Platform Choice by Mobile App Developers

Tim Bresnahan, Joe Orsini, and Pai-Ling Yin *

Stanford University

Abstract

For over a year, Apple's iOS and Google's Android operating systems have split the market share of smartphone devices and the mobile applications (apps) for those devices. We model and estimate the platform choice by mobile app developers, including the decision to multihome. Our model flexibly models the potential gap between the decision to multihome and the realized demand from that decision. We find far less difference in preferences across platforms than across types of developers and apps. We identify strong incentives for developers of the most popular apps to multihome, making tipping unlikely.

The application development platform is the industrial organization used to create value out of information technology (IT). Applications developers convert a technical opportunity into economic value. Platforms lower the cost of entry through the exploitation of social increasing returns to scale, allowing new ventures to focus on value-creating aspects of the business – e.g., innovative new functionalities and applications – while a platform sponsor or sponsors update and enhance the underlying infrastructure. If there are diverse successful applications, that increases the economic value of the technical progress embodied in a platform's general purpose components.

An outstanding and dynamic example of the platform industrial organization is the dramatic rise in consumer use of mobile devices and the explosion of applications software running on those devices – mobile "apps." The invention of mobile app platforms has permitted developers to offer a system solution to their customers – re-using, not reinventing, the technology of mobile phones, mobile transmission, wi-fi, cloud technologies, and many other features. This dramatically lowers the fixed costs of building a technical system. Most of the economics literature on the mobile(wireless) industry studies wireless carriers like AT&T and Verizon. A new literature studies hardware devices, such as the Apple iPhone, in connection the carriers (Wilkinson (2013), Chintagunta et al. (2013)). Our contribution, in this and in earlier papers (Davis et al. (2013), Bresnahan et al. (2013)) has been to look at the software that defines the competitive platforms (iOS, Android, Windows mobile, etc.) and at the application software that creates economic value on top of those competitive platforms.

In the early stages of an IT industry, competition between platforms can play several roles. It can improve the incentives of platform providers to take actions and set policies that are pro-user and pro-developer, incentives that are quite hard to replicate in a mature proprietary platform industry. If there is uncertainty about the socially optimal platform technology or business model, competition between platforms at an early stage of the industry life cycle permits market selection. The platform(s) selected by users and developers in the market will reflect their tastes to some degree.

However, competition between platforms has costs. Since the platform industrial organization has social increasing returns to scale, the deployment of multiple platforms involves wasteful duplication. That competition involves duplication is true in any increasing returns industry, of course. The difference in platform industries is that the costs of the duplication are external, i.e. borne in substantial part by users and developers rather than by platform suppliers. From a developer perspective, the costs of platform competition arise because potential customers (users) are spread across multiple platforms. Developers would have to incur the costs of multihoming (developing for both platforms) if they want to reach their entire potential demand. The incentive to multihome is particularly strong (for any given "porting" costs) when platforms are approximately equal sized. In addition to lessening the costs of platform competition for themselves, multihoming developers lessen it for users. Widespread multihoming offers the users of any particular platform access to a wider range of apps. Therefore, in this context, the developer's multihoming costs are important to understanding the degree to which market fragmentation is problematic. For over a year, Apple's iOS and Google's Android operating systems have split the market share of smartphone devices and the mobile applications (apps) for those devices. Our analytical goal is to understand developer platform choice (including the choice to multihome) and its implications for platform economics in the smartphone industry, notably for the competition between iOS and Android.

Applications developers choose platforms for access to markets, for technical features, and for economic/market features. Access to markets are the most familiar: the costs of writing an application are fixed, and the more users their are on a platform, the larger the market for an application. The technical features of an applications platform are a driver of application development cost, as the general-components in the platform are combined with the specific innovation of the application to create a usable product. Applications platforms often offer contracts, rules, and restrictions to developers that affect their profitability. When choosing between two platforms, developer profitability depends on the relative size and on

the technical and contractual features.

Another, perhaps less familiar, consideration for developer platform choice is the institution established to match the developers app to the highest value customer. The founding of a new industry with the applications platform industrial organization enables the creation of a wide number of markets. Founding new markets involves establishing the institutions by which the prospective buyers and sellers of newly invented products discover one another. A major (but often overlooked) function of platforms is to define the environment for buyers and sellers.[1] How well or poorly they do this will affect the developer's market costs, and therefore determine whether they facilitate or bottleneck the rate and direction of innovation by developers in the industry. If the platform fails to efficiently match product to the demand segment that most values those products, the diffusion of innovation in those products is slowed. If the platform asymmetrically matches some products efficiently but not others, the direction of innovation will favor some developers over others. Since invention takes on very different forms, one would expect the market institutions to vary across different kinds of buyers and sellers. For example, an incremental invention that improves an existing product for known customers relies much less heavily on market institutions than do more breakthrough inventions. A developer contemplating producing for a particular platform will need to consider the marketing costs of connecting to customers on that platform, if those institutions are not efficient. Since the fixed costs of application development are largely sunk, this profit calculation depends also on a developers' forecast of the future growth of users on a platform.

Many of the same issues seen in developer platform choice are also seen in developer choice of whether to multihome. Two different costs enter the developer's multihoming decision: the technical costs of porting to another platform, and then the costs of achieving success in the other market. (See Figure 1) Typically, models of the multihoming decision overlook the existence of the second cost, subsuming it as part of the developer's rational expectations as to whether to write the app. We explicitly model these as separate and

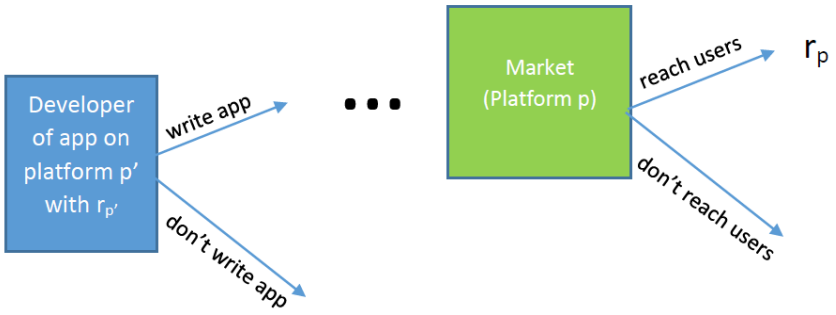


Figure 1: Model of Multihoming Decision

sequential components of the decision, since these costs are driven by two different sources. The first set of costs, the technical costs of porting, are determined by how feasible it is to write on the other platform. The second set of costs are determined by the market costs of being profitable on the other platform. Note that the second set of costs are defined by the market institutions implemented by the platform host.

Taken together, then, we see two elements on the cost side of a developer's choosing a particular platform either as its only market or for multihoming. First, there are the fixed costs of writing an app or of writing it for multiple platforms. Second, there are the costs of matching to users on each platform. Each of these costs may be larger or smaller for developers of different types.

The institutions of matching buyer and seller takes a particular form in mobile apps. Both Apple and Google supply app/user matching services in connection with their development platforms. Both are organized as collaborative filter recommendation systems, i.e., as "top lists." The commercialization / distribution facilities used by the platform providers reward success for app developers with more success, and reward near misses on success with very limited visibility. Thus app developers impose positive externalities on one another through the usual platform positive-feedback mechanisms, but also impose negative externalities on one another through congestion in visibility.

From a developer perspective, the organization of buyer seller matching around top lists

creates a particular marketing issue. To match to users through the app store, the app must be listed on top lists. To the extent the top lists are narrow (or that users typically visit only a narrow set of top lists) this limits the number of developers who can effectively use this channel. Developers who have other channels of communication with users will be less affected, as will developers with apps that would be on the top lists under perfect consumer information; other developers will face a bottleneck in gaining customers. The empirical size of that bottleneck, and the extent of blockage it offers to different types of developers, are open questions.

A very successful literature studies the choice of platforms or, very similarly, standards, by suppliers of complementary products. Rysman (2004) is the classic paper, where the complements to a yellow pages "platform" are advertisements. Some of this literature has taken up the question of complementor choice in applications platforms (Lee (2013), Dube et al. (2010)) and have considered changes in platform policies, such as openness. Most of these papers perforce study a mature platform or two sided market after the races, if any, for standardization have been run. Only a few papers (Augerereu et al. (2006), Dranove and Gandal (2003), Cantillon and Yin (2013), Church and Gandal (1992), Corts and Lederman (2009)) study the race itself. Uniquely among these papers, we study a development platform (as opposed to a trading platform) in a context where multihoming on one side of the market is possible.

We model and estimate the platform choice by mobile app developers, including the decision to multihome. We condition on satisfying a minimum threshold of quantity demanded in the market on at least one platform, so we are identifying multihoming costs off of developers who we know are capable of producing valuable apps. In this respect, we are closer to Berry (1992) and Sutton (1991) than to Bresnahan and Reiss (1991) in our treatment of entry. This allows us to address questions about the contribution to either single platform competitive advantage or to multihoming, i.e., to consumer advantage, without regard to consumer platform choice. We find that developers who develop non-games, are

publicly-traded firms, and employ preexisting user log-on credentials have a higher estimated demand in the market, and therefore have a higher incentive to multihome. The publicly-traded status may be an indicator of the ability of larger developers to employ existing customer relations and overcome the matching problem. The large numbers of competitors in the games category may exacerbate the negative externalities of the top list institution for matching. Since the most popular apps also successfully multihome, despite the usual incentives for the market tipping, the market is likely to remain split until one of the platforms can solve this matching problem.

The rest of the paper is structured as follows: Section I. reviews the industry setting of mobile apps. Section II. describes our comScore data and the mobile app questionnaire data we created to supplement that data. Section III. describes our model of developer choice over platforms and multihoming, and Section IV. describes how this is implemented in our data. We discuss our findings in Section V.. The last section concludes.

I. Industry Setting: Mobile Apps

The newness and popularity of the mobile apps industry has created many misconceptions about the industry structure and its practices, so we use this section to clarify the facts. Figures 2 and 3 show the history of users and apps on both platforms. It is apparent from these graphs that the industry is currently in an even split of market share, and has been in this fragmented state for quite a while given then short length of the industry's entire history.

In theory, and as speculated in the industry press, there are a number of factors that may explain the fragmentation in the installed base. On the consumer side, the demographics of the iOs and Android users are in fact different: Android users tend to have lower incomes, for example. The split between Android and iOS may be a vertical differentiation over the cost of the device. It may also reflect differentiation in preferences over the device. These may also be correlated with differentiated preferences in apps, which would then lead to

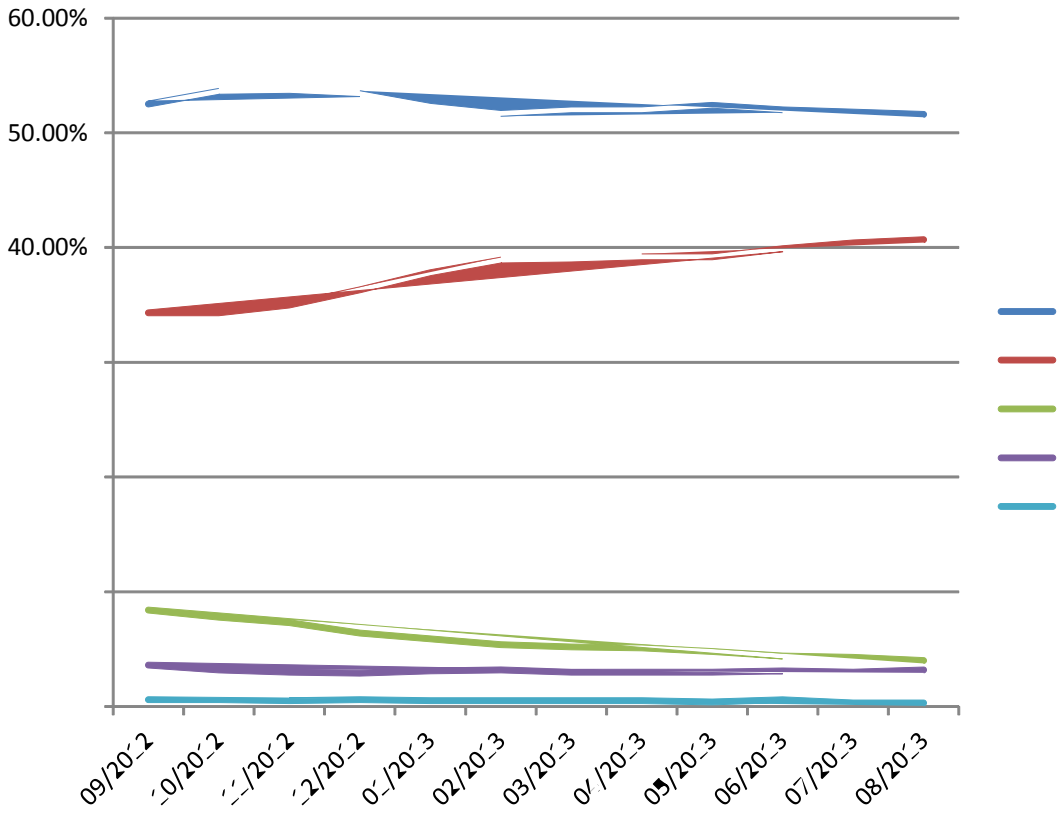


Figure 2: History of Smartphone Market Shares. Based on comScore Mobile Metrix data. Other sources have Android and iOS closer (e.g., Pew reports them as 28% and 25% of all smartphones).

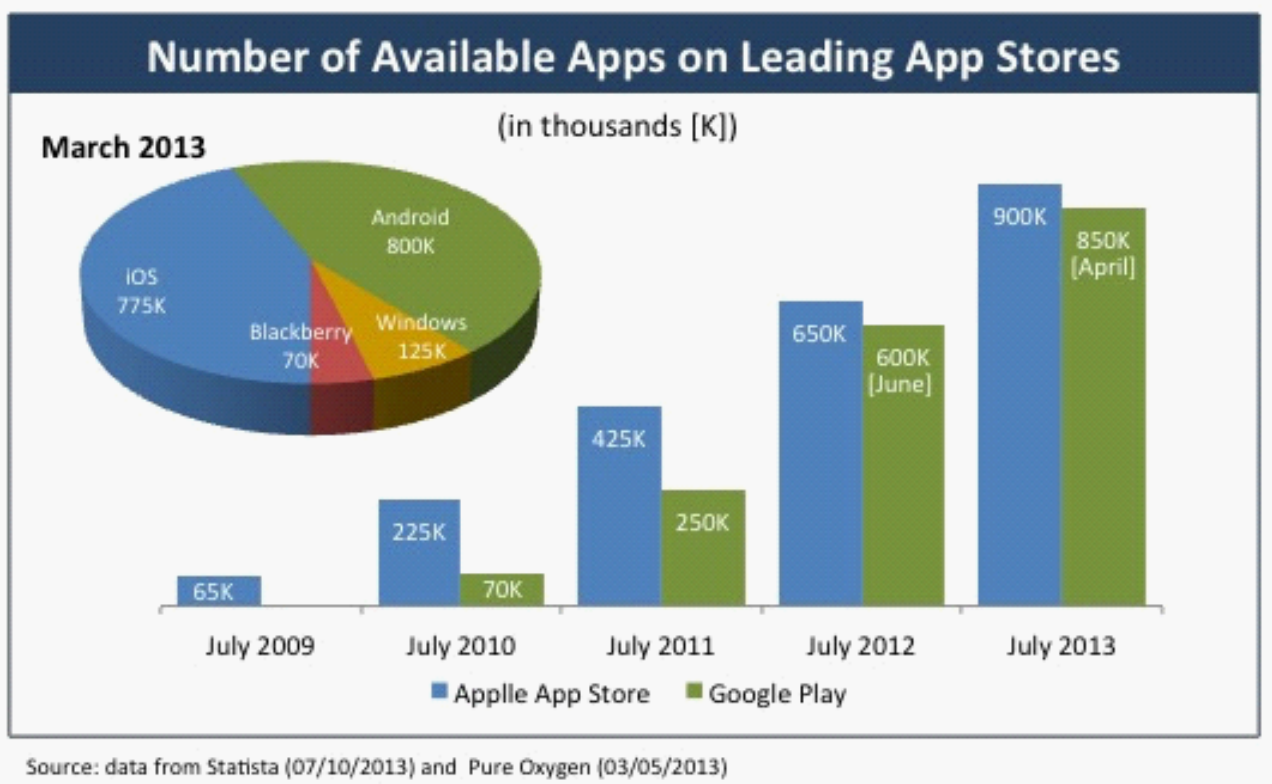


Figure 3: History of Apps Available for iOS vs. Android. Accessed from http://socialwebbiz.files.wordpress.com/2013/07/apps_availablenumbersonappstores.jpg

developers splitting themselves between the two platforms. It is possible that consumers are multihoming with multiple smartphones, but this would not be a large enough phenomenon to explain equal market shares on the consumer side.

Even if the preferences of consumers did not drive developers to different platforms, there are additional reasons that we might find an equal number of apps for both platforms. There are a number of ways in which the iOS and Android platforms differ from the perspective of the developer (see Bresnahan et al. (2013)). Finally, developers might be multihoming, if the costs of multihoming are low enough and the benefits are large enough. Indeed, if the benefits to reaching a large body of users on the other platform is high enough, it could outweigh even the benefits of platform differentiation that might drive a developer to favor one platform over the other.

One very important fact to note from Figure 3 is the huge number of apps on both platforms. These huge number of applications may make it easier for the consumer to adopt the iOS or Android platform without too much consideration of which platform has the most applications: indirect network effects for the consumer may be overwhelmed by a critical mass of apps on each platform that provide sufficient utility. However, this same fact makes it extremely difficult for app developers to gain visibility and get matched to the right consumer. The developers may be spreading across platforms to reduce competition as much as possible.

One way in which the platforms do not differ signficantly is the mechanism through which they predominantly match apps to users: top lists. These top lists tend to aggregate user downloads across categories, employing typical collaborative filter methods to try and identify the most useful or popular apps. Unfortunately, since the stores and these mechanisms are effectively the only channels through which apps can reach their customers, they create incentives for apps to game the system and try to "buy" their way into the top rankings. Even without these distortions, simply trying to get noticed out of the close to 1 million apps competing for a users attention will require some marketing expenditure, which

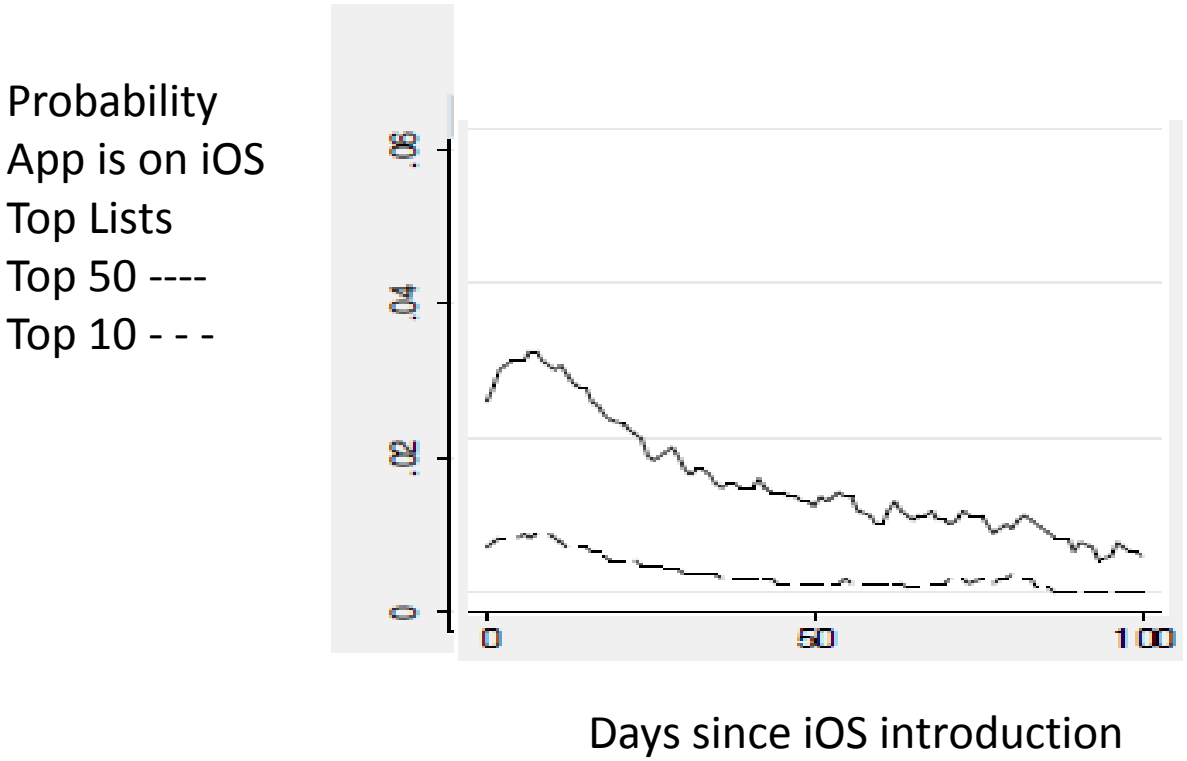


Figure 4: Costs and Risks of Buying into the App Store Rankings

is why we attempt to incorporate some flexibility to allow for a gap between the decision to multihome and the realized profitability in multihoming as expressed in Figure 1.

Our many discussions with industry participants corroborates patterns we find in the data regarding the high costs of marketing an app. In Figure 4, we profile the probability that apps make it into the Top 50 and Top 10 lists. The initial increase in the first few days after launch of the probability of being in these top lists is often fueled by apps who typically invest in marketing through "incentivized downloads" (purchasing downloads from users) and advertising their app in other apps for the express purpose of increasing their standings in the rankings. These launch campaigns average approximately $0.5 million. Despite these efforts, the probability of being on the top lists drops dramatically in the 2nd through 4th weeks, indicating that for many of the apps, these incentivized downloads and advertising campaigns do not work.

In this paper, we will be able to test whether strong platform preferences divide devel-

opers or multihoming makes fragmentation a stable equilibrium.

II. Data

We utilize the January 2013 Mobile Metrix dataset from comScore. Two panels, one with Android phones and the other with iPhones, of approximately 5,000 US adult users, allow comScore to track their possession and usage of apps. We observe the data aggregated to the app*platform*month level.

We also only observe apps which meet a minimum usage test for each month: for each platform, iOS (iPhone) or Android (smartphone), comScore includes data on the app only if it is used on that platform by more than 5 (at least 6) unique users. For each platform on which the app has met this usage criterion, comScore reports its estimate of the app's "reach", defined as the proportion of the population of mobile phone owners who use it. If the app does not meet this criterion on a particular platform, then nothing about it is reported on that platform. Consequently, based on the mobile metrix data alone, we cannot definitively infer the set of platforms on which the app has been published, as it is quite possible that the app has been written for both Android and iOS, but has only met comScore's usage criterion on one of those platforms. We drop apps produced by Apple, Google, and carriers, which are typically pre-loaded onto devices, so that an observation is an independent software vendor. This results in 1231 apps. The distribution of their reaches are presented in Figure 5.

In this paper, we augment our comScore sample with questionnaire data generated by our research team. We asked undergraduate research assistants to download each app in the sample and fill in a questionnaire with 200 questions regarding the app's use of advertising, in-app purchases, and other monetization strategies. Crucially, this questionnaire also asks about the app's platform availability and the developer type. By combining these two sources, we observe, for each app in the sample, the set of platforms on which the developer chose to publish that app, whether or not the app had at least 6 users on comScore's panel

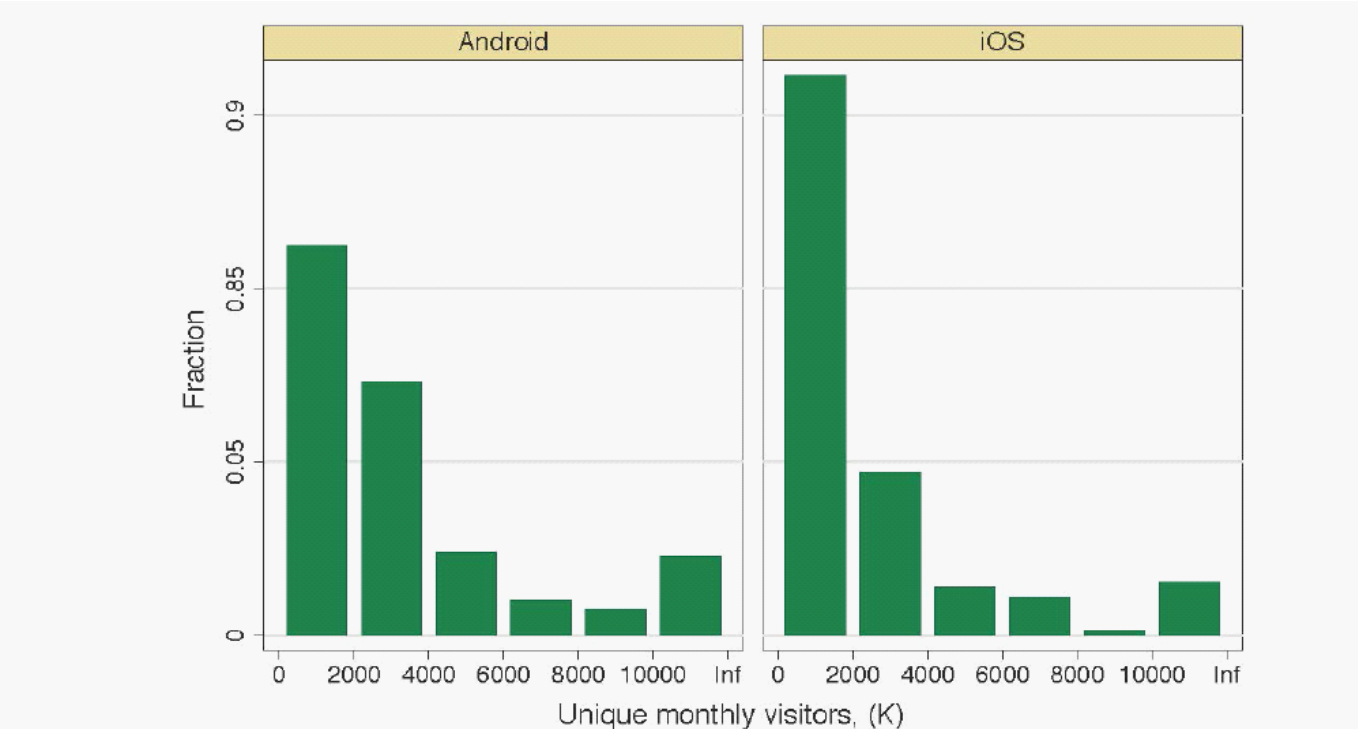


Figure 5: Distribution of Reaches over Apps. Source: comScore, January 2013. This histogram of app usage has equally spaced bins, except for the rightmost bin which has no upper limit. The vertical axis is the log of the fraction (reach) in the bin.

on each platform it was published on, and comScore's reach estimate on each platform for which the app had at least 6 users on the panel. We multiply the "reach" estimate on each platform by 5,000, the size of comScore's platform-specific sub-panels, to back out the number of panel-members who used the app. The set of indicator variables concerning the developer's publication decisions and the app's satisfaction of comScore's usage criterion, together with the number of panelists who use the app when that criterion is met, form the dependent variable in our model. Explanatory variables, collected in the vector X, will include indicators for the type of app (whether it is a game (27% of apps), exhibits advertising (54% of apps) or in-app purchases (IAP, 29% of apps), uses pre-existing login credentials (49% of apps) and information about whether the app's developer and/or parent company is publicly traded (37% of apps). It turns out that publicly traded apps are almost perfectly correlated with apps with no advertising and no IAP, so we only use publicly traded in our final estimation.[2] The distribution of these regressors can be found in Figure 6.

Figure 7 reveals the distribution of multihoming apps in our sample. Note that over half of the apps multihome, but that only one-third of those apps reach the threshhold for being observed by comScore on both platforms. The correlation between reaches in (B,B) is 0.60. When we restrict (B,B) to those apps for which the demand threshold is raised to 1%, the correlation is 0.86. Further note that there seems to be an equal number of apps on iOS and Android that reach the demand threshhold for comScore. There are two possible drivers of the apps which are available on both platforms but cannot reach even 0.1% of demand: either the preferences of the consumers on the platforms differ, or the apps were marketed differently (either due to different effort on the part of the developer or different competitive conditions on the platforms). Given our knowledge of the industry setting, we believe the latter is the stronger driver, and we will identify observables that will allow us to falsify or corroborate this hypothesis in our estimation.

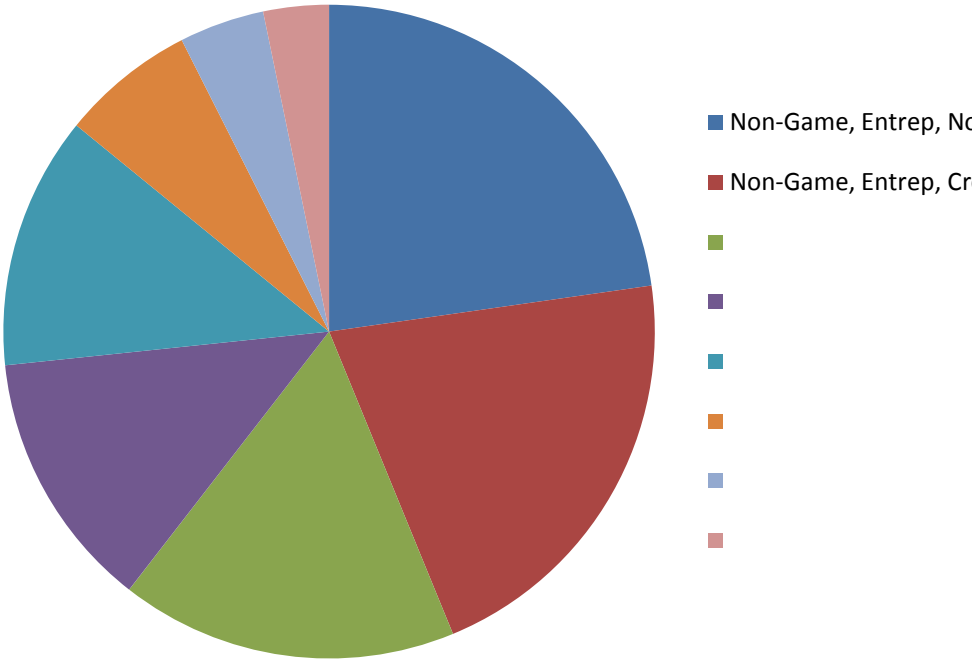


Figure 6: Distribution of Regressors. This is the proportion of the apps that fall into each combination of three regressors used in our estimation: whether the app is a game or non-game, whether its developer is publicly traded or an entrepreneur, and whether the app employs previously established login credentials.

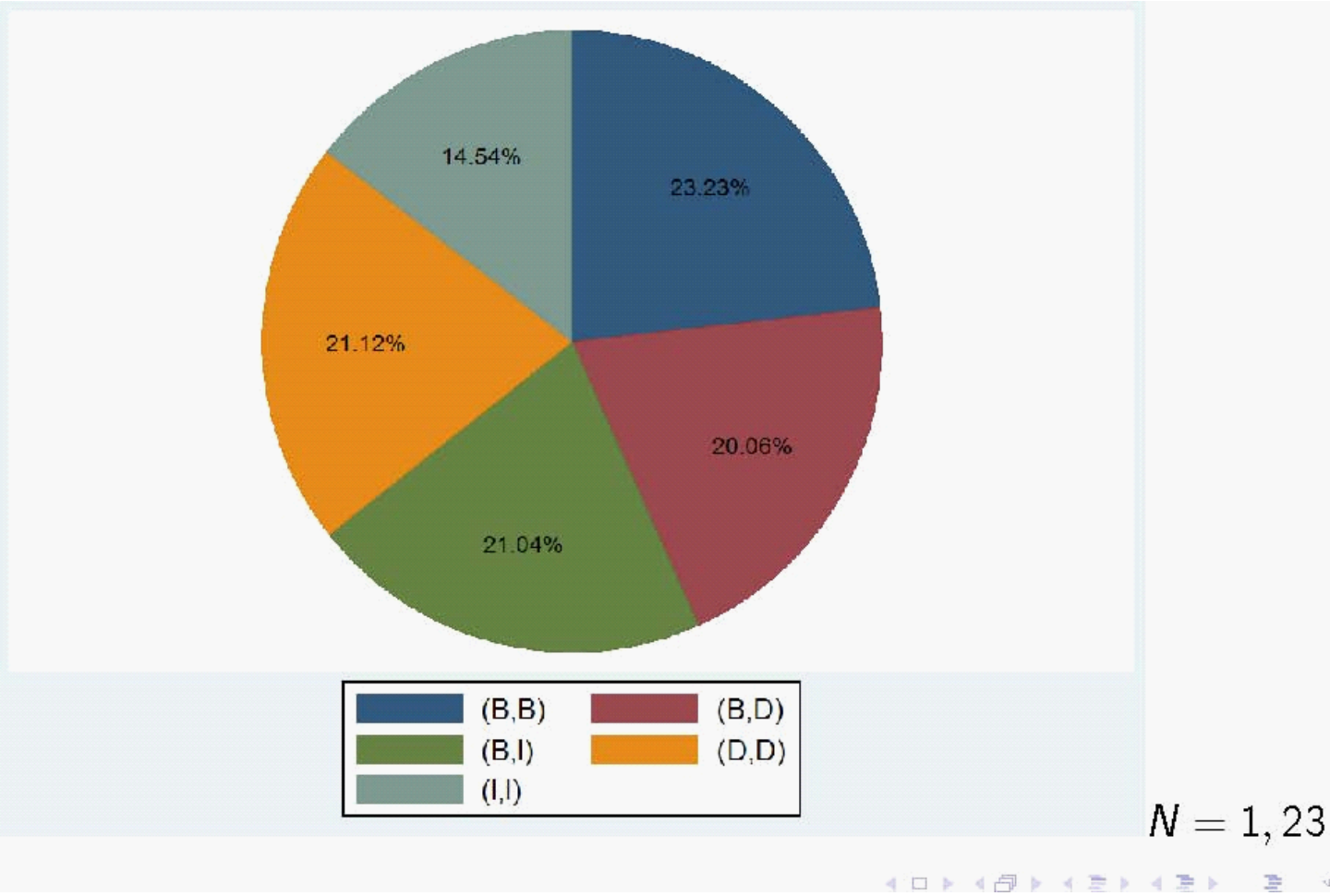


Figure 7: Multhoming in comScore Jan 2013. B= Both, D=Android, I=iPhone. The first letter in each pair indicates whether the app was written for both platforms or only Android or the iPhone, based on data from our undergrad questionnaire. The second letter in each pair indicates whether the app had enough quantity demanded to be observed in comScore on both or only one of the platforms.

III. Economic Model

The central features of our economic model of developer's supply decisions were laid out at the beginning of the paper. In this section, we make them more specific and focus them on the observables in our dataset.

Like all entry models, our model assumes that the developer has a fixed costs and will only produce if the size of the market is sufficiently large to cover those costs. In our context, the developer makes supply decisions about each of the two markets, I, the iOS platform and D, the Android platform, so we will need to consider the fixed costs of entering either or both. A second novel element of our model is the distinction between the technical fixed costs of writing an app for a platform and the costs of matching the app with consumers. Because our model of matching has a search flavor, we model one important "cost" of matching the app to consumers as a probability the app will be found by far fewer customers than is warranted by its true customer appeal.

A. Profits from an App (gross of fixed costs and marketing "costs")

Let the reach of an app (we will add app subscripts later) on platform p ($p = I$ or D) be denoted r_p. Reach is the size of the market for an app on a platform, measured as the fraction of platform users that demand the app. Let M_p be the variable profits per customer for the app on the platform and let U_p be the number of users of platform p. Then the variable profits of an app on a platform is $U_p \times M_p \times r_p$. Let the fixed costs of writing an app for only one platform be C_p and the technical costs of writing for both platforms be $C_B \leq C_D + C_I$.If there were only one platform, and if there were no fixed marketing costs, then the condition for it to be profitable to write for a platform would be

$$U_p \times M_p \times r_p \geq C_p. \tag{1}$$

B. The App Store and App Matching

In our approach, writing an app for a platform does not automatically expose it to all the available users. Once the app has been written for platform p, it appears in that platform's online app store.

Not all apps can appear on the app store's top list, and we model the probability that an app appears on the top list as $A(\widetilde{r}_p, \tau, E_p)$. Here $\widetilde{r}_p$ is the fraction of platform users that would choose the app if they knew of it, τ is a parameter, and E_p is a fixed cost of marketing the app. An app that appears in the top list is downloaded by all consumers that desire it. However, an app that is not on the top list is downloaded by fewer customers because some that desire it have not seen it.[3] Thus we interpret τ as a parameter that indexes the efficiency of the app store at matching.

We interpret $\widetilde{r}_p$ as the fraction of customers who would choose the app if they saw it displayed on the top list in the app store. Let q_p be another random variable, i.e. the fraction of customers who choose the app if it is *not* displayed on the platform's top lists. In general, we expect $q_p < \widetilde{r}_p$. Thus the effective demand for the app, r_p, is given by

$$r_p = \begin{cases} \widetilde{r}_p \text{ with probability } A(\widetilde{r}_p, \tau) \\ \\ q_p \text{ otherwise .} \end{cases}$$

To complete the model, we specify the information available to the developer at each choice point and thus state the developers' behavioral equation. In the simplest version of our model, the developer observes $(\widetilde{r}_I, \widetilde{r}_D)$, and decides whether to spend the fixed costs of writing the app for each platform and the fixed costs of marketing on each platform at once.

C. Three Simplifying assumptions

We now make three simplifying assumptions that we will maintain throughout. These assumptions make the model much more transparent, and only two of them are economic restrictions. First, we assume $E_B + C_B = E_D + C_D + E_I + C_I$. This is an economic

assumption that the platforms are very different, i.e. that there is no fixed-cost savings from writing for both. In our context, we make this assumption because of what developers have told us about their cost function. It is not that porting is extremely difficult technically, which would tend to make the C_B close to $C_D + C_I$. Instead, it is the large size of E relative to C. E is necessarily per-platform, so if most of the fixed costs are fixed marketing costs, then there is little scale economies to multi-homing.

Second, we change the units of the technical fixed costs and write $K_p = (E_p + C_p)/M_p/U_p$. This is simply a normalization. We will estimate K; the normalization determines the interpretation. Since, in our application, $U_D = U_I$, if we find that $K_D > K_I$ the interpretation is either that the fixed costs of writing an Android app are higher $E_D + C_D > E_I + C_I$ or that the per-customer profits of an Android app are lower $M_D < M_I$. We cannot distinguish these two hypotheses.

Third, in our statistical model of app supply, we normalize K_p to be a constant across apps, so that variation in profitability across apps is driven only by r_p. This is an economic assumption; since $K_p = C_p/M_p/U_p$, this is the assumption not only that there is no fixed-cost errors[4] but also that per-customer profits are constant across apps.[5]

D. Behavior and Developer Information

We assume that a developer chooses to write the app for platform p if and only if his expected variable profit from doing so, conditional on the signal he observes, exceeds the fixed cost. We further assume that the developer observes the reach of the app under the assumption that every user could see it, but does not ex ante observe whether the app goes to the top of the app store top list.[6] Therefore, the app is written for platform p if and only if the following condition is satisfied:

$$\mathbb{E}[r_p|\widetilde{r}_I, \widetilde{r}_D)] = \mathbb{E}[\widetilde{r}_p A(\widetilde{r}_p, .) + (1 - A(\widetilde{r}_p, .))q_p|\widetilde{r}_I, \widetilde{r}_D)] \geq K_p. \tag{2}$$

IV. Econometric Specification and Parameterization

To estimate the model, we need to specify the joint distributions of all the random variables $(\widetilde{r}_p,\, q_p)$, $A()$, and so on. A key feature of our data is that we observe reach in the ComScore dataset, which we call s_p. ComScore (1) truncates $s_p < .001$ and (2) ComScore has a small enough panel size – 5000 per platform – that we need to be careful about the truncation. It is particularly important to be careful about truncation if, as seems possible here, the threshold either for publishing the app on one platform or for multihoming the app might be close to the truncation point. Fortunately, in this instance we know the truncation rule. Accordingly, we specify the model so that we can calculate the probability of truncation on each platform ($\Pr(s_p < .001)$) easily. We label apps which are used by more that one in a thousand demanders on a platform, and thus which are visible in ComScore on that platform, as "hit" apps on a platform.

A. Sample

Our sample consists of apps that are hits on at least one platform. As a result, although there are 9 possible states (combinations of whether an app is available one each platform and whether it has acheived the demand threshold for each platform) for apps in total, we only observe 5 of them (shaded) in Figure 8. The reach that is observed for each state is presented in each cell. Since this sample definition is conditional on the dependent variable of app supply to platforms, we must also define the likelihood to adjust for this conditioning.

Entry models generally need to condition on some version of the dependent variable in order to define the set of potential entrants. The problem is that firms that have not entered any market do not exist. A variety of solutions have grown up to deal with this. Berry (1992), and many papers following in that tradition, define the set of potential entrants into a particular market by observing firms that have already entered related markets. In his example, potential entrants into a particular airline city pair market are firms serving other city pair markets already. This sample definition conditions on the lagged dependent

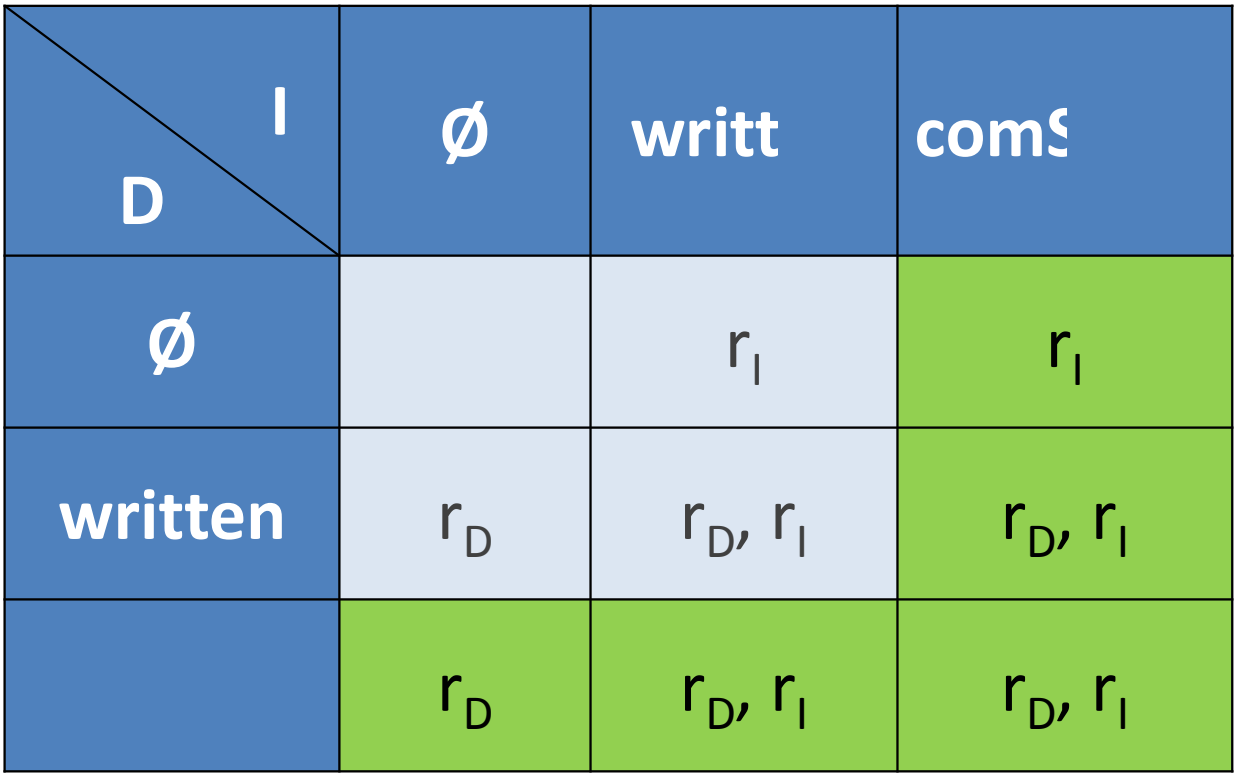


Figure 8: App Supply States. D= Android, I=iOS, ∅ = not written for platform D and/or I, written = written for platform D and/or I, comScore = acheived reach r above demand threshold on platform D and/or I.

variable in an adjacent market. One could hope that the double attenuation that comes both from lagging and from looking at an adjacent market renders any sample selection bias small (the most common solution) or one could attempt to correct for sample selection bias. Another solution is to avoid defining a set of potential entrants, but instead examine a fixed set of market niches each of which could (or could not) be filled by a particular entrant. Bresnahan and Reiss (1991) use this approach, as does Mazzeo (2002) who looks at entry in the categories of high-quality motels and low-quality motels. The Bresnahan-Reiss approach only works when one can define the market niches a priori, so we cannot use it. The Berry approach, with a correction for the sampling process, works for us because we observe a similar sample, i.e., for entry into iOS we look at hit apps on Android.

Our model can be understood as having many of the features of a "Type II" Tobit. This familiar model is written $y_2 = \left\{ \begin{array}{c} y_2^* \text{ if } y_1^* > 0 \\ 0 \text{ if } y_1^* \leq 0 \end{array} \right\}$, where y_2 is a variable, like reach on one platform for app i for us, that is only observed if an event, labeled in the Tobit as $y_1^* > 0$, occurs.

A simplified version of our model would have exactly this structure. Suppose we were only studying entry into platform D, and we gather our sample of firms that have entered and been successful on platform I. Then we would let $y_2^* \equiv r_D$ and introduce $y_1^* \leq 0$ as a model of sample selection. If these two random variables (demand on platform D and entry and success on platform I) are correlated, as one would expect, then we would use the Type II Tobit to deal with the relevant sample selection problem.

This simplified version is how we solve a classic econometric problem in models of entry. If the set of potential entrants into one market are actual entrants in another market, and if profits in the two markets are correlated, there is a sample selection problem.

We use this same sample selection logic but embed it in the structure of our model, which involves a few changes in the specifics. First, our model is symmetric across the two platforms, so that there are two y_2^* variables, and we have different conditions for observing

the two y_2 variables. We have two variables like y_2: r_D and r_I are both observed only conditional on events. Second, in our model, $y_1^* > 0$ (the condition for observing r_D or r_I) is a compound event, depending not only on the realization of reach but also on the app being included in our sample. We observe r_D if both (a) $\widetilde{r}_D > .001$ and (b) the app is part of our sample. Third, we observe not only the truncated reach, but also the fact of writing for the platform or not; these events are observable whenever the app is part of our sample. None of these complications changes the nature of our approach to sample selection profoundly.

We choose a symmetric (across platforms) sample selection rule, and thus study entry into (and success on) the markets defined by both platforms simultaneously. We have both economic and econometric reasons for taking this approach. From an economic perspective, our platform focus leads us to studying multihoming, i.e., entry into both platforms, which requires the symmetric treatment. From an econometric perspective, modeling both platforms at once is our best identification strategy. We have no instruments for the "selection equation," so we need to put economic restrictions on the conditioning event. In our approach, each conditioning event has a full economic model, since it is part of our dependent variable.

B. Sampling Correction

We start from the universe of potential app developers who have an idea for an app. Let the characteristics of a developer, d, be W_d. Recall that the event "written for neither, D, I, or both platforms" is named S. Using the same notation, define T to be the event "above threshold demand on neither platform, D, I, or both". Recall also $s = s_I$, s_D is the reach of the app conditional on it being observed. We are interested in the joint distribution of s, T, S[7] conditional on W and on the sampling frame.

Since the sampling frame is conditional on T but not directly on S or s, more specifically consists of the event $T \neq \varnothing$, we need to calculate the probability of the event $T \neq \varnothing$ to

convert from the unconditional distribution to the conditional distribution. What we need to know is $f(s, T, S|W, \theta)$, the unconditional likelihood of an app, which we will derive in the next subsection, and $F(T \neq \varnothing|W, \theta)$ the marginal probability of the event that the observation is not truncated.

To display $F(T \neq \varnothing|W, \theta)$ in our framework, first start from the probability of S, which follows from developer's optimum choice of S. Call this $P_S(W, \theta) = \begin{matrix} p_0 \\ p_I \\ p_D \\ p_B \end{matrix} (W, \theta)$. Second, we have P_{TS} , the probability of T given S. For example, the upper left element ($\Pr_{TS}(T = \varnothing|S = \varnothing)$) is 1. If the events "a hit on I conditional on being published on I" and "a hit on D conditional on being published on D" were independent, we could write the probability of these events as ψ_I and ψ_A. (Henceforth we suppress the dependence on W and θ.) Then we would have

$$\Pr_{TS} = \begin{matrix} 1 & 1-\psi_I & 1-\psi_D & (1-\psi_D)(1-\psi_I) \\ 0 & \psi_I & 0 & \psi_I(1-\psi_D) \\ 0 & 0 & \psi_D & \psi_D(1-\psi_I) \\ 0 & 0 & 0 & \psi_I\psi_D \end{matrix}$$

and thus we can see that

$$F(T) = \begin{matrix} p_0 - p_D(\psi_D - 1) - p_I(\psi_I - 1) + p_B(\psi_D - 1)(\psi_I - 1) \\ \psi_I p_I - \psi_I p_B(\psi_D - 1) \\ \psi_D p_D - \psi_D p_B(\psi_I - 1) \\ \psi_D \psi_I p_B \end{matrix}$$

The first row is $F(T = \varnothing)$ and illustrates the logic of the truncation in our sample. An app can be truncated because it was never written (p_0), because it was written for only on platform and not a hit on that platform, or because it was written for both and not a hit on either. The event $T = \varnothing$ is complex, so that calculating the likelihood conditional on our sampling rule requires a parametric approach to the probabilities.

In our model, the events "above the demand threshold on I conditional on being pub-

lished on I" and "above the demand threshold on D conditional on being published on D" are not independent. This implies a slightly more complex notation. We continue to write the probability of the event "above the demand threshold on p conditional on being published on p" as ψ_P but now write the probability of the event "below the threshold on either conditional on being published on both" as ψ_{-B} The distinction matters when the $\tilde{r}_p$ are correlated across platforms. Positive dependence between $\tilde{r}_I$ and $\tilde{r}_D$ leads to $\psi_{-B} < (\psi_D - 1)(\psi_I - 1)$. Thus in our model we calculate $F(T = \varnothing) = p_0 - p_D(\psi_D - 1) - p_I(\psi_I - 1) + p_B\psi_{-B}$.

We assume that $X \sim$ bernoulli(ω). Let $X \perp \tilde{r}_1 \perp \tilde{r}_2 \perp \tilde{r}_3$ with $\tilde{r}_i \sim$ beta$(\alpha_i, \beta_i), i = 1, 2, 3$. Assume that $\mu_3 = \frac{\alpha_3}{\alpha_3+\beta_3} = \frac{\mu_1}{2} + \frac{\mu_2}{2}$ and $\sigma_3^2 = \frac{\sigma_1^2}{2} + \frac{\sigma_2^2}{2}$.

Our dependence mixture model of the joint distribution of true reaches assumes that

$$r_I = Xr_3 + (1 - X)r_1$$

and

$$r_D = Xr_3 + (1 - X)r_2.$$

We assume that $Z_p|r_p \sim$ Bernoulli$(\gamma(\tilde{r}_p, \tau))$ and $r_{qp} \sim$ beta$(\alpha_{qp}, \beta_{qp})$ are independent

$$r_p = Z_p\tilde{r}_p + (1 - Z_p)q_p,$$

and that

$$\gamma(\tilde{r}_p) = r_p\mathbb{I}(\tilde{r}_p < \tau) + \mathbb{I}(\tilde{r}_p \geq \tau), \text{ where } \tau \in (0, 1).$$

V. Results

The costs of the app store E_p fall only on firms that are trying to aqcuire new customers in the mobile world (e.g. mobile entrepreneurs) and much less for those for whom the mobile world is an extension of existing customer relationships (e.g. banks). We use a publicly

traded dummy (PT) as a regressor to capture this disinction. Estimated parameters are presented in Tables 1 & 2.

Our estimates indicate that there is little developer heterogeneity in platform profitaibliy. This contradicts the explanation that strong platform preferences divide the developers. Over all the different types of developers, we find that the mean expected reach on Android and iOS and the thresholds K are approxmiately the same (Table 3). The most observable heterogeneity between platforms can be found in Table 2 for entrepreneurial non-games developers who do not seek user credentials (12.8% of the sample). There is fare more heterogeneity across developer/app types within a platform than across platforms. The large (publicly traded) developers multihome, and thus have access to the users on both platforms. They are not encumbered by the competition to rank highly on the collaborative filters, since they already have a relationship with their current installed base of customers. For the smaller, newer developers, the choice to single-home would drastically reduce their potential demand, given the split in the market. However, multihoming could be quite costly in terms of the marketing efforts to get noticed by customers (with whom they do not already have a relationship).

A. Other sampling schemes

We have chosen a particular sampling scheme – a hit on at least one platform – because it permits us to implement a parametric version of a correction for sample selection and because it matches our economic goals of understanding network effects. It is a model which might be thought of as a symmetric-across-markets version of the Berry (1992) approach of studying entry from one market into another. Other approaches are possible.

One approach would be to find the entry dates of each product onto each platform and then change our model to predict (a) which platform was entered first and (b) whether the second platform was later entered. We are working on estimating this model. We note here that this model creates a version of the same sample selection issues as our current model:

Table 1: Model Parameter Estimates (GAMES)

τ	.80	
	(.57,.95)	
q	.39	.72
	(.18,.89)	(.22,.97)
Parameter	Android	iOS
Not PT, No Credentials		
α	.402	.169
	(.070,.996)	(.060,.574)
β	68.0	61.3
	(27.7,80.9)	(27.7,78.4)
K	.0024	.0020
	(.0012,.0043)	(.0009,.0040)
ω	.38	
	(.23,.59)	
No PT, Credentials		
α	.642	.413
	(.175,1.65)	(.244,1.50)
β	71.7	68.4
	(30.5,87.6)	(34.7,86.5)
K	.0052	.0056
	(.0028,.0072)	(.0032,.0073)
ω	.84	
	(.52,.95)	
PT, No Credentials		
α	.481	.187
	(.157,1.166)	(.103,.809)
β	49.1	29.9
	(3.2,58.8)	(4.6,44.5)
K	.0170	.0176
	(.0083,.0277)	(.0083,.0242)
ω	.54	
	(.30,.80)	
PT, Credentials		
α	.722	.431
	(.236,1.85)	(.242,1.70)
β	52.8	37.0
	(5.6,62.7)	(10.5,47.4)
K	.0198	.0212
	(.0094,.0303)	(.0121,.0266)
ω	.99	
	(.88,1)	

Table 2: Model Parameter Estimates (Non-GAMES)

τ	.80	
	(.57,.95)	
q	.39	.72
	(.18,.89)	(.22,.97)
Parameter	Android	iOS
Not PT, No Credentials		
α	.212	.117
	(.077,.791)	(.070,.446)
β	25.3	36.7
	(17.2,44.8)	(25.2,56.1)
K	.0033	.0023
	(.0019,.0061)	(.0013,.0039)
ω	.32	
	(0,.56)	
No PT, Credentials		
α	.452	.361
	(.176,1.55)	(.255,1.36)
β	29.0	43.8
	(19.0,49.5)	(32.5,63.2)
K	.0062	.0059
	(.0049,.0090)	(.0045,.0081)
ω	.79	
	(.34,.90)	
PT, No Credentials		
α	.292	.135
	(.121,1.01)	(.072,.689)
β	6.37	5.33
	(3.9,22.0)	(2.8,25.6)
K	.0179	.0180
	(.0098,.0284)	(.0085,.0236)
ω	.49	
	.28,.63	
PT, Credentials		
α	.532	.379
	(.200,1.59)	(.220,1.54)
β	10.1	12.4
	(6.1,25.0)	(8.1,30.1)
K	.0208	.0215
	(.0116,.0310)	(.0130,.0262)
ω	.95	
	.66,.98	

Table 3: Model Estimates of Reach Moments

Android Means		iOS Means	
Potential Reach	Realized Reach	Potential Reach	realized reach
0.0178	0.0103	0.0149	0.0100
(0.0098, 0.0350)	(0.0067, 0.0150)	(0.0092, 0.0327)	(0.0067, 0.0130)
Threshold K		Threshold K	
0.0136		0.0141	
(0.0097, 0.0151)		(0.0081, 0.01790	

we cannot treat the fact that an app was written or that it was written first for a particular platform as uncorrelated with the error in the second platform entry choice.

We could use a sample of all the apps that have ever been written on either platform. Indeed, we have this sample. There is a practical problem in gathering the Ws for this sample, since it contains over a million apps and we gather some Ws by hand. More importantly, there would still be a sample selection problem. We would need to correct for $F(S \neq \varnothing)$ rather than our model, which needs to correct for $F(T \neq \varnothing)$. There is no escaping some degree of selection, i.e., there is no way to make the list of ideas which have ever occurred to potential developers.

Similarly, we could expand our sample beyond those which have been a hit based on the criterion used here to a different criterion, such as being on the app store top lists for some length of time, or being downloaded a given number of times (feasible on the Android side if not on the iOS side), etc. However, this would merely lead to another, less symmetric, version of the sample selection problem we have.

VI. Conclusion

By modeling the platform choice of app develoopers in a split platform market, we have ruled out differentiation as an explanation for fragmentation, and found strong evidence

for multihoming incentives. We find there is much less difference in preferences across platforms than across types of developers within platforms. Our model is flexible enough to allow for the possibility that the decision to multihome and the multihoming outcome in terms of demand may be separated by costs (such as marketing) that have been identified by industry practitioners. Given our findings, it seems unlikely that the market will tip in the near future. We have identified a strong incentive to multihome on the part of the most popular apps, and this will make the fragmented market a stable equilibrium.

VII. Bibliography

Augereau, Angelique, Shane Greenstein, and Marc Rysman. 2006. "Coordination vs. Differentiation in a Standards War: 56K Modems." The RAND Journal of Economics, 37(4): 771-88.

Berry, S.T. (1992), "Estimation of a Model of Entry in the Airline Industry", Econometrica 60(4): 889-917.

Berry, S.T., and E. Tamer (2006), "Identification in Models of Oligopoly Entry", xxxx

Berry, S.T., and J. Waldfogel (1999), "Social Inefficiency in Radio Broadcasting", Rand Journal of Economics, 30(3): 397-420.

Boudreau,KJ and A Hagiu Platform rules: Multi-sided platforms as regulators

Bresnahan, T.F. and P.C. Reiss (1990), "Entry in Monopoly Markets", Review of Economic Studies 57: 57-81.

Bresnahan Davis Yin XXX

Cantillon, E. and P. Yin, "Competition Between Exchanges: Lessons from the Battle of the Bund," 2013

Church, Jeffrey, and Neil Gandal. 1992. "Net work Effects, Software Provision, and Standardization." Journal of Industrial Economics, 40(1): 85-103.

Ciliberto, F., and E. Tamer (2003), "Market Structure and Multiple Equilibria in the Airline Industry", xxx

Corts, Ken, and Mara Lederman. "Software Exclusivity and Indirect Network Effects in the US Home Video Game Industry." International Journal of Industrial Organization. 259-70.

Damiano, Ettore, and Hao Li. 2008. "Competing Matchmaking." Journal of the European Economic Association, 6(4):789-818.

Davis, J., Y. Muzyrya, and P. Yin, "Killer Apps," 2013.

Davis, P. (1997), "Spatial Competition in Retail Markets: Motion Theaters", Rand Journal of Economics.

Dubé, Jean-Pierre, K. Sudhir, Andrew Ching, Gregory S. Crawford, Michaela Draganska, Jeremy T. Fox, Wesley Hartmann, Günter J. Hitsch, V. Brian Viard, Miguel Villas-Boas "Recent Advances in Structural Econometric Modeling: Dynamics, Product Positioning and Entry" Marketing letters '05

Ellison, Glenn and Drew Fudenberg Knife-Edge or Plateau: When Do Market Models Tip?

Greenstein, Shane, and Marc Rysman. 2007. "Coordination Costs and Standard Setting: Lessons from 56K Modems." In Standards and Public Policy, ed. S. Greenstein and V. Stango. Journal of the European Economic Association, 123-59. Cambridge University Press.

Lee, Robin. 2013. "Vertical Integration and Exclusivity in Platform and Two-Sided Markets."

Mazzeo, M. (2002), "Product Choice and Oligopoly Market Structure", Rand Journal of Economics 33(2): 1-22.

Rysman, M. (2004), "Competition Between Networks: A Study of the Market for Yellow Pages", Review of Economic Studies 71:

483-512.

Rysman, Marc, The Economics of Two-Sided Markets 111(2): 353-403.

Rysman, Marc. 2007. "An Empirical Analysis of Payment Card Usage." Journal of

Industrial Economics, 55(1): 1-36.

Seim, K. (2004), "An Empirical Model of Firm Entry With Endogenous Product-Type Choices", Rand

Sinkinson, Michael Pricing and Entry Incentives with Exclusive Contracts: Evidence from Smartphones xxx

Zhu, Ting, Hongju Liu and Pradeep K. Chintagunta. 2011. "Wireless Carriers' Exclusive Handset Arrangements: An Empirical Look at the iPhone," NET Institute

working paper.