgame 1		
CC;CC	$(1-\delta)vy_1+(1-\delta)2^{\alpha}\delta vy_2$	$(1-\delta)vy_1+(1-\delta)2^{\alpha}\delta vy_2$
CC;CD	$(1-\delta)vy_1+(1-\delta)2^{\alpha}\delta vy_2$	$(1-\delta)vy_1+2^{\alpha}\delta vy_2$
CC;DC	$(1-\delta)vy_1+2^{\alpha}\delta vy_2$	$(1-\delta)vy_1+(1-\delta)2^{\alpha}\delta vy_2$
CC;DD	$(1-\delta)vy_1+2^{\alpha}\delta vy_2$	$(1-\delta)vy_1+2^{\alpha}\delta vy_2$
Subgame 2		
CD;CC	$(1-\delta)vy_1+(1-\delta)\delta vy_2$	$vy_1+(1-\delta)2^{\alpha}\delta vy_2$
CD;CD	$(1-\delta)vy_1+(1-\delta)\delta vy_2$	$vy_1+2^{\alpha}\delta vy_2$
CD;DC	$(1-\delta)vy_1+\delta vy_2$	$vy_1+(1-\delta)2^{\alpha}\delta vy_2$
CD;DD	$(1-\delta)vy_1+\delta vy_2$	$vy_1+2^{\alpha}\delta vy_2$
Subgame 3		
DC;CC	$vy_1+(1-\delta)2^{\alpha}\delta vy_2$	$(1-\delta)vy_1+(1-\delta)\delta vy_2$
DC;CD	$vy_1+(1-\delta)2^{\alpha}\delta vy_2$	$(1-\delta)vy_1+\delta vy_2$
DC;DC	$vy_1+2^{\alpha}\delta vy_2$	$(1-\delta)vy_1+(1-\delta)\delta vy_2$
DC;DD	$vy_1+2^{\alpha}\delta vy_2$	$(1-\delta)vy_1+\delta vy_2$
Subgame 4		
DD;CC	$vy_1+(1-\delta)\delta vy_2$	$vy_1+(1-\delta)\delta vy_2$
DD;CD	$vy_1+(1-\delta)\delta vy_2$	$vy_1+\delta vy_2$
DD;DC	$vy_1+\delta vy_2$	$vy_1+(1-\delta)\delta vy_2$
DD;DD	$vy_1+\delta vy_2$	$vy_1+\delta vy_2$

Table 3: 2-Player Payoff Table

Proof.

For clarity, we first provide the intuition for closing code using the 2-player case. We then extend this to the N-player case. Finally, we show that benefits of the cooperative open code solution increase in the size of the developer pool.

To analyze the outcome of the two-stage two-person prisoner's dilemma game, we will refer to Table 3 which lists the player payoffs to each player action. *2-Player Case:* The payoff matrices for each subgame are shown below. Note that in order to fit the strategic form payoffs onto a printed page, within each cell we list the payoff to Player 1 on top, and the payoff to Player 2 on the bottom.

Subgame 1 Payoff Matrix given first stage actions CC

	Developer 2: C	Developer 2: D
Developer 1: C	$(1-\delta)vy_1 + (1-\delta)2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + (1-\delta)2^{\alpha}\delta vy_2$	$(1-\delta)vy_1 + (1-\delta)2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + 2^{\alpha}\delta vy_2$
Developer 1: D	$(1-\delta)vy_1 + 2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + (1-\delta)2^{\alpha}\delta vy_2$	$(1-\delta)vy_1 + 2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + 2^{\alpha}\delta vy_2$

Subgame 2 Payoff Matrix given first stage actions CD

	Developer 2: C	Developer 2: D
Developer 1: C	$(1-\delta)vy_1 + (1-\delta)\delta vy_2$ $vy_1 + (1-\delta)2^{\alpha}\delta vy_2$	$(1-\delta)vy_1 + (1-\delta)\delta vy_2$ $vy_1 + 2^{\alpha}\delta vy_2$
Developer 1: D	$(1-\delta)vy_1 + \delta vy_2$ $vy_1 + (1-\delta)2^{\alpha}\delta vy_2$	$(1-\delta)vy_1 + \delta vy_2$ $vy_1 + 2^{\alpha}\delta vy_2$

Subgame 3 Payoff Matrix given first stage actions DC

Developer 2

Developer 1	C	D
C	$vy_1 + (1-\delta)2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + (1-\delta)\delta vy_2$	$vy_1 + (1-\delta)2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + \delta vy_2$
D	$vy_1 + 2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + (1-\delta)\delta vy_2$	$vy_1 + 2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + \delta vy_2$

Subgame 4 Payoff Matrix given first stage actions DD

Developer 2

Developer 1	C	D
C	$vy_1 + (1-\delta)\delta vy_2$ $vy_1 + (1-\delta)\delta vy_2$	$vy_1 + (1-\delta)\delta vy_2$ $vy_1 + \delta vy_2$
D	$vy_1 + \delta vy_2$ $vy_1 + (1-\delta)\delta vy_2$	$vy_1 + \delta vy_2$ $vy_1 + \delta vy_2$

In each of the four subgames, action D dominates action C. Thus, DD is a nash equilibrium in each of the subgames. We then solve the stage 1 game as follows.

Stage 1 Payoff Matrix after promotion of solutions to subgames 1:4

Developer 2

Developer 1	C	D
C	$(1-\delta)vy_1 + 2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + 2^{\alpha}\delta vy_2$	$(1-\delta)vy_1 + \delta vy_2$ $vy_1 + 2^{\alpha}\delta vy_2$
D	$vy_1 + 2^{\alpha}\delta vy_2$ $(1-\delta)vy_1 + \delta vy_2$	$vy_1 + \delta vy_2$ $vy_1 + \delta vy_2$

Stage 1 analysis: First, assume that Player 1 picks C. Then Player 2's best response is to choose D. Now assume that Player 1 chooses D. Player 2's best response is D. Now assume that Player 2 chooses D. Player 1's best response is D.

Thus $s_i = (s_i^1, s_i^{CC}, s_i^{CD}, s_i^{DC}, s_i^{DD}) = (D, D, D, D, D)$ is a subgame perfect nash equilibrium strategy for both developers.

■

N-Player Case: Table 4 lays out the payoffs $C(i)$ and $D(i)$ that a single player faces where i is the number of cooperating players among the remaining $N-1$ players.

Number (i) of cooperators among the remaining (N-1) players

		0	1	2		N-1
Single Player	Cooperate	C(0)	C(1)	C(2)	…	C(N-1)
	Defect	D(0)	D(1)	D(2)	…	D(N-1)

Table 4: Individual payoffs under decisions made by the remaining $N-1$ players.

Any single player's best response to the $N-1$ other players is always to defect. That is, at each stage, $D(i) > C(i)$. To see this, consider the left- and right-most columns of Table 4 where all remaining $N-1$ players either cooperate or defect. Adopting this binary choice allows us to use the standard two-player cooperate/defect construct to analyze the N-player game. This also makes sense because the payoffs are increasing in the number of cooperators, so we look at the extreme cases of maximum and minimum reward.

Now, consider the two-stage game where N developers ($N \geq 2$) choose to cooperate or defect in each stage. Denote $n_i \in [0, N]$ the number of developers who cooperate in stage $i \in \{1, 2\}$.

Proof. : We prove via backward induction.

At the beginning of stage 2, given that there are $n_1 \in [0, N]$ developers who have already cooperated in stage 1, if a developer is among those n_1 cooperating developers in stage 1, her stage 2 payoff of staying cooperating is

$$\pi_2(CC|n_1) = (1-\delta)n_1^{\alpha}\delta v y_2, \tag{31}$$

where CC represents cooperating in both periods. Alternatively, she can switch to defect in stage 2 and receive a payoff of

$$\pi_2(CD|n_1) = n_1^{\alpha}\delta v y_2. \tag{32}$$

Since $\pi_2(CD|n_1) > \pi_2(CC|n_1)$ when $\delta > 0$ and $n_1 > 0$[14], a stage 1 cooperating developer chooses to defect in stage 2.

Otherwise if a developer is among those n_1 defecting developers in stage 1, her stage 2 payoff of staying defecting is

$$\pi_2(DD|n_1) = (n_1+1)^{\alpha}\delta v y_2, \tag{33}$$

and the payoff for switching to cooperate in stage 2 is

$$\pi_2(DC|n_1) = (1-\delta)(n_1+1)^{\alpha}\delta v y_2. \tag{34}$$

[14]The case of $n_1 = 0$ does not exist if there is at least one cooperating developer in stage 1.

Again $\pi_2(DD|n_1) > \pi_2(DC|n_1)$ when $\delta > 0$ and $n_1 \in [0, N)$[15]. The above analysis shows that the unique Nash equilibrium for stage 2 is all developers defect (i.e., $n_2^* = 0$).

Now consider stage 1. The SPNE in period 2 has shown that all developers would defect in stage 2. There are only two possible equilibrium outcomes of n_1: (1) $n_1 = 0$, (2) $n_1 \in (0, N]$.

Outcome 1. $n_1 = 0$, all developers defect in stage 1. A developer's stage 1 payoff of deviating from $n_1 = 0$ is

$$\pi_1(C|n_1 = 0) = (1-\delta)vy_1 + \delta vy_2. \tag{35}$$

The payoff she receives by continuing to defect in stage 1 is

$$\pi_1(D|n_1 = 0) = vy_1 + \delta vy_2. \tag{36}$$

Since $\pi_1(C|n_1 = 0) < \pi_1(D|n_1 = 0)$ when $\delta > 0$, $n_1^* = 0$ constitutes a Nash equilibrium because no developer has an incentive to deviate.

Outcome 2. $n_1 \in (0, N]$. At equilibrium, there exists at lease one cooperating developer at stage 1. However, her payoff of cooperating is

$$\pi_1^C(C|n_1 \in (0, N)) = (1-\delta)vy_1 + n_1^\alpha \delta vy_2. \tag{37}$$

If she switches to defect, her payoff becomes

$$\pi_1^C(D|n_1 \in (0, N)) = vy_1 + n_1^\alpha \delta vy_2. \tag{38}$$

Since $\pi_1^C(D|n_1 \in (0, N)) > \pi_1^C(C|n_1 \in (0, N))$, a cooperating developer has an incentive to deviate. Thus, $n_1 \in (0, N]$ does not constitute a Nash equilibrium.

Combing the above two cases, we conclude that there is a unique equilibrium at stage 1 where $n_1^* = 0$. Take together our two-stage backward induction analysis, the unique pure strategy equilibrium is $n_i^* = 0$ for $i \in \{1, 2\}$. ■

Proof of Proposition 5 - Threshold for Cooperation

Proof.

Finding the threshold for cooperation is a straightforward comparison of the cooperative outcome with the uncooperative outcome. Thus we evaluate:

$$\begin{aligned}
(1-\delta)vy_1 + N^\alpha \delta vy_2 &\geq vy_1 + \delta vy_2 \\
\implies N^\alpha \delta vy_2 &\geq \delta vy_1 + \delta vy_2 \\
\implies N^\alpha y_2 &\geq y_1 + y_2 \\
\implies N^\alpha y_2 &> y_2 + y_2 \ (Tighter\, bound\, by\, Prop.\, 1) \\
\implies N &> 2^{1/\alpha}
\end{aligned}$$

Now, consider the individual change in benefits of adding a new member to a group that cooperates relative to a group that does not cooperate. That is, compare the benefits of cooperation for a group of size $N + 1$ to a group of size N.

[15] The case of $n_1 = N$ does not exist if there is at least one defecting developer in stage 1.

$$
\begin{aligned}
[(1-\delta)vy_1 + (1-\delta)(N+1)^{\alpha}\delta vy_2] - [(1-\delta)vy_1 + (1-\delta)N^{\alpha}\delta vy_2] &\geq [vy_1 + \delta vy_2] - [vy_1 + \delta vy_2] \\
\implies (1-\delta)(N+1)^{\alpha}\delta y_2 - (1-\delta)N^{\alpha}\delta y_2 &\geq 0 \\
\implies (N+1)^{\alpha} - N^{\alpha} &\geq 0
\end{aligned}
$$

Thus the cooperative payoff grows faster than the non-cooperative payoff for any N, $\alpha > 0$. A cooperating group of size N, growing to size $N+1$, should continue to cooperate as it adds new members. And, this incentive to cooperate increases with increases in technology α. ■