Dec 11th, 12:00 AM

Mastering Efficiency: Leveraging Multihoming Boundary Resources for Mobile App Development

Ziyi Zhao
Temple University, tuo84469@temple.edu

Aleksi Aaltonen
Temple University, aleksi@temple.edu

Mastering Efficiency: Leveraging Multihoming Boundary Resources for Mobile App Development

Short Paper

Ziyi Zhao
Temple University
Alter Hall, Philadelphia, PA, US
ziyi.zhao0001@temple.edu

Aleksi Aaltonen
Temple University
Alter Hall, Philadelphia, PA, US
aleksi@temple.edu

Abstract

Platform complementors (third-party software developers) play a critical role in enriching platform ecosystems. As app development becomes more costly and time-consuming, complementors must strategically allocate scarce resources, which includes selecting the right platforms to target and identifying appropriate boundary resources, such as software development kits (SDKs). Although complementors may aim to maximize market reach by developing apps for different platforms (a practice known as multihoming), multihoming can potentially spread resources thinly across different app versions and compromise app quality. Multihoming SDKs offer a solution by enabling app deployment across multiple platforms using a single codebase. However, this approach can compromise app quality due to insufficient platform specificity. This research examines the impact of adopting multihoming SDKs on app quality, providing theoretical insights at the intersection of technical design and platform governance. In addition, it provides practical guidance for complementors to navigate trade-offs when aligning boundary resource selection with strategic goals.

Keywords: App development, boundary resources, software development kits, multihoming

Introduction

The current platform economy is supported by more than 3 million mobile apps, which are expected to generate $526.3 billion in revenue in 2023 (Statista, 2021). These apps are created by thousands of complementors, that is, third-party software developers, who play a crucial role in platform ecosystems by bringing diverse apps to mobile platforms such as iOS and Android (Bianco et al., 2014). In the United States alone, platform complementors and their affiliates directly and indirectly employ approximately 5.9 million people, whose efforts sustain a $1.7 trillion app economy (ACT, 2020).

To develop apps, complementors use software development kits (SDKs) and related application programming interfaces (APIs), which are *boundary resource* provided by platform owners. Boundary resources provide necessary tools for complementors to extend a digital platform by creating various apps for the platform while also allowing a platform owner to control its platform. As such, boundary resources are critical to balancing between openness and control in platform ecosystems (Eaton et al., 2015; Ghazawneh & Henfridsson, 2013).

Over the years, mobile app development has become increasingly costly and time-consuming, forcing complementors to strategically allocate their resources. This entails carefully selecting which platforms to target and subsequently identifying appropriate boundary resources for app development. Complementors often intend to distribute their apps across various platforms to broaden their market reach, reduce reliance on a single platform, and to realize economies of scale by spreading technical and marketing costs over a larger user base (Droge et al., 2004; Nagesh & Caicedo, 2012; Selander et al., 2013). This approach, known as *multihoming*, can be accomplished by (1) creating parallel app versions using native SDKs on each platform, or by (2) using cross-platform SDKs, which are gaining favor among complementors for their support of multiplatform development with a single codebase. These cross-platform SDKs are defined as multihoming SDKs below. It is worth noting that using multihoming SDKs is one of the approaches to developing multihoming applications.

Although multihoming would seem to provide numerous benefits, it can also present challenges such as high asset specificity in app development, limited features, and reduced complement quality across platforms (Anderson et al., 2014; Chen et al., 2022; Ozalp et al., 2018). A particularly powerful category of cross-platform SDKs we call *multihoming SDKs* offers a potential solution to these challenges. We define a *multihoming SDK*, such as Cordova, PhoneGap, or Xamarin, as a cross-platform SDK that enables complementors to distribute an app on multiple platforms using a single codebase. By utilizing multihoming SDKs, complementors can reduce development time and maintenance costs by implementing reusable code (Amatya & Kurti, 2014; Charkaoui et al., 2014; Heitkötter et al., 2013).

However, complementors often approach multihoming SDK adoption cautiously, concerned about whether they can maintain consistent quality across different platforms (Cennamo et al., 2018). While these SDKs present opportunities for enhanced app development efficiency, faster innovation, and reduced platform owner market power, their adoption may be disadvantageous if it results in substandard complements. This could harm complementors' reputations, negating potential cost and time savings. Moreover, inconsistent user experiences and inferior complements may have negative repercussions for platform providers, leading to user dissatisfaction, increased monitoring and quality control costs, and weakened competitiveness. Consequently, the primary challenge lies in balancing the potential advantages of multihoming SDKs with the need to preserve app quality. To further explore this issue, our research examines the following question: *how does the adoption of multihoming SDKs impact app quality?*

Using a large dataset from iOS applications, we employ a staggered difference-in-differences approach combined with propensity score matching. Our aim is to contribute to the literature on boundary resources by conducting an empirical investigation into the effects of multihoming SDKs on app quality and other outcomes of interest. Our study contributes theoretically to IS research by addressing calls to examine the "intersection of technical design (software engineering view) and governance (management view)" (Tiwana et al., 2010, p. 686). Additionally, the study offers managerial insights concerning complementors' decisions to participate in multihoming activities and the complex trade-offs they face when selecting boundary resources that align with their strategic objectives.

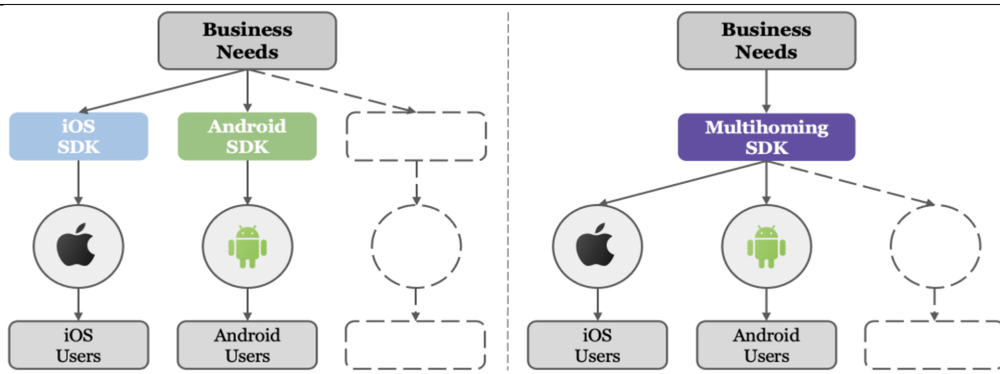

Figure 1a. Using native SDK for each platform

Figure 1b. Using multihoming SDK for multiple platforms

Figure 1. Development using (a) Native vs. (b) Multihoming SDKs

Literature Review

SDKs as Boundary Resources

Apps function as technical add-on modules to software platforms, which are defined as "extensible codebases of software-based systems that provide core functionality shared by the modules that interoperate with it and the interfaces through which they interoperate" (Tiwana et al., 2010, p.675). To fuel value creation through generativity in an ecosystem that revolves around a mobile platform, platform owners provide boundary resources such as SDKs and related APIs to platform complementors (Ghazawneh & Henfridsson, 2013). As such, platform *boundary resources* are described as “software tools and regulations that facilitate the arm's-length relationship between platform owners and complementors” (Ghazawneh & Henfridsson, 2013, p.174).

Although computer science and engineering research (Rieger & Majchrzak, 2019) have primarily focused on the functionality and efficiency of APIs and SDKs, there is a dearth of studies on the managerial implications of different types of boundary resources. Platform boundaries involve intricate issues wherein technological and design aspects intersect with economic and organizational concerns (Gawer, 2014). To this end, management scholars have introduced the concept of boundary resources to investigate the pivotal role that diverse SDKs, APIs, and other resources at the platform boundary play in determining modularity and integration, authority and governance, coordination, and value creation for the platform and its stakeholders (Bender, 2021; Bonina et al., 2021; Eaton et al., 2015; Karhu et al., 2018).

Boundary resources enable complementors to develop apps based on standardized system architecture and communication protocols (Bianco et al., 2014). From a platform owner’s standpoint, these resources are essential for striking a balance between openness and control on the platform (Ghazawneh & Henfridsson, 2013). On one hand, boundary resources facilitate the integration of complementors and provide access to core modules that enable software development on the platform. On the other hand, platform owners use them to exercise control over mobile app development, aligning actors within the ecosystem with the owners' strategic objectives (Ghazawneh & Henfridsson, 2013). In situations where diverse communities need to collaborate, but interests differ, boundary objects are especially important (Star, 2010). Boundary resources, when effectively deployed, play a critical role in mitigating the tension between innovation and oversight in digital service systems (Eaton et al., 2015).

Complementors also often aim to make their applications available across multiple platforms. However, developing apps for various platforms typically necessitates utilizing platform-specific programming languages, APIs, development environments, and app stores. This results in a wide array of skills and knowledge required to create applications targeting multiple platforms, often leading to higher development and maintenance costs (Amatya & Kurti, 2014). Multihoming SDKs offer a potential solution to this issue by enabling apps to be deployed on multiple platforms using a single codebase, eliminating the need to develop distinct versions of the same application for different platforms (Charkaoui et al., 2014; Heitkötter et al., 2013).

Multihoming SDKs

In the realm of mobile app development, a complementor that offers its app on both Google Play and the Apple App Store is regarded as engaging in multihoming (Engert et al., 2022). Early investigations into multihoming primarily adopted an economic standpoint, examining which party partakes in multihoming and who reaps the majority of economic surplus from activities (Armstrong & Wright, 2007; Rochet & Tirole, 2003). Moreover, contemporary research delves into the implications of multihoming on platform competition (Boudreau, 2008; Landsman & Stremersch, 2011). Evidence indicates that multihoming has vital implications for mobile ecosystems, as it influences the competitive dynamics among diverse platforms and the nature of interactions between complementors and platform owners (Cennamo et al., 2018).

The adoption of multihoming development has surged in popularity for various reasons. First, it enables apps to broaden their market reach by catering to users across multiple platforms (Bresnahan et al., 2015). Second, it mitigates the complementor's reliance on a singular platform and its owner, which may prove risky if platform governance and competition change in a way that is detrimental to complementors over time (Hyrynsalmi et al., 2016; Selander et al., 2013). Lastly, multihoming can yield economies of scale by dispersing technical and marketing expenses across a more extensive user base (Bresnahan et al., 2015).

Despite its potential advantages, multihoming presents significant challenges and concerns for complementors (Chen et al., 2022). The primary issue stems from the costs involved in adapting various complement versions to distinct platform architectures, resulting in potentially high asset specificity in app development (Anderson et al., 2014; Ozalp et al., 2018). Consequently, multihoming may lead to limited features and diminished complement quality across platforms, as multihoming complementors might struggle to accumulate necessary resources and manage the increased complexity associated with developing apps for multiple platforms.

To mitigate issues related to multihoming development, complementors pursuing a multihoming strategy can harness multihoming SDKs (as illustrated in Figure 1b), instead of utilizing native SDKs for each platform separately (as depicted in Figure 1a). *Multihoming SDKs*, such as Cordova, PhoneGap, or Xamarin, are defined as cross-platform SDKs that enable complementors to distribute an app on multiple platforms using a single codebase. To assist developers in making informed decisions about the SDKs best suited for their multihoming strategy, we examine the impact of multihoming SDK usage on app quality.

The Impact of SDKs on App Quality

To evaluate the quality of apps, researchers typically rely on proxy measures such as average rating (Ali et al., 2017; Mojica Ruiz et al., 2016; Siegfried et al., 2015), app store ranking (Cennamo et al., 2018; Kapoor & Agarwal, 2017; Lee & Raghu, 2014), and user reviews (Ali et al., 2017; Khalid et al., 2015). Specifically, prior research has investigated the connection between SDK characteristics and app success, using the average app rating as the dependent variable (Bavota et al., 2015; Linares-Vásquez et al., 2013; Tian et al., 2015). Aligned with this stream of research, we measure mobile app quality using their average rating on the widely recognized five-point scale in the Google Play and Apple App Store (Bender, 2021). We plan to enhance our findings in future studies by incorporating app store rankings and metrics derived from user review texts as alternative dependent variables.

The adoption of multihoming SDKs may influence app quality. The advantages of multihoming SDK adoption include accelerated development leading to reduced time-to-market, which can facilitate capturing a considerable market share and reaching a broader audience (Droge et al., 2004). Furthermore, multihoming SDKs provide the potential for additional revenue streams with minimal development costs, thanks to reusable code (Amatya & Kurti, 2014; Charkaoui et al., 2014). However, developers must also account for user expectations that complementary products be customized for each platform (Tiwana, 2015; Yoo et al., 2010). Consequently, employing multihoming SDKs may introduce various performance drawbacks, such as the absence of platform specificity, which can pose challenges in achieving consistent quality across multiple platforms (Cennamo et al., 2018). Additionally, when apps lack specialized functionality, it may prove difficult to deliver exceptional features that set the app apart from competitors on each platform (Bosch & Bosch-Sijtsema, 2010).

Research Design

Data

We collected data from a major app analytics company, encompassing diverse information about more than 2 million iOS and Android apps. The comprehensive dataset granted access to an extensive array of app-related information, such as average ratings, genre, publisher, version, price, and monthly active users (MAU). In addition, the dataset was supplemented with installation history data for over 3,000 SDKs, which recorded the app name, category, and installation date. This enables us to scrutinize the influence of adopting various SDKs on app ratings. Taking into account the heterogeneity of mobile applications, we restricted our analysis to apps with over 1,000 MAUs, ensuring that only applications with a substantial active user base were included in the results (Ali et al., 2017). Moreover, to guarantee that our dependent variable, the average rating, accurately represents the treatment effect, we excluded apps that were either too new, possessing only a handful of ratings that fail to reflect the general consensus, or too old, with a large number of ratings that may be influenced by significant events or changes in the app's lifecycle, potentially contaminating the treatment effect (Li & Hitt, 2008). We choose to use data from September 2019 to September 2020, a period devoid of any iOS updates that might confound our treatment effect. Though our primary focus is to evaluate app quality on the iOS platform, we intend to expand our analysis to include the Android platform in the future.

Methods

In our study, we solely examine the apps that installed multihoming SDKs in their version updates, rather than those that used multihoming SDKs in the initial development of the app. This approach allows us to observe the change in quality before and after the installation of multihoming SDKs. It is noteworthy that the adoption of multihoming SDKs may occur at different times for different apps. Thus, we use a staggered difference-in-differences (DID) approach, a method that has been previously used in IS research (Greenwood & Wattal, 2017). This method allows us to estimate the average treatment effect by examining treatments that occur at different times for apps in a treatment group (Boedeker, 2022). The treatment of the study is an app's installation of a multihoming SDK, so our treatment group is the apps that installed multihoming SDKs, and the control group is the apps that never installed multihoming SDKs.

To establish a control group, we employ propensity score matching (PSM), a method that facilitates the selection of a comparable control group of apps. The criteria for matching include the app's genre, primary device category, pricing model (free or paid), and app age, as proposed by Bender (2020). Furthermore, we take into account the presence of in-app subscriptions, as identified by Eaton et al. (2015), and the number of SDKs used, as described by Chen (2022) and Schilling (2000). The dependent variable of the study is the cumulative average rating of apps, which is a widely accepted measure of the overall quality of mobile applications (Cennamo et al., 2018). However, we will also collect additional data on individual ratings to better capture the dynamic nature of rating changes due to the treatment effect. The inclusion of individual rating data will provide a more nuanced understanding of how the adoption of multihoming SDKs affects the quality of mobile applications.

Preliminary Results

We assess the average treatment effect of multihoming SDK adoption on app quality using a model specified by Equation 1 (Khurana et al.,2019).

$$AppRating_{it} = \beta_0 + \beta_1 PostAdoption_t + \beta_2 Treated_i + \beta_3 \, PostAdoption_t{*}Treated_i + Controls + \varepsilon_{it} \qquad (1)$$

In the equation, the dependent variable, $AppRating_{it}$, represents the cumulative average rating of app *i* at time *t*, measured in weeks. $PostAdoption_t$ distinguishes between pre- and post-treatment periods, while the variable $Treated_i$ differentiates between treated and control groups. The coefficient of interest, β_3, captures the treatment effect. The treatment and control groups satisfy the parallel trend assumption. Based on 10,696 observations, the coefficient of $Treated_i$ is found to be -0.1194 (p-value = 0.000), indicating that the treatment group is rated lower than the control group. In addition, the coefficient of $PostAdoption_t$ is 0.0716 (p-value = 0.002), suggesting that app ratings increase over time. Ultimately, the coefficient of $PostAdoption_t{*}Treated_i$ is 0.0793 (p-value = 0.017), indicating that multihoming SDK adoption is associated with a higher average rating (refer to Table 1).

Variable	App Rating
PostAdoption	0.0716*** (0.023)
Treated	-0.1194*** (0.023)
PostAdoption*Treated	0.0793*** (0.033)
Controls	Yes
Constant	4.235*** (0.017)
Observations	10,696
R-squared	0.007
Standard errors in parentheses. *** p<0.01, ** p<0.05, * p<0.1.	

Table 1. Results of Difference-in-differences Estimation

Our initial results suggest a potential enhancement in app quality with the implementation of multihoming SDK, thus warranting deeper inquiry into the underlying mechanisms influencing the effect and possible moderating influences. For example, similar research to our study by Cennamo et al. (2018) found that multihoming games exhibit inferior performance when played on a more complex console. It is important to note that our study differs in that we examine the use of multihoming SDKs rather than the decision to multihome or not. Nonetheless, this research has inspired us to investigate whether app genre could serve as a moderating factor, thereby offering insight into the boundary conditions of the observed impact.

In addition to genre, our study endeavors to incorporate several other potential moderating factors derived from previous literature. These factors include platform policy and ecosystem complexity (Chen, 2022), prior ratings and user expectations informed by expectation confirmation theory (Ali et al., 2017; Bhattacherjee, 2001; Tian et al., 2015), launch sequence and cost (paid vs. free) (Ali et al., 2017; Cennamo et al., 2018), and developer experience (Bavota et al., 2014). By scrutinizing these factors, our study aims to offer a more holistic understanding of the impact of multihoming SDK adoption on app quality.

Discussion

In recent academic literature, the role of platform owners in governing complementors has been examined extensively (Eaton et al., 2015; Rietveld et al., 2019). However, there is a relative scarcity of research focusing on the behaviors, struggles, and decision-making processes of the complementors themselves (Agarwal & Kapoor, 2017). As application development represents a multifaceted, costly endeavor, developers frequently face challenging choices concerning the selection of diverse boundary resources in a multihoming context. Although the concept of boundary resources has gained increasing interest within the IS field since the publication of Ghazawneh and Henfridsson's (2013) seminal paper, empirical research and thorough exploration of the concept continue to be limited. Our study seeks to enrich the platform literature by delving into the quality trade-offs associated with utilizing multihoming SDKs.

Our research is distinct in that it introduces the multihoming SDKs to the IS literature accompanied by managerial implications. While the concept has been previously explored in the realms of computer science and engineering (Ohrt & Turau, 2012; Rieger & Majchrzak, 2019), it remains a relatively underexplored area in the field of IS. The present study offers valuable insights into the decision-making processes of complementors as they engage in multihoming activities, as well as the complex trade-offs they face when selecting boundary resources that align with their strategic goals. By examining these trade-offs, we aim to elucidate the challenges and opportunities complementors encounter in their pursuit of effective application development and platform integration.

Future Work

In our future research, we aim to explore various moderating factors that may affect the primary effect. Moreover, the empirical challenge of establishing the causal effect of multihoming SDK adoption remains, as the adoption may be endogenously determined through self-selection. To tackle this issue, we will employ instrumental variables to better isolate the causal effects of multihoming SDK adoption, as suggested by Wooldridge (2010). Methodologically, a two-way fixed effect approach will be utilized to account for time-varying and time-invariant unobserved factors (Boedeker, 2022). It is also important to note that different measures of app quality may yield contrasting results. For instance, Bender (2020) found that cross-platform availability positively affects app ranking but negatively impacts user satisfaction, as measured by average app rating. Consequently, we will employ additional metrics, such as app store ranking, to more comprehensively assess the impact of multihoming SDK adoption.

REFERENCES

ACT. (2020). State of the U.S. App Economy: 2020 By ACT | The App Association 7th Edition. Retrieved from https://actonline.org/wp-content/uploads/2020-App-economy-Report.pdf

Ali, M., Joorabchi, M. E., & Mesbah, A. (2017). Same App, Different App Stores: A Comparative Study. *2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, 79–90. https://doi.org/10.1109/MOBILESoft.2017.3

Amatya, S., & Kurti, A. (2014). Cross-Platform Mobile Development: Challenges and Opportunities. In V. Trajkovik & M. Anastas (Eds.), *ICT Innovations 2013* (pp. 219–229). Springer International Publishing. https://doi.org/10.1007/978-3-319-01466-1_21

Anderson, E. G., Parker, G. G., & Tan, B. (2014). Platform Performance Investment in the Presence of Network Externalities. *Information Systems Research*, *25*(1), 152–172. https://doi.org/10.1287/isre.2013.0505

Armstrong, M., & Wright, J. (2007). Two-sided Markets, Competitive Bottlenecks and Exclusive Contracts. *Economic Theory*, *32*(2), 353–380. https://doi.org/10.1007/s00199-006-0114-6

Bavota, G., Linares-Vásquez, M., Bernal-Cárdenas, C. E., Penta, M. D., Oliveto, R., & Poshyvanyk, D. (2015). The Impact of API Change- and Fault-Proneness on the User Ratings of Android Apps. *IEEE Transactions on Software Engineering*, *41*(4), 384–407. https://doi.org/10.1109/TSE.2014.2367027

Bender, B. (2021). The Impact of Integration on Application Success and Customer Satisfaction in Mobile Device Platforms. In B. Bender (Ed.), *Platform Coring on Digital Software Platforms* (pp. 79–118). Springer Fachmedien. https://doi.org/10.1007/978-3-658-34799-4_5

Bhattacherjee, A. (2001). Understanding information systems continuance: An expectation-confirmation model. *MIS quarterly*, 351-370.

Bianco, V. D., Myllarniemi, V., Komssi, M., & Raatikainen, M. (2014). The Role of Platform Boundary Resources in Software Ecosystems: A Case Study. *2014 IEEE/IFIP Conference on Software Architecture*, 11–20. https://doi.org/10.1109/WICSA.2014.41

Boedeker, P. (2022). Propensity Score Methods and Difference-in-Differences with an Exogenous Time-Varying Confounder: Evaluation of Methods. *Journal of Research on Educational Effectiveness*, 1–21. https://doi.org/10.1080/19345747.2022.2128485

Bonina, C., Koskinen, K., Eaton, B., & Gawer, A. (2021). Digital platforms for development: Foundations and research agenda. *Information Systems Journal*, *31*(6), 869–902. https://doi.org/10.1111/isj.12326

Bosch, J., & Bosch-Sijtsema, P. (2010). From integration to composition: On the impact of software product lines, global development and ecosystems. *Journal of Systems and Software*, *83*(1), 67–76. https://doi.org/10.1016/j.jss.2009.06.051

Boudreau, K. (2008). *The Boundaries of the Platform* (SSRN Scholarly Paper No. 834984). https://doi.org/10.2139/ssrn.834984

Bresnahan, T., Orsini, J., & Yin, P.-L. (2015). *Demand Heterogeneity, Inframarginal Multihoming, and Platform Market Stability: Mobile Apps*.

Cennamo, C., Ozalp, H., & Kretschmer, T. (2018). Platform Architecture and Quality Trade-offs of Multihoming Complements. *Information Systems Research*, *29*(2), 461–478. https://doi.org/10.1287/isre.2018.0779

Charkaoui, S., Adraoui, Z., & Benlahmar, E. H. (2014). Cross-platform mobile development approaches. *2014 Third IEEE International Colloquium in Information Science and Technology (CIST)*, 188–191. https://doi.org/10.1109/CIST.2014.7016616

Chen, L., Yi, J., Li, S., & Tong, T. W. (2022). Platform Governance Design in Platform Ecosystems: Implications for Complementors' Multihoming Decision. *Journal of Management*, *48*(3), 630–656. https://doi.org/10.1177/0149206320988337

Droge, C., Jayaram, J., & Vickery, S. K. (2004). The effects of internal versus external integration practices on time-based performance and overall firm performance. *Journal of Operations Management*, *22*(6), 557–573. https://doi.org/10.1016/j.jom.2004.08.001

Eaton, B., Elaluf-Calderwood, S., Sørensen, C., & Yoo, Y. (2015). Distributed Tuning of Boundary Resources: The Case of Apple's iOS Service System. *MIS Quarterly*, *39*(1), 217–244.

Engert, M., Evers, J., Hein, A., & Krcmar, H. (2022). The Engagement of Complementors and the Role of Platform Boundary Resources in e-Commerce Platform Ecosystems. *Information Systems Frontiers*, *24*(6), 2007–2025. https://doi.org/10.1007/s10796-021-10236-3

Gawer, A. (2014). Bridging differing perspectives on technological platforms: Toward an integrative framework. *Research Policy*, *43*(7), 1239–1249. https://doi.org/10.1016/j.respol.2014.03.006

Ghazawneh, A., & Henfridsson, O. (2013). Balancing platform control and external contribution in third-party development: The boundary resources model: Control and contribution in third-party development. *Information Systems Journal*, *23*(2), 173–192. https://doi.org/10.1111/j.1365-2575.2012.00406.x

Greenwood, B. N., & Wattal, S. (2017). Show Me the Way to Go Home. *MIS quarterly*, *41*(1), 163-188.

Heitkötter, H., Hanschke, S., & Majchrzak, T. A. (2013). Evaluating Cross-Platform Development Approaches for Mobile Applications. In J. Cordeiro & K.-H. Krempels (Eds.), *Web Information Systems and Technologies* (pp. 120–138). Springer. https://doi.org/10.1007/978-3-642-36608-6_8

Hyrynsalmi, S., Suominen, A., & Mäntymäki, M. (2016). The influence of developer multi-homing on competition between software ecosystems. *Journal of Systems and Software*, *111*, 119–127. https://doi.org/10.1016/j.jss.2015.08.053

Kapoor, R., & Agarwal, S. (2017). Sustaining Superior Performance in Business Ecosystems: Evidence from Application Software Developers in the iOS and Android Smartphone Ecosystems. *Organization Science*, *28*(3), 531–551. https://doi.org/10.1287/orsc.2017.1122

Karhu, K., Gustafsson, R., & Lyytinen, K. (2018). Exploiting and Defending Open Digital Platforms with Boundary Resources: Android's Five Platform Forks. *Information Systems Research*, *29*(2), 479–497. https://doi.org/10.1287/isre.2018.0786

Khalid, H., Shihab, E., Nagappan, M., & Hassan, A. E. (2015). What Do Mobile App Users Complain About? *IEEE Software*, *32*(3), 70–77. https://doi.org/10.1109/MS.2014.50

Khurana, S., Qiu, L., & Kumar, S. (2019). When a doctor knows, it shows: An empirical analysis of doctors' responses in a Q&A forum of an online healthcare portal. *Information Systems Research*, *30*(3), 872-891.

Landsman, V., & Stremersch, S. (2011). Multihoming in Two-Sided Markets: An Empirical Inquiry in the Video Game Console Industry. *Journal of Marketing*, *75*(6), 39–54. https://doi.org/10.1509/jm.09.0199

Lee, G., & Raghu, T. S. (2014). Determinants of Mobile Apps' Success: Evidence from the App Store Market. *Journal of Management Information Systems*, *31*(2), 133–169.

Li, X., & Hitt, L. M. (2008). Self-Selection and Information Role of Online Product Reviews. *Information Systems Research*, *19*(4), 456–474. https://doi.org/10.1287/isre.1070.0154

Linares-Vásquez, M., Bavota, G., Bernal-Cárdenas, C., Di Penta, M., Oliveto, R., & Poshyvanyk, D. (2013). API change and fault proneness: A threat to the success of Android apps. *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*, 477–487. https://doi.org/10.1145/2491411.2491428

Mojica Ruiz, I. J., Nagappan, M., Adams, B., Berger, T., Dienst, S., & Hassan, A. E. (2016). Examining the Rating System Used in Mobile-App Stores. *IEEE Software*, *33*(6), 86–92. https://doi.org/10.1109/MS.2015.56

Nagesh, A., & Caicedo, C. (2012). *Cross-Platform Mobile Application Development*.

Ohrt, J., & Turau, V. (2012). Cross-Platform Development Tools for Smartphone Applications. *Computer*, *45*(09), 72–79. https://doi.org/10.1109/MC.2012.121

Ozalp, H., Cennamo, C., & Gawer, A. (2018). Disruption in Platform-Based Ecosystems. *Journal of Management Studies*, *55*(7), 1203–1241. https://doi.org/10.1111/joms.12351

Rieger, C., & Majchrzak, T. A. (2019). Towards the definitive evaluation framework for cross-platform app development approaches. *Journal of Systems and Software*, *153*, 175–199. https://doi.org/10.1016/j.jss.2019.04.001

Rietveld, J., Schilling, M. A., & Bellavitis, C. (2019). Platform Strategy: Managing Ecosystem Value Through Selective Promotion of Complements. *Organization Science*, *30*(6), 1232–1251. https://doi.org/10.1287/orsc.2019.1290

Rochet, J.-C., & Tirole, J. (2003). Platform Competition in Two-sided Markets. *Journal of the European Economic Association*, *1*(4), 990–1029.

Selander, L., Henfridsson, O., & Svahn, F. (2013). Capability Search and Redeem across Digital Ecosystems. *Journal of Information Technology*, *28*(3), 183–197. https://doi.org/10.1057/jit.2013.14

Siegfried, N., Koch, O., & Benlian, A. (2015). *Drivers of App Installation Likelihood – A Conjoint Analysis of Quality Signals in Mobile Ecosystems*.

Star, S.L. (2010). This is not a boundary object: Reflections on the origin of a concept. *Science, technology, & human values*, *35*(5), 601-617.

Statista. (2021). *Revenue of mobile apps worldwide 2017-2025*. https://www.statista.com/forecasts/1262892/mobile-app-revenue-worldwide-by-segment

Tian, Y., Nagappan, M., Lo, D., & Hassan, A. E. (2015). What are the characteristics of high-rated apps? A case study on free Android Applications. *2015 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, 301–310. https://doi.org/10.1109/ICSM.2015.7332476

Tiwana, A. (2015). Evolutionary Competition in Platform Ecosystems. *Information Systems Research*, *26*(2), 266–281. https://doi.org/10.1287/isre.2015.0573

Tiwana, A., Konsynski, B., & Bush, A. A. (2010). Research Commentary—Platform Evolution: Coevolution of Platform Architecture, Governance, and Environmental Dynamics. *Information Systems Research*, *21*(4), 675–687. https://doi.org/10.1287/isre.1100.0323

Wooldridge, J. M. (2010). *Econometric Analysis of Cross Section and Panel Data, second edition*. MIT Press.

Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). Research Commentary—The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research. *Information Systems Research*, *21*(4), 724–735. https://doi.org/10.1287/isre.1100.0322