Before the
U.S. Department of Commerce
National Telecommunications and Information Administration
Washington, DC 20230

In the Matter of Developing a Report on Competition in the Mobile App Ecosystem	Docket No. 220418–0099

COMMENTS OF PUBLIC KNOWLEDGE

Public Knowledge welcomes the opportunity to offer these comments for the NTIA's forthcoming report on competition in the mobile app ecosystem. The app ecosystem is a pillar of the digital economy and a source of much innovation. Unfortunately, many of the valuable tools and business models that have made mobile devices, in general, more secure than traditional desktop computers, and that have made apps and app stores commercial successes, can also be used to harm competition. Policymakers can and should pursue policies that open up mobile devices to more competition without opening them up to bad actors and malware. Despite rhetoric from opponents of increased competition, this is more than feasible. It's necessary. The distribution and control of software on dominant platforms cannot be left in the hands of the platform vendors alone.

It is hard to fashion bright-line rules or tests in this area, in large part because technology is so malleable. One platform might offer a feature as part of an app, another might bake it into the operating system, or the hardware of a device itself. Policies designed to enhance competition should be written in a way that implementation details like this do not frustrate their aims. NTIA's report should help policymakers do this, offering not only data on the state of the market, but, as a technical, expert agency, how the state of competition for apps relates to the technical features of mobile devices, and the legitimate threats that users face from bad actors and malware.

2. Are there any important and specific entities (or categories of entities) such that it would be a mistake to omit—or improperly include—them by defining the "mobile app ecosystem" to focus on mobile devices, such as phones and tablets?

The focus on mobile devices is appropriate. Like any market definition question, the proper starting point is consumer behavior. General purpose mobile devices like smartphones and tablets do not compete in the same market as dedicated devices, like gaming consoles or e-readers, that are optimized for fewer uses (even though all these things are in a technical sense just different kinds of computer). And, while high-end tablets might be substitutable with laptops for some users, in general, tablets and smartphones are complementary to desktop and laptop computers that run "desktop" operating systems.

2b. For example, should mobile apps offered specifically for enterprise use (e.g., for use by businesses, not for consumers) be considered in this study?

4. How should web apps (browser-based) or other apps that operate on a mobile middleware layer be categorized?

(Answer to 2b and 4)

Enterprise apps that are offered directly by businesses to their employees, and that are not available for use by ordinary consumers, should not be part of this study. Enterprise apps are apps, but they are not part of the consumer app market (and may offer different functionality than is available in consumer apps).

One aspect of the app market the study should consider is the extent to which different apps in the app store use standard APIs and operate under standard terms with the platform, or whether they have negotiated special deals (e.g., a lower percentage for in-app purchases) or use private APIs or entitlements that are not available to ordinary app developers. For example, iOS developers that want to use CarPlay (integration with car infotainment systems) must request it specially from Apple; CarPlay is otherwise not available to third-party developers. Recently, Google has entered into special deals regarding app store terms with Epic Games, Spotify, and the Match Group.[1]

Web apps should only be considered to the extent that they have identical performance and capabilities as native software—according to both technical benchmarks, and in terms of user-facing features, appearance, and performance. Additionally, the discovery and management of such apps must be as easy as installing software from an app store. Absent these things, while web apps remain a valuable release valve for some kinds of software and content that is not available on app stores, they should not be considered substitutes for native software. This is for two reasons: first, web apps cannot be considered substitutes for native applications until they are truly comparable. But also, on both iOS and Android, the platform vendor controls both the primary (or only) app store, and the web browser and rendering engine that web apps require. In the case of Apple, developers have long complained that Apple limits the functionality of web apps to benefit its app store. A company should not be able to benefit from an expanded market definition that downplays its competitive significance while also taking steps to limit the true competitive potential of the web.

5. There are some indicators that there is a difference in kind between some apps that generate large amounts of money or are downloaded often and most other apps. For example, one industry analyst reported that 97% of publishers that monetize through the Apple App Store earned less than $1 million per annum in 2021, compared to other reports of more than $1 billion earned by the top 13 apps (including games) on both Apple and Google platforms.[23]

[1] https://techcrunch.com/2022/05/23/epic-games-bandcamp-temporarily-wins-right-to-use-its-own-payments-system-on-google-play; https://newsroom.spotify.com/2022-03-23/spotify-and-google-announce-user-choice-billing/

What is the best way to assess the competition environment for less popular apps and start-ups?

a. Can any potential harms, such as deficiencies in data security and privacy protections, be traced back to the current market imbalance?

b. Is there evidence to suggest that consumers are less likely to avoid or stop using a particular app even if they would prefer a more privacy enhancing environment because of a lack of competitors offering similar services?

Smaller developers can be disadvantaged due to market pressure driving the initial cost of most apps to zero, with monetization happening though ads, in-app purchases, subscriptions, and other things. Monetization strategies of these are more complex and riskier than with traditional a la carte software purchases, and users might be more comfortable with paying a subscription to a major, established company like Microsoft or Adobe, than to a new entrant.

In this market, a user might download a free app instead of a paid one, and might prefer to see ads than to pay for some rarely-used app. But that app or its adtech vendor might be harvesting personal data and selling it to data brokers, unbeknownst to the user. This creates a market pressure in favor of apps that engage in these unscrupulous behaviors. Just as the dominance of a particular app cannot always be separated from the dominance of the company that creates it, or the service it is a window to, a full assessment of competition in the app space may also have to include privacy, data security, and other factors that can have effects on app competition. A more competitive marketplace does not and should not mean a race to the bottom in terms of user privacy, and a strong nationwide privacy law would likely benefit competition and new entrants, while reducing the dominance of incumbent developers and the strength of platforms.

9. What role does interoperability play in supporting and advancing a competitive mobile app ecosystem?

a. What are the key characteristics of interoperability as it relates to the mobile app ecosystem?

b. What other barriers (e.g., legal, technical, market, pricing of interface access such as Application Programing Interfaces [APIs]) exist, if any, in fostering effective interoperability in this ecosystem? How are these barriers different or similar than those present in other ecosystems?

c. How does data portability, or lack thereof, factor into consumers keeping the same app if they switch from one operating system (iOS or Android) to another? [29]

Interoperability is a cornerstone of a pro-competition approach to technology policy. But it is a broad term, and more than meaning different things to different people (though it does), there are different, sometimes incompatible, ways to achieve interoperability even within the same market, and some forms of interoperability may come with heavy tradeoffs, others not at all. Identifying the places where and how interoperability, writ large, might benefit the mobile

app ecosystem would be a substantial undertaking. Here however Public Knowledge can offer initial thoughts regarding promising approaches and potential pitfalls.

The strongest case for interoperability in the mobile app ecosystems is 1) enabling users to switch from one app to another, and from one platform to another, and 2) enabling users to communicate between apps, and between platforms.

A lack of data portability (data that can be exported from one app or platform, and imported into another, by a user with limited technical expertise) can a barrier to interoperability. The NOI asks about barriers to users "keeping the same app" moving from one platform to another. One barrier might be that an Android app does not exist on iOS or vice versa. Or, a developer, having expended signifiant resources creating paid iOS and Android apps, might prefer a user to buy the new platform's version of app if she switches.[2]

These issues aside, and considering that many apps are free, it may be beneficial if there was a way for users to port a list of apps from one platform to another and to automatically install corresponding apps on the new platform. Even measures that merely reduce friction and hassle for users can benefit competition and aid switching. But still other barriers might remain: Not just apps are purchased, but in-app purchases and subscriptions. A developer's app on one platform might make use platform features that make moving data even to another app from the same developer difficult.

Competition between software platforms like Android and iOS is good. This means that they should be able to differentiate from each other, each adding new features and capabilities to win over users and developers. Interoperability does not require uniformity, [footnote: This is is not to downplay the important role of standards even in operating system design. See POSIX.] and it may be difficult for a user to switch from one platform to another, or for a developer to write apps for different platforms, if certain operating system features that exist on one platform do not exist on the other. Such a feature may be valuable differentiation, or anti-competitive lock-in. For example, one platform may offer audio APIs that make it easier for developers to create podcast or music apps. The other might not. Porting an app from the first platform to the second might require the developer to re-implement low-level audio processing functions that might be highly technical—and so, it might not happen. But this is not a "barrier" to competition, provided that developers are not *forced* to use features (e.g., proprietary cloud storage or authentication) that are not available on other platforms. Different platforms trying different things to attract users and developers is generally beneficial—provided that platforms remain free, as they are today, to copy good ideas from each other, and to reverse-engineer and re-implement each other's proprietary APIs and features free of baseless legal threats. *See Google v. Oracle*, 141 S.CT. 1183 (2021) (it does not infringe copyright to re-implement an API for the purpose of attracting developers already familiar with it on another platform).

[2] An iOS and Android app might be in some sense "versions" of each other but they are not "the same app," even if they look and behave the same to users. While the only barrier to accessing some kinds of content such as books and movies on different platforms and in different apps is artificial (digital rights management), apps typically need to be written specifically for, and tailored to, a particular platform.

13. Some mobile apps are pre-loaded on mobile devices or set as default apps, while others are only available through an app store, through a browser (web apps), or, for devices using the Android system, by sideloading. Is there data comparing these mechanisms and their effect on app distribution?

a. Is there a competitive advantage to being preloaded or available by default to the users of phones and tablets? What is the evidence to support or contradict there being an advantage? [30]

b. Is there data on the number of developers that have been able to have their apps preloaded or available as default apps or the types of apps?

c. What information is available on the types of agreements these developers reached and with whom to preload or set their app as a default app?

Customers expect a certain number of basic apps to be pre-loaded on their phones—and phones would be considered broken if they did not come with basic apps like a web browser, an email client, or a calendar. Yet the anti-competitive harms of preloaded and bundled apps, including from app functionality that is included directly in an operating system, are well-known. Not only is it "hard to compete with free" (even though many "free" apps carry hidden costs), but the power of defaults alone has major competitive implications.

There are many ways to balance the need for a minimally functional out-of-the-box device with the goal of ensuring competition. As a start, all default apps should be un-installable, and users should have the ability to install third-party apps as defaults. But features like this, while welcome, will never completely overcome the advantage apps have from coming pre-installed on user devices. In some cases, a dominant platform vendor pre-installing its own app, or that of a partner, maybe be so competitively harmful that it should not be allowed. It may be very fact-intensive to determine when this is the case, but one starting point is to examine which pre-installed apps (or operating system features) are truly offered to users as part of the initial sale, or not. Apps that are pre-installed on a phone that require a separate purchase or subscription (regardless of whether there was a "free trial" or some other promotion) or has some other monetization strategy, such as ads or the collection and sale of user data, are more likely to be harmful to competition, and to offer limited consumer benefit. Additionally, treating a bundled app as an additional source of revenue is a good indicator that it is a separate product that is simply tied to a different product, as opposed to a natural evolution of the minimum feature set of a mobile phone.

15. How do, or might, alternative app stores (other than Google Play or the Apple App Store), affect competition in the mobile app ecosystem?

a. What data is there to assess how well existing alternative stores distribute apps, in general or specific types of apps?

b. What unique barriers are there affecting each of the main operating systems (Android, iOS) that might prevent web apps or—to the extent allowed on Android system—alternative app stores and sideloading, from gaining more popularity with users and app developers than they currently have?

c. Is there analysis comparing competition on iOS ecosystem (where app distribution is limited) to that of alternative distribution mechanisms on Android operating systems?

Sideloading (including the sideloading of alternative app stores) is one of the best ways to increase user choice in the mobile app market. It gives users the option of installing apps that are not or cannot be made available on app stores, forces platform app stores themselves to compete and improve, and it provides a release valve for free expression and other values that are harder to quantify in a purely economic analysis.

Sideloading exists on Android, but it's a cumbersome process that, because Google does not require sideloaded apps to be cryptographically signed by developers, does carry more risk than necessary. But this specific implementation should not be taken to mean that sideloading is not important, or is risky—after all, most Windows and Mac software, to this day, is "sideloaded,"[3] not installed via an app store, and on iOS, developer-installed and enterprise apps have to be signed just as app store ones are.

16. What evidence is there to assess whether an app store model is necessary for mobile devices, instead of the general-purpose model used for desktop computing applications?

The goals of privacy, security, and competition are not at odds, and increasing competition in the app ecosystem does not require that platforms discard the valuable technical protections they have built in to devices and operating systems to protect users from bad actors, malware, and even from inadvertently misconfiguring their devices, or installing software that harms their device performance or battery life.

The security and privacy enhancements present in mobile operating systems are there in part because of lessons learned from desktop operating systems. It is good that an app, by default, cannot see the data saved by other apps, due to sandboxing. It is good that apps cannot store data and executables wherever they want on a device's storage, that apps are limited in the background tasks they perform, and that deleting an app on a mobile operating system fully deletes all of its components. It is good that apps cannot access sensitive data like photos, contacts, or location information without express user opt-in.

App stores, too, are beneficial, in that they make finding and installing software much easier for ordinary users. The policy problem is not that these things exist, but that they are abused for business reasons, or in the case of app stores, monopolized.

[3] Like "acoustic guitar," "sideloading" is a term that did not need to exist until a newer alternative was invented. Just as acoustic guitars used to just be "guitars," and "sideloading" software used to just be called "installing" software.

Policymakers and those interested in increasing mobile app competition can do so without compromising privacy and security. This requires not only expertise in how markets function, but about the threats users face from bad actors and malware, how mobile devices and operating systems protect against them, and how platforms have addressed similar challenges before.

18. Are there other areas, specific technologies or procedures, that offer lessons on more and less successful ways to screen out problematic apps? What are the characteristics of such success?

a. Are there good examples by enterprise users? [35]

b. For example, some devices allow sideloading only after warning the user to make sure they trust the app before proceeding with the download, in a way similar to how some browsers issue warnings for unknown websites. What material exists about the efficacy of such methods?

c. What roles, if any, do independent or third party security testing play in the app store ecosystem?

d. Does the current model discourage competition and innovation in the development or advancement of security testing?

Sideloaded apps are not inherently more risky than app store apps from a technical perspective. They can be subject to the same security protections, sandboxing, and permission models that app store apps are subject to. These technical security features are the first line of defense, protecting users, and they are not related to app stores or app store policies.

The primary difference between app store apps, and sideloaded apps is one of policy. App stores can reject apps that engage in certain behaviors that it is hard to put a stop to via technical means, and the abuse of certain kinds of APIs can best be addressed via policy choices. For example, many of the same technical capabilities that are useful for parental control services can also be used for "stalkerware." But there are ways of addressing these issues besides anticompetitively preventing users from accessing apps from sources other than platform app stores, and different app stores might enforce policy restrictions of this kind better than the first-party app store itself does. For example, in a more competitive app market, a third-party store might specialize in apps that highly value user privacy, dedicated more resources to vetting these apps for policy compliance, and even putting into place stronger policies.

Of course, policymakers and regulators around the world have recognized that app store "policies" can be anticompetitive—policy, not technological limitations, is all that stands in front of app developers and a more competitive billing market, for instance.

Public Knowledge believes that users should be able to sideload apps if they so choose, and that sideloaded apps can have the same security protections as app store apps (including sandboxing and code-signing). Informing users that sideloaded apps are not subject to policy

review may be appropriate, but it would be inappropriate for a platform to imply that sideloaded apps are riskier from a technical perspective, or to imply that they have greater access to personal data than app store apps.

24. Some apps make use, or would like to make use, of additional mobile device components beyond those that are more commonly accessible (e.g., camera, microphone, contacts) in order to offer an innovative product or service, but the operating system or device provider does not allow such access.[36]

Similarly, for some apps, it might be essential to be able to interconnect to other hardware and services, such as cloud services. What are the valid security concerns and technical limitations on what device functionality an app can access?

a. What factors should be considered in striking a balance between encouraging companies to ensure proper security measures, while allowing third parties to access the protected features that might allow for further innovation and competition?

b. Are there specific unnecessary (e.g., technical) constraints placed on this ability of app developers to make use of device capabilities, whether by device-makers, service providers or operating system providers, that impact competition?

c. Are there other means or factors to consider for mitigating specific risks that would not inhibit competition?

In general, third-party app developers should have the same access to the same user-facing capabilities as the platform vendor's first-party apps, via supported, secure, and well-documented APIs, whether they are hardware or software features. This is the case whether a platform vendor simply gives its own apps capabilities not available to other apps, or has incorporated features directly into the operating system.

As an example, some time ago, Apple made it more difficult for apps to constantly monitor a user's location in the background. This was a beneficial move that benefits user privacy, and even national security. But Apple exempted itself from this rule. It did not do this by giving Apple Maps or another user-facing app special permissions not available to third parties. In fact, Apple Maps (the app) has no special location privileges at all. It doesn't need them, since Apple baked location-tracking into the operating system itself, and it is this tracking information that Apple uses to improve Apple Maps (the service). Apple claims it does this in a privacy-preserving way. In a context like this, the best policy solution would not be to require Apple to allow apps to indiscriminately track user location, but to develop an API that allows other apps or services to make use of location information in the same privacy-preserving way that Apple does. Ensuring a fair marketplace for apps requires addressing this sort of self-preferencing.[4]

Some developers might want more than being able to match the consumer-facing features of a platform or its apps. For example, a developer might want more granular location tracking

[4] *See* John Bergmayer, Tending the Garden (2020) at 49-50, for more on this example.

than Apple itself performs. But a "level playing field" does not depend on an app having the same access to low-level hardware features (here, mobile networking data and GPS) that Apple necessarily can access in its role as the operating system vendor. It is the role of an OS to mediate between apps and a device, but a policy that required platforms to grant third-party apps the same level of hardware access that the operating itself has would be likely be harmful to user security and ease of use, with minimal benefit to competition beyond alternative approaches.

27. What specific measures might the federal government take to foster healthy competition—especially for nascent app innovation—in the mobile app ecosystem?

In the paper Tending the Garden (2020), Public Knowledge put forth a number of policy proposals that, if adopted, would promote app competition and user rights. These include: Permitting sideloading, limiting in-app-purchase requirements to app functionality, ensuring that developers can truthfully communicate with their customers about app store policies, allowing users to set and change default apps, limiting pre-installed apps to essentials, app store transparency, limiting the use of proprietary developer data by platforms, offering secure APIs proactively to new hardware and software features, due process, increased business model flexibility, and more.

Respectfully submitted,

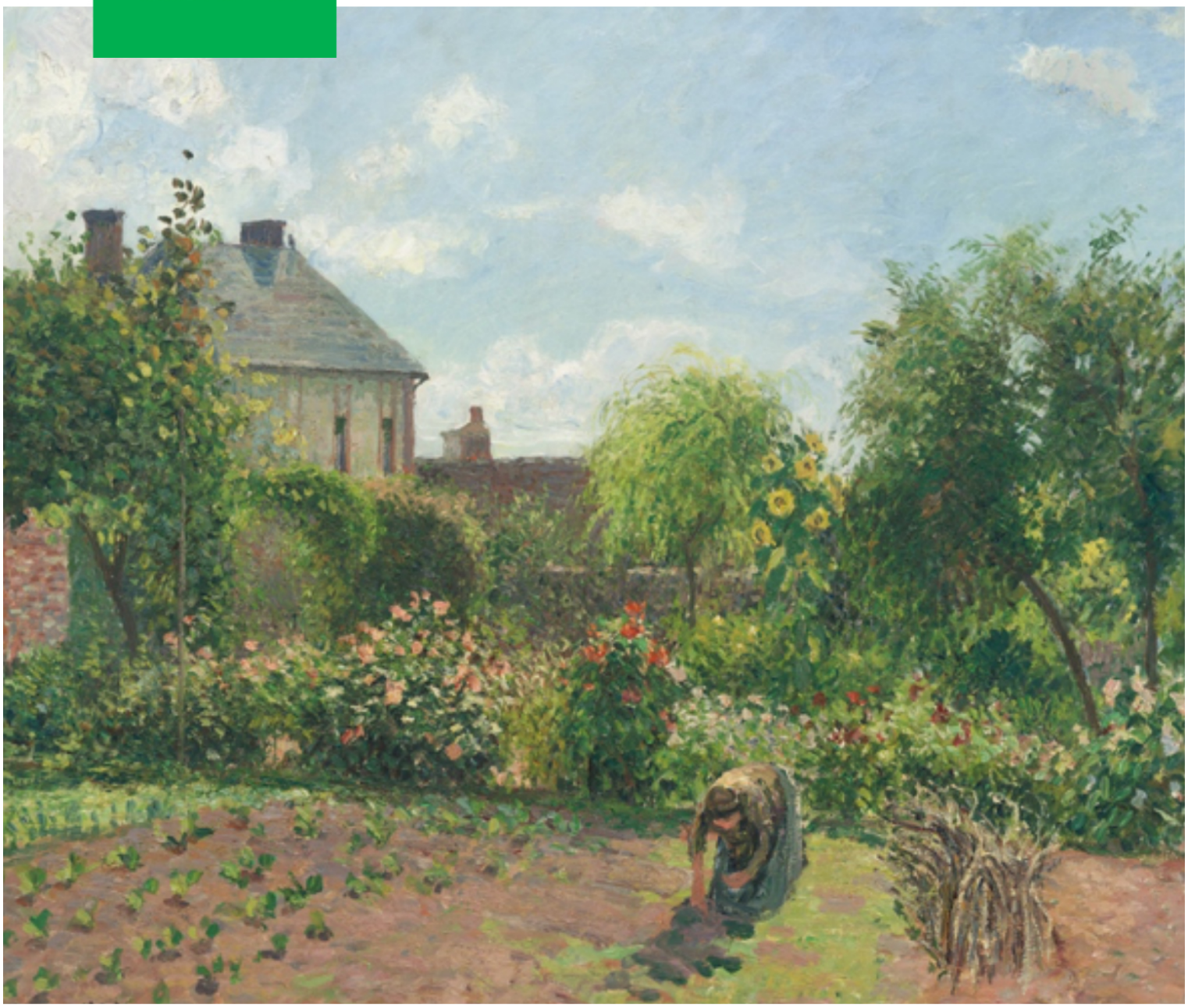
John Bergmayer
Legal Director
PUBLIC KNOWLEDGE
1818 N St. NW
Suite 410
Washington, DC 20782

May 23, 2022

PUBLIC KNOWLEDGE

Tending the Garden:

How to Ensure That App Stores Put Users First



June 2020

John Bergmayer

Acknowledgements

The author would like to thank those that provided feedback during the drafting of this paper, including Blake Reid at the University of Colorado Law School, Hal Singer, managing director at Econ One and an adjunct professor at Georgetown's McDonough School of Business, and Will Jennings, student at the Indiana University McKinney School of Law and intern at Public Knowledge, for editing assistance. This paper, along with other work from Public Knowledge on platform competition, was made possible by the support of the Omidyar Network.

Table of Contents

Executive Summary

App stores provide security, privacy, and trust for users, while giving platform maintainers significant gatekeeper control over the software that users can access, what that software can do, and how it can be monetized. This gatekeeper control can be used to benefit platforms at the expense of independent software developers as well as users. Switching costs, network effects, and other factors mean that competition between platforms for users and developers cannot be enough to ensure that app stores and their associated software platforms will be operated in a way that promotes consumer rights, the public interest, and broader economic benefits. This paper suggests specific measures that should be implemented by dominant app stores to promote these interests—reducing the gatekeeper control that app stores inherently have, but not eliminating it. These measures are suggested as baseline structural remedies that would apply broadly, and do not fully displace the need for a competition law framework, an individualized, complaint-driven procedure that addresses matters these measures do not address, or other remedies.

Specifically, this paper recommends that platforms allow users and developers to bypass the app store entirely through side-loading, but only subject to strict code-signing requirements. Code-signing ensures that only software from known developers can run on a device. At the same time, this paper suggests that code-signing authorities themselves can be decentralized. It also recommends a few measures that app stores can implement to reduce the advantage their first-party apps have over competitors, such as allowing users to change defaults, and proactively providing third-party application programming interfaces, or APIs, for major new features at a more rapid cadence. This paper also calls for app stores to allow greater business model flexibility to developers, such as allowing things like paid upgrades and not requiring the use of in-app purchase systems for media purchases and subscriptions. It calls for due process for developers to ensure consistency in the application of rules. Finally, even older versions of dominant platforms and software that run on them can be of historical and technological interest. Platforms should, therefore, ensure that it remains possible to archive and emulate software that may still be protected by copyright but is of limited commercial significance.

These measures are not costless to implement. Dominant platform maintainers (the companies that operate the app stores and their associated software platforms) would have to spend time and money, hire extra staff, and add more internal processes to carry them out. This give-and-take is how the nature of regulation of private enterprises "of public consequence" necessarily works.[1]

[1] *See Munn v. Illinois*, 94 U.S. 113 (1876) ("Property does become clothed with a public interest when used in a manner to make it of public consequence, and affect the community at large. When,

App stores that follow these practices can still be successful, self-funding business endeavors—that is, not merely operated as a loss leader for hardware sales or service upselling. Indeed, the broader positive economic effects promoted by these measures may indirectly benefit the platforms as well. In this paper's final analysis, even the recommendations aimed at benefitting developers should be considered consumer protection measures, as the aim of this paper is to advocate that users have more control, more ability to access content, and access to a more competitive software market than an unregulated, dominant software platform is likely to provide absent intervention.

Introduction

App stores—along with broadband internet access, cloud services, and the like—are part of the infrastructure of the internet and of the digital economy. Much of how people now engage in commerce, look up information, communicate with family and friends, find dates, work, and entertain themselves is mediated through apps. While apps and app stores exist on various computing platforms—smart TVs, desktop PCs, and game consoles, for instance—they are of central importance to smartphones and tablets. Due to platform design decisions as well as technological constraints, functions that could be done via a web browser on the relatively uncontrolled open web on a desktop PC or a laptop, often must be mediated through an app on mobile devices. This is not necessarily a bad thing—so-called "native" apps can offer features, performance, and security unavailable to web apps—but it brings into focus the power that app stores, the software platforms they run on, and the companies that control them (platform maintainers), have over the lives and activities of users, and the livelihoods of the developers who need access to app stores to reach users. (This paper uses the term "platform maintainer" rather than "platform owner" or some other term to avoid suggesting that the company that develops the platform owns or should be credited with the economic and social value of a platform. Users and third-party developers are equally, if not more responsible; the world's greatest operating system, app store, and devices are nothing without them.)

---

therefore, one devotes his property to a use in which the public has an interest, he, in effect, grants to the public an interest in that use, and must submit to be controlled by the public for the common good, to the extent of the interest he has thus created.") To be as clear as possible: the regulation of dominant platforms does not imply that platforms are somehow at "fault" for being dominant. It is not punishment, it is not a question of "fairness," and regulation of this kind does not have to be justified in terms of antitrust or other forms of competition law. It is a response to their importance to the public interest and the economy, an importance their smaller rivals may lack, which justifies imposing extra regulatory requirements on dominant platforms and dominant platforms alone. One consequence of this is that smaller rivals may be permitted to engage in activities that dominant platforms cannot. If the consequence of this is that smaller rivals gain a relative advantage, then this is a pro-competitive side effect of regulation, but not the intended consequence.

Broadly defined, the "app economy" has been variously estimated to be worth $950 billion (as of 2018, domestically),[2] $6.3 trillion (globally, by 2021),[3] and $568 billion.[4] These different numbers include different things—direct app sales and in-app purchases, purchases of physical and in-person goods and services through apps, the value of advertising, the value of devices controlled through apps, the jobs directly created by apps, and the economic ripple effects created by all these things. For this paper, it is enough to say that the economic impact of app stores is large, and their role in the economy is important.

The same features that make app stores compelling for users and (at times) the developers of third-party apps also give the companies who run them an unprecedented level of control over both what users can do with their devices, and developer livelihoods. App stores are one component of software platforms,[5] particularly mobile software platforms, that have become essential infrastructure enabling commerce, communication, and culture. Dominant app stores and their associated software platforms deserve scrutiny because decisions taken by platform maintainers like Apple and Google can harm competition more broadly. Moreover, absent some intervention, these platform maintainers may not have the incentive to invest in areas of the platform that create positive externalities that ripple through the economy, but that they cannot directly capture. Whatever the estimate of the "app economy" you pick, platform maintainers shouldn't capture that full value for themselves. (Similarly, broadband providers shouldn't try to capture the full value of commerce their infrastructure enables, and the maintainers of toll roads do not deserve a percentage of the profits or salary of the people who drive on their roads.)

This paper's recommendations are meant to benefit users and independent developers directly. Another way of looking at them, however, is as designed to ensure that platform maintainers operate in a way that maximizes the overall positive economic effects of app stores. This is why this paper is concerned with "dominant" app stores—they are the ones that, due to their popularity or the markets they serve, have wide-ranging economic effects, and affect the most users.

---

[2] The App Association, *State of the App Economy (6th Ed.)*, (2018), https://actonline.org/wp-content/uploads/ACT_2018-State-of-the-App-Economy-Report_4.pdf.
[3] App Annie, *The State of Mobile 2020*, (January 15, 2020), https://www.appannie.com/en/go/state-of-mobile-2020; *see also* Sarah Perez, *App Economy to Grow to $6.3 Trillion in 2021*, TECHCRUNCH (June 27, 2017), https://techcrunch.com/2017/06/27/app-economy-to-grow-to-6-3-trillion-in-2021-user-base-to-nearly-double-to-6-3-billion.
[4] Deloitte, *The App Economy in the United States*, (August 20, 2018), https://www.ftc.gov/system/files/documents/public_comments/2018/08/ftc-2018-0048-d-0121-155299.pdf.
[5] Some of the recommendations of this paper directly concern app store policies, such as the kinds of apps allowed in the store, what those apps can do, and how they can be sold. Others may require changes to the underlying software platform itself as a prerequisite before such apps can be carried in the store, or even relate to integration with specific hardware features.

The recommendations of this paper are not all costless for platform maintainers to implement. Some may require hiring extra staff, taking more time to roll out features, or even signaling to the marketplace planned changes. They may interfere with monetization opportunities or make it more difficult for a platform maintainer to enter new lines of business while opening opportunities for smaller competitors who do. (They are, however, designed to preserve the main security benefits of modern secure computing platforms.) Nevertheless, the broader social and economic effects of following these recommendations for dominant platforms suggests they're worth it. The public benefits outweigh the private costs, without making app stores unprofitable to operate.

While this paper does not dwell on whether platform maintainers should voluntarily follow these practices or somehow be required to, Public Knowledge has extensively argued for a digital platform regulator that would have the legal authority to require that platforms adopt specific policies,[6] as well as enact structural changes to platform marketplaces as a whole. It is also worth briefly mentioning the connection of these issues to antitrust, especially given heightened attention to Apple's highly restrictive policies around in-app purchases, which will be discussed at length below. The EU has opened an investigation into Apple's practices in this area,[7] and Representative Cicilline, Chairman of the House Subcommittee on Antitrust, which is currently investigating tech industry practices generally, has called Apple's practices "highway robbery."[8] The question of antitrust is therefore unavoidable, and in many areas, there seems to be a strong *prima facie* antitrust case that specific actions by Apple or Google in terms of how they manage their platforms are unlawful. Nevertheless, antitrust is unlikely to be the best tool to bring about most of these paper's recommendations. First, antitrust is often narrowly focused on specific anticompetitive actions, as opposed to market structure and incentives. In the case of private antitrust suits, any relief might be targeted just to one company, and any policy changes that result might only affect that company, or companies in an identical situation. Second, many of the harms this paper seeks to address—for example, issues around free expression and cultural archiving—are simply not competition harms that can be addressed by antitrust at all. Still, antitrust investigations by public authorities are likely to bring to light facts about the marketplace, differential treatment given to some developers, and other matters that could be generally informative to regulatory and policy efforts of all kinds. Further, need for a comprehensive, regulatory approach should not dissuade litigants from bringing cases where the facts merit it.

---

[6] Harold Feld, THE CASE FOR THE DIGITAL PLATFORM ACT, (May 2019), https://www.publicknowledge.org/assets/uploads/documents/Case_for_the_Digital_Platform_Act_Harold_Feld_2019.pdf

[7] European Commission, *Antitrust: Commission Opens Investigations Into Apple's App Store Rules*, Press Release (June 16, 2020), https://ec.europa.eu/commission/presscorner/detail/en/ip_20_1073.

[8] Nilay Patel, *Apple's App Store Fees Are 'Highway Robbery,' Says House Antitrust Committee Chair*, THE VERGE (June 18, 2020), https://www.theverge.com/2020/6/18/21295778/apple-app-store-hey-email-fees-policies-antitrust-wwdc-2020.

Thus, the best way to contextualize this paper's recommendations is as the specific requirements sector-specific platform regulator, applicable to platform maintainers that operate dominant app stores. That said, smaller platforms should feel free to follow them if they so choose—all of the recommendations, in the end, benefit users (users, of course, are the ultimate beneficiaries even of pro-developer changes). Additionally, it may be that specific recommendations are closer to baseline consumer protection measures that every platform maintainer, however small, should follow.[9]

While theoretical considerations of this kind motivate this paper, its purpose is not to provide an economic, political, or legal theory of infrastructure, competition, or dominance. Rather, it is to address some of the technical details of app stores that platform maintainers—and potentially, regulators—may have to consider in this area. Its fundamental thesis is that app stores and controlled software platforms are in general beneficial, and that their benefits to user security and confidence should not be thrown out in favor of a vision of "openness" that, in practice, puts ordinary users at risk. Nevertheless, the loss of openness has downsides. The challenge then is to manage the disadvantages of greater control by the platform maintainer without creating problems for security or privacy (or even device stability, battery life, and performance). After discussing some of the techniques and policies that platform maintainers deploy to ensure control, this paper will review some of the problems that app stores can create before suggesting ways to address them.

* * *

App stores have empowered users by allowing them to confidently install software from a trusted source with the knowledge they can just as easily uninstall it, and that it won't damage their device. This is a significant improvement from the days when ordinary users had no way of knowing whether a given piece of software was a legitimate application or a data-harvesting, spam-sending piece of malware. Those experiences often resulted in users avoiding installing software on their computers altogether, or leaving it to experts to decide what was safe.

App stores have benefitted many developers, too, providing them with a large addressable market, a distribution and payments infrastructure, a community of developers, tools and documentation, and other services. They have given rise to the "app economy," and many successful and sustainable small businesses, as well as many headline-grabbing billion-dollar ones, would not have been possible without the reach and services that app stores provide.

---

[9] While privacy rules are beyond the scope of this paper, they offer a good example of rules that digital services of all kind should be required to follow regardless of whether they are "dominant" in the marketplace. Rules against deceptive user interface techniques ("dark patterns") may be in the same category.

But app stores have well-known downsides. The very features that allow platform owners to protect their users also enable them to restrict developers. The processes designed to keep malicious apps off of the store can keep apps that pose a competitive threat to the platform owner off the store, as well. The same policy might protect user privacy and inhibit competition. It is not always clear at the outset whether it is possible to protect user privacy in ways that do not have competitive side-effects, or even how one is to compare harms to competition to offsetting, but incommensurable benefits.

Criticism of app stores has been ardent, but often contradictory, and often centers on whether the platform exercises too much control or not enough.

App stores have taken upon themselves the role of curating the material they carry. As a result, people blame them when bad stuff slips through, or argue that Apple or Google fails to adequately prevent harmful apps, or apps from businesses with harmful business models, from being listed in its app store. For the Washington Post, Geoffrey Fowler wrote, "You might assume you can count on Apple to sweat all the privacy details. After all, it touted in a recent ad, 'What happens on your iPhone stays on your iPhone.' My investigation suggests otherwise." His investigation showed that popular iOS apps, including "Microsoft OneDrive, Intuit's Mint, Nike, Spotify, The Washington Post and IBM's the Weather Channel" were sending personal data to the cloud. He called for Apple to do more to prevent apps on its store from doing these things, though stated that "The result shouldn't be to increase Apple's power."[10] Joanna Stern of the Wall Street Journal has also shown how popular apps, including kid's apps, have questionable privacy practices, writing that "My app-tracking adventures show that more transparency and stronger protections are needed."[11] Offering similar but more extensive criticisms, Ian Bogost has observed,

> Companies such as Google and Facebook get access to iPhone users by offering their apps—Messenger, Gmail, Google Maps, and so on—for download from the Apple App Store. Most cost consumers nothing, because they exist to trade software services, such as email or mapping, for data. That business model helped stimulate the data-privacy dystopia we now occupy.

[10] Geoffrey A. Fowler, *It's the Middle of the Night. Do You Know Who Your iPhone is Talking To?*, WASHINGTON POST (May 28, 2019), https://www.washingtonpost.com/technology/2019/05/28/its-middle-night-do-you-know-who-your-iphone-is-talking.
[11] Joanna Stern, *iPhone Privacy is Broken…and Apps Are to Blame*, WALL STREET JOURNAL (May 31, 2019), https://www.wsj.com/articles/iphone-privacy-is-brokenand-apps-are-to-blame-1155931640.

Arguing that "Apple reaps huge profits selling the glass rectangles on which the more invasive apps run,"[12] he has called for Apple to do more to restrict these apps from gathering user data. Indeed calls for Apple (or another platform maintainer) to remove an app or category of apps from the app store are relatively common. Mark Cuban even called for Apple to remove Twitter from its app store.[13]

App stores have also created an easy way for developers to reach users. Still, the low barriers to entry lead to a saturated market, and downward pressure on prices, which app stores fail to alleviate by offering features such as free trials and upgrade pricing. Additionally, app store features like in-app purchasing may have been introduced with the best of intentions but often used as part of addictive, casino-like game mechanics, such as loot boxes[14] and pay-to-win designs.

Another line of criticism is uneasy with the idea of app stores themselves, and the control they afford platform owners—the ability to privilege some apps over others, the ability to impose cultural or moral norms on users, and the ability to extract revenue from apps the platform maintainers did not themselves create. David Weinberger once called the Apple app store, "the seductive angel of death for computing. It enables Apple to keep quality up and, more important, to keep support costs down. But a computer that can't be programmed except by its manufacturer (or with the permission of its manufacturer) isn't a real computer."[15] Less poetically, but along the same lines, Jonathan Zittrain has argued that the iOS app store is "much worse" than the anticompetitive behavior that Microsoft was sued over in the 1990s, calling for users to abandon closed platforms for open ones in a spirit of "hewing to the original spirit of the PC."[16]

App stores are not wholly "good" or "bad," but they can be operated well or poorly. Some tradeoffs are inevitable: a fully "open" software platform for everyone is risky from a privacy and security perspective, while a closed one would lack the third-party energy and creativity that generates new ideas, drives the platform forward, and benefits users. Rather, after discussing some of the issues that have arisen, the paper will suggest specific ways that these tradeoffs can be *managed* in ways that

---

[12] Ian Bogost, *Apple's Empty Grandstanding About Privacy*, THE ATLANTIC (January 31, 2019), https://www.theatlantic.com/technology/archive/2019/01/apples-hypocritical-defense-data-privacy/581680.
[13] Hope King, *Mark Cuban Says Apple Should Ban Twitter*, CNN (December 10, 2015), https://money.cnn.com/2015/12/10/technology/mark-cuban-apple-twitter-criticism/index.html.
[14] Responding to criticism, Apple now requires that games disclose the odds of obtaining particular rewards. Chaim Gartenberg, *Apple Now Requires Games with Loot Boxes to Disclose Odds*, THE VERGE (December 21, 2017), https://www.theverge.com/2017/12/21/16805674/apple-loot-box-app-store-games-odds-probability-disclosure.
[15] Jonathan Zittrain, *Tales of Technolust: the appStoreless Droid*, JoHo the Blog (October 18, 2019), https://web.archive.org/web/20100415155852/https://www.hyperorg.com/blogger/2009/10/18/tales-of-technolust-the-appstoreless-droid/.
[16] Jonathan Zittrain, *The Personal Computer is Dead*, MIT Technology Review (November 30, 2011), https://www.technologyreview.com/2011/11/30/189455/the-personal-computer-is-dead/.

maintain the benefits of both security and control, as well as competition and dynamism.

It is undeniable that platform maintainers—and Apple will be the primary but not sole focus of this paper[17]—have the incentive to manage these issues themselves, since they benefit both from a platform that users trust and can rely on, and from a vibrant ecosystem that includes many third-party developers. Still, a platform maintainer can sometimes err on the side of self-preferencing, or even in favor of making policy choices that are easier to administer or cheaper to implement. When a platform such as iOS or Android achieves a certain amount of marketplace dominance, the platform maintainer in its role as a gatekeeper may fail, at times, to make the socially optimum tradeoff or one that benefits users. When switching from one platform to another carries high costs due to a user's software investment or exclusive services, and when a platform's value is enhanced via various forms of network effect, normal market forces alone are not enough to discipline a platform's behavior.

Any software platform might raise questions of consumer protection, user rights, or competition. However, *dominant* platforms in particular, if managed correctly, can create positive externalities that ripple through the economy that the platform maintainer might not (and should not) be able to capture for itself. While these positive externalities are one of the chief benefits of well-maintained infrastructure, platform maintainers may lack the incentives to maximize them. Again, while various policy proposals exist that discuss precisely *how* to encourage or require a platform owner to make various changes (e.g., through antitrust, or sector-specific regulation), this paper will focus on what those changes should be.

Examples of App Stores

The app stores that this paper will primarily address are the dominant and default ones on the two major mobile platforms: Apple's App Store on iOS, and Google's Play Store on Android. They are the app stores that control access to mobile devices, which are the most indispensable general-purpose computing devices in most people's lives, and the two platforms that most easily meet the standards of "dominance."

[17] Globally, the market share of iOS is much smaller than that of Android. iOS has greater market share in the US overall, but is dominant in the premium segment of the smartphone market, and in tablets. Additionally, Apple simply has a more controlling approach to its platform than other platform maintainers like Google and Microsoft so it simply offers more fodder for discussion. Being more "controlling" in this context is not necessarily a bad thing; this approach also means that in many ways Apple protects user security and privacy better than the competition. But its policies are stricter in many senses and thus deserve more consideration.

As Public Knowledge has discussed before, "dominance" is a complex question, but determining whether or not a platform is dominant can be guided by looking at the "cost of exclusion" from a platform—a measure of what costs a business, organization, or individual would bear from being excluded from the platform.[18] A platform where being excluded carries a high cost would most likely be considered dominant.

Additionally, dominance may correlate with but is not identical to "market power." By way of analogy, when the Financial Stability Oversight Council (FSOC), which identifies risks and responds to emerging threats to financial stability, considers whether a bank is "systematically important," its inquiry is not limited to market share, but its structural role.[19] Market power itself must be considered both from the perspective of users and developers. From the user's perspective, what matters is not just whether a consumer can choose an Android phone or an iPhone, but whether, once invested in a particular platform, it is easy to switch. A user switching from Android to iOS not only must purchase a new device but also might lose access to a library of purchased software or full integration with Google Assistant. Conversely, a user switching from iOS to Android would lose access to iMessage and other Apple services as well as a library of software.

But even if users could seamlessly switch between numerous platforms at a moment's notice, developers still have no choice but to meet users where they are. Other factors, such as the extent of bottleneck control, barriers to entry, the nature of the user base, and industry structure, also contribute to a finding of dominance. As with other aspects of proper implementation, the exact legal mechanisms by which an app store (or any other platform) would be found to be dominant is beyond the scope of this paper, which is more concerned with what the consequences of such a finding should be.

Apple and Google stores have significant policy differences. For example, Apple is more stringent in its app review standards, while iOS devices do not allow "sideloading," or the installation of apps from other sources than the official app store; Android devices do. But, Android devices are not just controlled by Google—while Google does have contractual relationships with phone manufacturers that ship devices that can access the Google Play Store and other Google services, manufacturers still have some leeway, and can include their own apps and even app stores on devices in addition to Google apps, and this can raise competitive concerns. There are also policy differences with respect to in-app purchase requirements and other matters. That being said, the stores are broadly similar in

---

[18] Harold Feld, THE CASE FOR THE DIGITAL PLATFORM ACT, (May 2019), https://www.publicknowledge.org/assets/uploads/documents/Case_for_the_Digital_Platform_Act_Harold_Feld_2019.pdf.
[19] John Bergmayer, EVEN UNDER KIND MASTERS, (May 2018), https://www.publicknowledge.org/assets/uploads/blog/Even_Under_Kind_Masters.pdf

most ways, and even though, unlike on iOS, the Play Store is not the only way for most users to load software on their devices, as a practical matter being listed on the Play Store is vital to most developers.

However, these two are not the only app stores. Apple also has an app store for the Mac, for instance. Installing software on the Mac is still possible from other sources, though through user interface choices, Apple is increasingly making it difficult for users to install *unsigned* software—a technical point (discussed in more detail below) that provides security benefits, but is not without its own potential pitfalls. Similarly, Microsoft has an app store for the Windows platform, and Amazon has one for its Fire tablets (and the ability to purchase "skills" for the Alexa voice platform). While examples from these app stores may be illustrative to show what different practices are possible, the Amazon Fire platform for tablets, and the Mac and Windows app stores (which are lightly used, compared with traditional software installation methods) are not likely to be found dominant.

Modern video game console platforms all include app stores. While it is (for now) still possible to buy a video game on disc, or memory card, increasingly games are distributed as digital downloads, and if anything the policies of video game consoles are much stricter than found anywhere else. Console platforms have more rigorous quality, content, and performance requirements than Apple or Google, and they also approve all disc- and card-based software before it's distributed. However, because of the limited range of software that these app stores distribute, and because their business model is consistent with the decades-old model for the distribution of console games, they appear to be less controversial. From a computer science perspective, a video game console is just a computer. But modern microwaves and cars include powerful (by historical standards, at least) computers as well. Video game consoles are different, of course, in that they are platforms that run third-party software, and because video games are an important means of cultural expression. Nevertheless, the existence of more open, general-purpose platforms like the PC and modern mobile devices, and the competitive nature of the game console market, slightly lessens the need for the same level of openness from dedicated devices with a limited purposes, and factors into the "cost of exclusion" factor in a dominance analysis. Thus, without disregarding that video game consoles (as well as cars and microwaves) can carry with them competition and consumer protection problems of their own, this paper will largely concern itself with general-purpose devices.[20]

TV app stores pose a similar issue as video game consoles. Modern TVs (and TV-connected devices such as the Apple TV or a Roku) are full-fledged computers, integrated with various app stores. However, the predominant apps for TVs are for

[20] It is also the case that video game consoles are often sold at a subsidized price, or at cost, with the expectation that software licensing fees will make up the difference. The same is not true of most other devices, such as phones, where relatively few users pay for third-party apps at all.

streaming media, and to a lesser extent, games—the occasional productivity app like James Thompson's PCalc[21] for Apple TV notwithstanding. There is likely a stronger case for applying the recommendations of this paper to TV devices than to game consoles but owning to their relatively narrow use cases, and because it is not clear which, if any, TV platforms are dominant, this paper will not discuss them in-depth.

On desktop platforms—which are typically the most open of all major consumer computing platforms—several third-party app stores thrive. The most successful is Steam, which has almost become the default way to distribute PC games. Users are rarely forced to go through Steam to obtain games, and PCs can have multiple game stores installed simultaneously, as well as supporting installing software directly from a developer (what would be called "sideloading" on mobile devices). Yet developers still choose to distribute through Steam, even though it takes a 30 percent cut of sales,[22] because users demand it. This is a useful demonstration of how, while it may be true that some app stores abuse their gatekeeper position, many users still prefer to rely on them despite having alternatives, and that users can benefit developers by generating and aggregating demand. (Other popular third-party app stores on PC platforms include the Epic Games store and GOG ("Good Old Games"), which offers titles free of DRM (digital rights management).) For reasons similar to the issues with game consoles, and because these third-party app stores, by definition, lack the same level of gatekeeper control as app stores that are integrated and run by the platform vendor itself, these app stores are beyond the scope of this paper's recommendations, while serving as examples of how the policies of the dominant mobile app stores are not set in stone.

App stores vs security architectures

App stores are typically associated with a series of security and control measures that, strictly speaking, are conceptually separate. That is, it is possible to have an app store without any security measures. And it is possible to implement most security measures without an app store—that is, security measures are platform features, or features of the operating system or device, more than "app store" features. Much of this paper will be concerned with Apple's iOS App Store and platform for a number of reasons, as it presents an example of an app store and platform that implement all of the security and control measures discussed here. In discussions of Apple's App Store, the distinction between what counts as a platform restriction and an app store restriction may be overlooked. But, for clarity, this section will briefly describe the various security and control measures that an app

[21] James Thomson, *About PCalc for Apple TV*, https://pcalc.com/tv/index.html.
[22] Tom Marks, *Report: Steam's 30% Cut Is Actually the Industry Standard*, IGN (October 7, 2019, updated January 13, 2020), https://www.ign.com/articles/2019/10/07/report-steams-30-cut-is-actually-the-industry-standard.

store and platform may implement, and which Apple's platform does implement, before providing examples of app stores that implement only some or none of them.

Exclusive source of software

An app store may be, practically speaking, the only way to install software on a platform, which gives it competitive importance and makes it a key gatekeeper—if a developer can't reach the app store, it can't reach users, and users look exclusively to the app store to install software.[23]

An app store doesn't have to be completely airtight as a source of software for it to have effective gatekeeper power.

For example, on iOS it is always possible to run web apps. In fact, after the iPhone was announced, but before the App Store was, web apps were the only way to run third-party software on the iPhone. Steve Jobs called this a "pretty sweet solution." Apple commentator John Gruber called it a "shit sandwich," instead. This is because web apps are widely viewed as inferior to "native" software, which is[24] faster, more capable, and behaves in a more predictable manner due to using system-standard controls and features. Since that time, web apps have become significantly more capable. Apple has also been accused of slow-rolling the implementation of the new web standards that allow its Safari browser, while not allowing browsers with competing web rendering engines into its App Store.[25]

---

[23] The app developer Rogue Amoeba has noted that this gatekeeper power makes it difficult for developers to publicly air grievances with how app stores function. Rogue Amoeba, https://twitter.com/RogueAmoeba/status/1273637152685985795 (June 18, 2020) ("Unfortunately, shipping iOS software means being on the iOS App Store. Because of this, many developers are scared to speak out.")

[24] Web apps are typically run in the browser, though they can be "installed" to a user's home screen as well, and newer "progressive web apps" can offer features previously restricted to native apps (and can raise similar security and privacy considerations). Web apps can also be simply packaged up as "native" apps and released into the app store. Mark Zuckerberg once called using web technology rather than native code for its app Facebook's "biggest mistake," Christina Warren, *Zuckerberg's Biggest Mistake? 'Betting on HTML5'*, (September 11, 2012), https://mashable.com/2012/09/11/html5-biggest-mistake.

[25] While third-party browsers are allowed in the app store, they must use Safari's WebKit rendering engine, which is available system-wide to apps that want to display web pages. So, while Firefox, Chrome, and other browsers can compete on the basis of UI enhancements, and can sync with their desktop counterparts, they cannot compete with Safari in terms of implementing new web standards more quickly, or JavaScript performance. In fact, until iOS 8, third-party apps could not even use the same JavaScript engine that Safari used, and were stuck with a slower version, due to apparent security considerations that Apple proved able to work around. Finally, even with those caveats, third-party browsers still cannot be set as system defaults, meaning that links from apps will still default to opening in Safari instead of what a user's preference might be.

Web apps aren't the only alternative to the App Store. iOS developers, for instance, can install their own apps on their devices after compiling them from source code. In fact, an iOS developer can download, compile, and install any app for which the source code is available, including apps in categories that Apple does not allow in the iOS app store at all. While various tools try to automate this procedure for less advanced users, it is in general a technically daunting process requiring detailed knowledge of software development processes. And there are other reasons why this is hardly a loophole that undermines Apple's gatekeeper control: Without a paid developer account (which costs $100 per year), apps installed in this way expire after a few days and need to be re-compiled and re-installed. That might be fine for a student or a hobbyist to test an in-progress app, but it is hardly a convenient alternative to the app store.

Finally, it is possible to install software outside of the app store by using an "enterprise certificate." This is a capability offered to businesses that want to make job-specific apps that don't necessarily belong in a public app store—apps that concern inventory management, reserving company resources, and internal communications, for instance. Apple's control over these certificates is not fool-proof. Facebook and Google were found to be using them to distribute apps to the public that would not be allowed in the Apple App Store, due to how they intrusively monitored user activity; Apple subsequently temporarily revoked both company's certificates, not only disabling the customer-monitoring apps but also internal iOS apps used by both companies. Various grey-market companies also use enterprise certificates in an unauthorized way to sell users apps in categories not allowed in the app store (e.g., BitTorrent clients and game emulators) as well as pirated software.

Code-signing

Modern cryptography often uses a pair of keys, public and private. With this sort of public-key cryptography, anyone can encrypt a message to a desired recipient using the public key, which can be freely shared, but only that recipient can read it by using the private key, which is not shared.

Code-signing is a variant of this design. It allows a sender to cryptographically "sign" a message—that is, attach a sequence of characters and numbers to it—that can only be generated with the private key, but that can be verified as having come from a particular sender with the public key.[26] The exact methods vary, but what they have in common is that if even one character of the message is changed, removed, or deleted, the signature verification will fail. Code-signing thus provides

[26] The "message" that is signed in the case of code-signing is simply the computer code of a given software program instead of "Attack at dawn" but this makes no fundamental difference—computers store everything as numbers and the same kinds of cryptographic techniques can be used on any kind of data.

a way to verify that a given piece of computer code is exactly what a developer shipped and hasn't been changed or added to by malware or other sources.

By itself, code-signing doesn't do anything besides provide a method of verification. But if the platform also enforces rules related to code-signing—namely, that all code must be signed, and that unsigned code will not be run or will be subject to greater restrictions, then it can be quite powerful, both as a means to protect users and as a method of establishing and strengthening the control of platform maintainers.[27]

Notably, code-signing not only allows a user's device to verify the integrity of computer code, but also enables the platform maintainer, which typically issues developer certificates, to revoke software that has already shipped—meaning that already-installed software on user devices will no longer launch.[28] This can be useful if a developer's app is found to be harmful after it's available to the public. It is also another powerful tool in a platform maintainer's arsenal that could potentially be misused, or that the platform owner can be compelled to use at the request of a government.

Sandboxing and permissions

Sandboxing refers to techniques at the operating system level that ensure apps only have access to certain data and features— i.e., they have to "play in the sandbox." It might be implemented in many ways, and it might entirely prohibit apps from certain behaviors and certain kinds of access, or grant them that access in a controlled way (e.g., after user consent or via a trusted, system-provided intermediary process known as a "powerbox").[29]

Take the example of a user's contact database. Early versions of iOS let any app access this information, which follows the standard set by desktop operating systems such as Mac OS and Windows. However, users began to see this capability as a privacy violation, as some apps would access a user's contacts database and

---

[27] In the early 2000s Microsoft worked on a trusted computing platform "that could make PCs resistant to tampering even by those who have physical access and control. The initiative would go under a variety of names, including Palladium, TCPA, and the Next-Generation Secure Computing Base." *See* Timothy B. Lee, *How four Microsoft engineers proved that the "darknet" would defeat DRM,* (November 24, 2017), https://arstechnica.com/tech-policy/2017/11/how-four-microsoft-engineers-proved-copy-protection-would-fail/. These efforts provoked serious backlash, and were in many ways ahead of their time. One difference between secure computing efforts like this and code-signing is that code-signing is typically enforced by the operating system; more comprehensive efforts are enforced at a hardware level, and in the case of PCs, could prevent the installation of alternative operating systems (e.g., Linux instead of Microsoft Windows). Modern smartphones and tablets do often implement security technologies of this kind. However, discussion of the extent to which secure hardware platforms limit competition at the operating system level.

[28] Apple's implementation of this used to require that it revoke all of a developer's apps at once, but its newer implementation, which is called "app notarization," allows it to revoke only specific apps.

[29] C2 Wiki, "PowerBox", http://wiki.c2.com/?PowerBox.

upload it to a server, for instance, to spam people with messages asking them to join a new social network. Subsequently, Apple began requiring apps to ask for permission to access contacts when they needed to do so (and preventing apps that do not truly need access to contact information from refusing to run until such consent is granted.)

Another example is document storage. Again, historically, desktop operating systems allowed all apps to read and write files from common storage areas in folders named things like "My Documents" or the Desktop. But in today's era of increased security consciousness, even this can cause problems. For example, if an app that has access to sensitive-use data stores it in such a location, other apps might access it without user consent. Platforms can design around this by denying apps access to such common storage areas, requiring explicit user permission before being granted such access, or by having such access being granted only during explicit user interaction and mediated by a "system-provided dialog." In other words, users can choose to open a file in an app, but that does not give the app unrestricted background access to the folder it was stored in).

Other features that may be "sandboxed" include camera and location access, the ability to view connected Bluetooth devices or Wi-Fi networks, other device sensors, and microphones, calendars, and photo library access.

Sandboxing can be incompatible with many kinds of apps—for example, an app that makes a complete backup of a user's computer or device may need unrestricted levels of system access. It can also cause competitive issues if first-party apps from a platform owner are not sandboxed, and thus have greater or more seamless capabilities than third-party apps, which are.

API restrictions

Software platforms offer "application programming interfaces" (APIs) which provide common functionality, preventing developers from having to write their own code to handle solved problems, or for apps to include third-party libraries. APIs can also offer functionality that can only be accessed through the API—for example, access to device sensors, personal data, or location information. Additionally, some APIs are "private" which means that third-party applications cannot use them. Undoubtedly restricting access to certain potentially sensitive functions can be good for security.[30] However, as with other points of control, the way that access to certain APIs is gated can raise competitive concerns, as first-party applications

[30] For instance, Uber was calling private APIs to access device serial numbers, so that it could track users even after they had uninstalled and reinstalled its app. The API that allowed this was not intended to be accessed by third-party developers and Apple required that Uber stop the practice. *See* Thomas Claburn, *Apple Frees a Few Private APIs, Makes Them Public*, THE REGISTER (June 13, 2017), https://www.theregister.co.uk/2017/06/13/apple_inches_toward_openness.

might have access to a functionality that third-party apps cannot replicate. For example, until iOS 8, only Apple's own apps could use the superior Nitro JavaScript engine, meaning that web views, and third-party browsers on iOS, were at a performance disadvantage relative to Apple's own Safari. On other computing platforms, developers were always free to use private APIs at their own risk. However, because they were not officially supported and documented, the platform might change how they work at any time without notice, breaking apps that rely on them. Whatever the merits of this *caveat emptor* approach, the app store review process adds yet another way to discourage the use of private APIs: namely, scanning submitted applications to see if they make calls to these APIs, and rejecting them if they do.

DRM

While code-signing ensures that a particular app is the actual code released by a particular developer, DRM ties a copy of an app to a particular user. Short for "Digital Rights Management," DRM can be controversial (for example, when it locks ebooks to particular services and devices like the Amazon Kindle, creating an obstacle to users switching devices or even shopping around for a new source to access ebooks). Even so, many developers approve of DRM as they view it as a way to reduce piracy. In the context of modern devices with app stores and other methods of control, DRM is just one piece of the puzzle, but it serves critical technical functions, such as preventing apps without proper DRM from running on a device, and preventing users from moving even a fully code-signed app from one device to another.

Benefits of App Stores

App stores have benefits and drawbacks, and a primary purpose of this paper is to propose remedies for those drawbacks. But it is important to review the ways that app stores have been good for developers and for users—a narrow focus only on the drawbacks of app stores could lead one to reasonably ask why we should have them at all. But even with their various drawbacks, app stores have likely done more good than harm, and the recommendations of this paper are not only designed to promote the interests of users and independent developers, but to ensure that app stores continue to be a net benefit.

Perhaps the most vital feature that app stores provide users is trust.[31] Before home internet use was ubiquitous, and for some time thereafter, those relatively few

[31] F. Cuadrado and J. C. Dueñas, Mobile Application Stores: Success Factors, Existing Approaches, and Future Developments, IEEE COMMUNICATIONS MAGAZINE, vol. 50, no. 11, pp. 160-167, (November 2012) ("We believe that consumer trust towards the individual applications from the platform is influenced by

homes that had PCs would (apart from the technically adept few who used bulletin board systems and the like) typically buy a PC in a box from a reputable retailer. Retailers served a number of important functions, such as curating and selecting software of reasonable quality. This reduced a user's search costs—consumers could just look through the software on the shelf, rather than researching every offering on the market—and perhaps ask store clerks and other experts what they recommended. And the buyers could be reasonably sure that the software they purchased from such retail establishments worked, more or less, and would not damage their machines or be malicious in other ways. This trust between consumers and home-PC retailers enabled the widespread adoption of one of the most important communication devices we've ever seen.

The advent of ubiquitous home internet access combined with the ability to install software directly from the web would seem to be another boon for both developers and users. No more gatekeepers! No more price markups! But the reality was far more mixed, since without the architecture of trust and selection that retailers provided, most users were at a loss as to what software was actually worth installing. Only enthusiasts would have the inclination to do the necessary research to figure out what software to install and what to avoid. It was worse than that, even, since nefarious users took advantage of the situation to release actively harmful software. Our ability to disseminate computer code far outpaced security practices that keep people, data, and computers safe.

The result of this wasn't just that people were confused as to what software to install, but that one group of ordinary users was often too afraid to install software at all, while another group would install software willy-nilly. (This latter group being inflated, perhaps, by the poor browser security architectures prevalent at the time, whereby simply visiting a web page and clicking a dialog box could result in the installation of fraudulent "antivirus" software, harmful browser toolbars, and so on.)

Web apps offered one solution to this problem by enabling people to use different software and services without "installing" anything at all—their computers became, in effect, terminals interacting with distant servers, along the lines of old time-sharing systems. (The ability to "install" a web app so that it can work offline and even interact with local data came later.)

App stores offered the other solution. By the time Apple launched the iOS App Store, app stores (and related ideas, like package managers in open source

---

the approval process, and consequently the willingness of users to pay."), https://eecs.qmul.ac.uk/~fcuadrado/papers/commag12-appstore.pdf; *see also* S. Hyrynsalmi, M. Seppänen, & A. Suominen, *Sources of Value in Application Ecosystems*, JOURNAL OF SYSTEMS AND SOFTWARE, 96, 61-72, (2014), http://tutcris.tut.fi/portal/files/3591220/hyrynsalmi_sources_of_value_in_application_ecosystems.pdf.

operating systems) had been around for a number of years, and commercial app stores, like Valve's Steam, a very popular game store, had already achieved some success. And while even app stores on mobile platforms had pre-Apple precursors, it was Apple's App Store that gave the model for mainstream success, with ordinary users installing general-purpose software on a popular platform.

Apple's App Store became popular, in part, because there was no other way to install software on iPhones or distribute software to them. The iPhone was a popular platform and the App Store was popular because of that. But the success of optional, third-party app stores like Steam does show that the model has some inherent benefits, and that given the choice between buying software from an app store, and buying it directly from a developer, many users prefer the intermediation of the app store. The various benefits that app stores provide can help explain this.

Like traditional retail stores, app stores perform a curation and trust function. Because shelf space is not limited with digital app stores, curation becomes more a matter of editorial promotion, top-selling lists, user reviews, and recommendations, but the basic function is the same. App stores also provide trust—apps must be approved before being listed in the store, in most app stores by a human reviewer, and on all of them software is subject to batteries of tests to ensure compatibility, security, functionality, accessibility, and other issues.

App stores give users a place to search for software to install that is more structured than just searching the web. Because there is typically just one major app store for a platform (though not always), it gives users just one place to search—not great for competition, to be sure, but it does at least mean that if a developer is listed in that app store, a user is likely to encounter its app. This reach can benefit developers—if you're listed in the app store, you can be sure that you can reach the entire addressable market.

App stores also provide various services to developers and users. App stores install updates to software, ensuring that users have the latest features and security patches. They provide payment services, both for paid apps and in-app purchases. Deloitte categorizes some of the benefits of app stores as follows:

- *ubiquity in user interface/user experience features,*
- *a secure platform to promote their products,*
- *storage systems for hosting apps and managing downloads,*
- *billing service,*
- *a payment management system (micropayments) which makes it easy for mobile app developers to recover sales revenue.*[32]

[32] Deloitte, THE APP ECONOMY IN THE UNITED STATES, (August 20, 2018), Pg. 8, https://www.ftc.gov/system/files/documents/public_comments/2018/08/ftc-2018-0048-d-0121-155299.pdf.

It points out that these features generally reduce transaction costs—that is, they make it easier for developers to connect to users. This function—along with security and curation, and the concomitant increase in user trust—can be viewed as the primary benefits of app stores.

There are a few other benefits of app stores as well. Apps distributed via app stores may also have access to system APIs that might not otherwise be available; for instance, for a time iCloud synchronization services were only available to Mac apps distributed through the Mac app store.[33] (This paper will argue later that one solution to competition issues created by app stores without compromising security is to extend this idea of differential access to certain system features.) Finally, it should not be overlooked that app stores prevent piracy. While DRM can cause as many or more problems than it solves (particularly in the area of media), and while it is not foolproof, developers that choose to sell software on a commercial basis appreciate that platform enforcement of copy protection technology reduces unauthorized copying.

Drawbacks of App Stores

The same control and curation that make app stores an appealing method of software distribution, given the nature of computer security, privacy concerns, and other matters, also give the companies that control app stores and their associated platforms a tremendous degree of control over how users can interact with their devices, and whether and how developers can reach users. This control can be abused, and even if not abused, can lead to negative consequences. The most salient of these are harms to competition (and relatedly, constraint of developer business models), and harms to free expression. This section will discuss these in broad terms, and this paper will later detail a few specific instances that could be remedied with app store or platform policy changes.

It is hard to compete with a platform

One fundamental competitive concern is that it is very difficult to compete with a platform's own first-party apps. This manifests in a number of ways.

The power of defaults

The "power of defaults," or the default effect,[34] is a well-known phenomenon where default choices, options, and software settings will be what the majority of users

[33] While certain Android APIs are only available on Google-approved devices that carry the Play Store, sideloaded Android apps on official devices are not restricted in the APIs they can access.
[34] Wikipedia, "Default effect", https://en.wikipedia.org/wiki/Default_effect.

stick with, even if they can freely change them or choose a competitive alternative. Awareness of the power of defaults underlies many policy debates in the technology space; e.g., whether particular privacy regimes should be opt-in or opt-out. It also explains why a platform maintainer (or, in some cases on Android, a phone manufacturer) has such power—it sets the defaults, and can structure them in ways it finds advantageous. This plays out a number of ways. First, a platform's own apps may be preinstalled on user devices, and thus bypass the app store entirely. Users may not even bother to look for alternatives, or even be aware that they can. Even if first-party apps are downloaded through an app store, they may have other advantages: the user may be prompted to download them, they may get priority promotion within the store, or they may rank higher in app store search results. For example, the Wall Street Journal found that Apple's own apps routinely rank above those in app store search results.[35] While many developers attribute this to poor design rather than anticompetitive intent, the result can be the same.[36]

An extreme example of this is known as "Sherlocking"—when a platform maintainer takes a concept from a third-party app and creates its own similar app, or incorporates the functionality into the system itself. (The term derives from an instance where version 3 of Apple's Sherlock program replicated the functionality of a third-party app called Holmes.[37]) In economic terms, it has been "suggested that platform owners can bundle their own complementary application with their platforms to foreclose competitors' access to their customers and profitably capture the whole of their markets."[38] Even when a platform buys third-party software, the effect can be the same on competitors—for example, when Apple bought SoundJam and turned it into iTunes, the popular app Audion was driven out of the market.[39] Particularly when a platform maintainer bundles its own application, or uses data it has access through platform or app store metrics to decide which products or features to copy, this can be anticompetitive. Indeed, some developers have complained that Apple uses them for "market research"—using its visibility into what third-party apps are popular on its platforms to help decide what features it should introduce itself.[40]

---

[35] Tripp Mickle, *Apple Dominates App Store Search Results, Thwarting Competitors*, WALL STREET JOURNAL (July 23, 2019), https://www.wsj.com/articles/apple-dominates-app-store-search-results-thwarting-competitors-11563897221.
[36] John Gruber, *Malice or Incompetence: On the Rankings of Apple's Own Apps in App Store Results*, DARING FIREBALL (September 12, 2019), https://daringfireball.net/2019/09/on_the_rankings_of_apples_own_apps_in_app_store_results
[37] *You've Been Sherlocked,* THE ECONOMIST (July 13, 2012), https://www.economist.com/babbage/2012/07/13/youve-been-sherlocked.
[38] Fung Zhu, *Friends or Foes? Examining Platform Owners' Entry into Complementors' Spaces*, J. Econ. Manag. Strat. 23, 24 (Spring 2019).
[39] Cabel Sasser, *The True Story of Audion,* (November 11, 2004), http://panic.com/extras/audionstory.
[40] William Gallagher, *Developers Talk About Being 'Sherlocked' as Apple Uses Them 'For Market Research'*, APPLE INSIDER (June 6, 2019), https://appleinsider.com/articles/19/06/06/developers-talk-about-being-sherlocked-as-apple-uses-them-for-market-research.

At the same time, "ideas" are not protectable by intellectual property rights (copyrights protect specific expression and patents protect specific implementations; neither protect general concepts) and it is beneficial when good ideas are copied widely. Platforms even copy good ideas from each other—for example, the Macintosh operating system was heavily inspired by the Xerox Alto, and in turn influenced Microsoft Windows—to, in turn, copy Windows features such as Command-Tab app switching (Alt-Tab on Windows)[41] and contextual "right click" menus. Additionally it makes sense to distinguish between a platform maintainer entering an adjacent or complementary market (such as an operating system vendor creating media streaming services and bundling them), versus a platform maintainer improving the platform itself, even in instances where the idea for the improvement was first implemented by a third party.

For example, hundreds of app developers hit on the idea of "flashlight apps" before iOS or Android included a feature that activated a phone's camera flash for that purpose. But it is difficult to see flashlight apps as a true independent market worthy of public policy concern. This applies broadly to "system utility" apps and, more generally, implies that when analyzing competition in platforms, it is important to distinguish a platform leveraging its position to capture an adjacent market from a platform improving its core product. Even when certain ideas are first introduced by third parties, the history of technology, and consumer products more generally, shows that some features should be included in the core product itself.[42]

Thus, while this paper argues that dominant platform maintainers should institute guardrails to protect independent developers, particularly when it comes to a platform maintainer selling or bundling standalone apps and giving its own apps and services technical or business model advantages not available to third parties, it does not recommend a complete ban on platform maintainers integrating new features into their system. Promoting competition does not requiring freezing technology products in time.

To be sure, distinguishing a bundled app from a feature of the system itself may be difficult, and is necessarily imprecise endeavor that requires knowledge of broader market and technological trends as well as user expectations. The analysis may

---

[41] Prior to Apple directly incorporating this feature, it was available from a third-party application called LiteSwitch X. *See* Ammon Skidmore's Old Apps, https://sysbeep.com.

[42] Another example is how a third-party utility, Growl, first introduced a system-wide standard notification system for the Mac. Notifications are now seen as a basic feature of computing platforms. Further examples are almost endless. Cars did not initially ship with radios, now they do. Modems used to be aftermarket add-ons to computers, then were bundled, with their successors eventually integrated directly into a computer's motherboard. On mobile devices, taking panoramic photos used to require a third-party app, now this is a build-in camera mode. Image search used to be performed by specialized search engines, and now major search engines all have an image tab. It would not benefit users to prohibit these kinds of developments.

even change over time. Microsoft bundling its Internet Explorer browser with Windows as widely seen as anticompetitive in the 1990s. Now, a computing device that does not include a browser would be considered broken, since web browsing is a primary purpose of computing devices, and the web may be the only way to install software onto a desktop computer. (The changes that have occurred since the 1990s may be most profoundly illustrated by the fact that while Microsoft still bundles a browser with Windows, Google's Chrome, which has to be installed by users, nevertheless has much higher usage share, showing that user preference can still overcome the power of defaults.)

The advantage that first-party apps may have on a system can even be as mundane as the platform reserving basic, generic terms to itself, naming its music app "Music," its notes app "Notes," or its browser, "Internet" (the name for Samsung's browser)—words that, among other things, users are more likely to search for and associate with their basic function.

First-party apps may also be set as defaults, even when third-party apps are installed, with no way to change this setting. Thus, even if a user would prefer to use Outlook or Gmail as their email client, on iOS, "mailto" links open in Apple Mail, and even if a user would prefer to use Firefox or Brave, web links open in Safari.

First-party apps exempt from platform rules and given special privileges

Similarly, first-party apps may simply play by different rules and have different access to technical features of the system than third-party apps available on the app store. Apple's own apps can take advantage of private system APIs that third-party apps cannot.[43] There are good reasons for third-party developers not to use private APIs; namely, they can change without notice, rendering an app non-functional. Public, documented APIs are typically only changed with ample notice to developers who depend on them.[44] Private APIs may also have access to information that may be kept from third-party apps for privacy reasons, such as a device's serial number. At the same time, when key, useful functionality is only available through a private API, or not at all, third-party developers can be put at a disadvantage—the fundamental disparity is not the restriction on the use of private APIs but that certain functionality is only available through them.

First-party apps may even have default access to information that, for third-party apps, must be affirmatively granted. As discussed more at length below, Apple's

---

[43] Michael Mimoso, *Apple to Remove 256 iOS Apps Using Private API's, Collecting Personal Data*, THREATPOST (October 19, 2015), https://threatpost.com/apple-to-remove-256-ios-apps-using-private-apis-collecting-personal-data/115098.
[44] *See* John Gruber, *Private*, DARING FIREBALL (December 22, 2018), https://daringfireball.net/2008/12/private.

first-party apps and services have access to system-level location information that third-party apps must ask for.

First-party apps may also be able to access certain sensors or data without first asking for permission, or may be the only apps that can access certain things at all. For example, until Apple released iOS 13, only Apple apps could use an iPhone's near-field communication (NFC) chip, which is used for such functions as contactless payments.[45] A similar scenario is currently playing out with the iPhone's new ultra-wideband chip, which only Apple apps can access.[46]

A platform maintainer can better plan for platform changes

Additionally, when system features change from one OS version to the next, third-party apps may suddenly become incompatible or exhibit new bugs, putting them at a major disadvantage until developers update them to address the changes. In the meantime, platform-provided apps tend to be updated along with the OS itself, and the platform company's own developers have access to the future platform roadmap in ways that outside developers do not.

One extreme example of the difficulty third-party apps can have when competing with first-party apps is from early in the iOS App Store history, when Apple forbade third-party apps from "duplicating the functionality" of first-party apps.[47] While Apple eventually eased off this policy, it shows the power and control that app stores have.

---

[45] Blue Bite, iPhone NFC Compatibility, (April 16, 2020), https://www.bluebite.com/nfc/iphone-nfc-compatibility.

[46] Nick Statt, *Airdrop on the iPhone 11 Will Let You Point at People to Share Photos*, THE VERGE (September 10, 2019), https://www.theverge.com/2019/9/10/20859550/apple-iphone-11-pro-airdrop-u1-locator-chip-tag-tile-bluetooth-tracking. Among other things, UWB allows devices to more accurately gauge the position and distance of other UWB devices, and is harder to spoof. While macOS is not the specific topic of this paper, it is notable that on the Mac App Store, Apple has exempted its own apps from sandboxing rules that third-party apps were required to follow. For some time, its own iWork apps were available on the Mac App Store, but not sandboxed. *See Michael Tsai, iLife, iWork, and the Sandbox*, MICHAEL TSAI BLOG (October 30, 2013), https://mjtsai.com/blog/2013/10/30/ilife-iwork-and-the-sandbox/. XCode, Apple's development environment for iOS and Mac apps, is distributed through the Mac App Store and is not sandboxed to this day.

[47] Charles Arthur, DIGITAL WARS 186 (Kogan Page 2012) ("Apple began behaving in odd ways: it would ban apps for seemingly random reasons, such as 'duplicating the functionality of existing iPhone apps'. Mail programs, browsers, podcast downloaders – those would be rejected. Developers howled, and howled even more loudly at being told they couldn't howl publicly because of Apple's non-disclosure agreement on rejections." *See, e.g.*, Angelo DiNardi, *MailWrangler and the Apple App Store* (September 20, 2008) https://angelo.dinardi.name/2008/09/20/mailwrangler-and-the-apple-app-store.

A platform maintainer doesn't need to make money on apps or services

Finally, platforms don't need to make money on any of their apps at all, at least not their basic apps. Apple, for instance, makes money selling iPhones, not licenses to its Notes app. The purpose of the Notes app is simply to enhance the value of the iPhone. This paper does not suggest that it is anticompetitive for platforms to provide basic apps of this kind for free, as devices should have basic out-of-the-box functionality. But competing with free is obviously a challenge for any developer.

App stores place business model constraints on developers

App store requirements and policies can also place competitive constraints on developer business models.

The sales commission reduces developer revenue

The most straightforward of these is the often mandatory commission that app stores charge developers for paid apps, which reduces developer revenue. Most app stores—Apple's, Google's, and Steam, for instance—take around 30% of the list price.[48] Although this is a lower rate than what physical retailers typically charge for boxed software, it's still much higher than the rates charged by pure payment processors and pure storage providers. App stores do provide value beyond those simple services, of course, but in many cases, developers do not have the choice of whether to avail themselves of that value. On the Mac, for instance, developers can choose whether the distribution, exposure, update mechanisms, subscription systems, and promotion that come with distributing on the Mac App Store is worth it or not. Many apps find that it is. Many do not. In this instance, this has forced Apple to improve the Mac app store while also ensuring that apps distributed outside of it have some measure of security and are not locked out of platform features like iCloud sync. However, iOS provides no alternative. On Android, the other major mobile platform, it is possible to distribute apps outside of the app store, but it is a somewhat complex process that most users will only undertake for the most compelling of apps (e.g. the popular game, Fortnite).

Restrictions on free software prevent users from accessing some apps

"Free software" is released under a license such as the GNU General Public License (GPL) that allows software to be redistributed and modified freely, but only on the condition that the application's source code be made publicly available.[49] (As the

---

[48] Tom Marks, *Report: Steam's 30% Cut Is Actually the Industry Standard*, IGN (October 7, 2019, updated January 13, 2020), https://www.ign.com/articles/2019/10/07/report-steams-30-cut-is-actually-the-industry-standard.

[49] "Free software" is a more restrictive category than software that is merely "open source," which may not face the same difficulties with app store distribution.

saying goes, it is free as in speech, not free as in beer, though it is often both.) Of course, the sole author or rightsholder of a piece of free software can release that software on any app store, since a rightsholder is not bound by the terms of a license it issues to others. But a developer that incorporates a piece of free software into its own app, or even a person that wants to distribute an existing free software project on an app store, may find that doing so would violate the terms of the license for any number of reasons. The problem can be particularly complex with free software projects where individual volunteer developers retain the rights to their individual contributions.[50] The DRM that app stores may require may itself violate the terms of a free software license that requires the software to be redistributable, for instance, or it may not be possible to distribute or make source code available using app store tools.

A lack of paid upgrades removes a traditional source of recurring revenue

Another example is paid upgrades—or the lack thereof. A standard software business model has historically been to sell customers an initial license at one price and future licenses at a discounted rate. This rewards loyal customers and gives developers a recurring source of revenue and the ability to make a living from a fixed customer base instead of continually needing to reach new customers. It allows users to decide whether particular upgrades are worth paying for—some users might choose to skip a version or two. At the same time, from the developer's perspective, this model makes customers more likely to continue using the developer's software instead of shopping around for a better deal.

None of the major mobile app stores have a paid upgrade mechanism, however. This has several consequences. It might mean that a developer sells a license to its software once and continues to provide updates for free, indefinitely.[51] This may seem to be the most customer-friendly option. However, almost all software requires

---

[50] This was the issue that popular free video player VLC faced. After being pulled from the app store because of complaints from individual contributors / rightsholders, *see* Marco Tabini, *Licensing Dispute Could Drive VLC Out of the App Store*, MACWORLD (November 1, 2010), https://www.macworld.com/article/1155338/vlc_licensing.html, its returned after a substantial rewrite. *See* Felix Paul Kühne, Press Release, *VLC for iOS 2.0*, VideoLAN (July 18, 2013), https://www.videolan.org/press/ios2out.html.

[51] Certain updates—such as security patches, and compatibility updates that require minimal developer effort (for example, a recompile with some settings adjusted)—perhaps *should* be provided for free. But developers of commercial software often depend on recurring revenue from upgrades, and the alternatives (subscription models, selling new versions as new apps) can fall short in a number of ways. This issue has been often discussed in the Apple community. *See* Nat Swanner, *Apple's Refusal to Allow Paid Upgrades in the Mac App Store Hurts Developers and Users*, THE NEXT WEB (August 16, 2015), https://thenextweb.com/insider/2015/08/16/apples-refusal-to-allow-paid-upgrades-in-the-app-store-hurts-developers-and-users/, also Kirk McElhearn, *App Store upgrades: It's not that Apple can't do it, but they won't do it*, Macworld (May 12, 2017), https://www.macworld.com/article/3196016/app-store-upgrades-it-s-not-that-apple-can-t-do-it-but-they-won-t-do-it.html.

ongoing maintenance to add new features, patch security flaws, and keep up with platform changes. A business with one-time income but recurring costs is simply not sustainable. It may work if the initial payment is high enough or the velocity of new customers is great enough—but it rarely is, and many applications soon reach a saturation point. In the initial growth phase of mobile platforms, perhaps these problems were masked. But now, the lack of upgrade pricing causes many developers to abandon apps after release or to seek out an alternative business model where possible.

One alternative to upgrades is to list major app updates on the app store as a new app, rather than an update to the existing app. Under this model, an existing app may get basic security and compatibility updates for a time, but a user would have to buy the new app—or rather, the new listing for the updated version of the same app—to get major updates. But this approach has costs. Users who are accustomed to free updates, or discounted upgrades, may feel they are getting ripped off—even if the cost of the new app, up front, is simply what the cost of an upgrade would have been to begin with.[52] Upgrade pricing feels like a perk for loyal customers, and this model lacks that. More saliently to developers, the lack of a lower-cost upgrade path for existing users makes it more likely they will just buy an entirely different app rather than upgrading. While arguably good for competition, this leads to less predictability for existing developers.

It is possible to work around these shortcomings—for example, the iOS App Store has a "bundle" feature that enables consumers to buy apps together at a discount, which is pro-rated if a user has already purchased an app in the bundle. By creating a bundle containing both the old app and the new app, some developers have created what amounts to a discount on the new app for people who have already purchased the old app. While this does somewhat replicate upgrade pricing, it has downsides as well; namely complexity, and the fact that it requires that the old app continue to be listed on the store, which can lead to inadvertent purchases of the old, rather than the new, app.

The increased ease of creating and distributing software that platform maintainers have created, as well as forcing all sales into a single channel, has created a downward pressure on app pricing—it is difficult for a developer to charge an up-front price for an app at all (or at least not a significant one) which to an extent makes problems around paid upgrades beside the point—the problem is getting paid at all. As early as 2009 observers were noting that the pricing of apps was headed close zero,[53] and according to Business of Apps, "In May 2019, the average

---

[52] Selling all copies of a new app at the price of an upgrade could lead to lower developer revenue, or lead to more due to increased sales. The lack of flexibility for developers means it is hard to run natural experiments.

[53] John Herrman, *The App Store Effect: Are iPhone Apps Headed for Oblivion?*, GIZMODO (October 16, 2009), https://gizmodo.com/the-app-store-effect-are-iphone-apps-headed-for-oblivi-5378390.

cost of apps on the Apple App Store came to a $1.01. Games (in which the freemium revenue model is particularly prevalent) average $0.49. Collectively the average price for all apps is $0.88."[54] While niche software developers have found ways to charge up-front prices for apps,[55] the lack of paid upgrades as well as the general downward pressure has led to various other business models.

Arbitrary rule enforcement discourages developer innovation

Arbitrary and inconsistent application of app store rules can also prevent developers from creating certain kinds of apps, even if they do not violate any written rule. When Apple first introduced a "widget" function, for instance, it at first disallowed complex widgets like calculators,[56] before changing course.[57] Independent app developer Rogue Amoeba has observed, "Apple first approved, then removed, a major v3 update, seemingly because it competed with an Apple HARDWARE product. A huge feature, which violated no written rule, had to be pulled. Months of work were lost,"[58] and "After an incident-free v1.0 launch, v1.0.1 bug fixes were held up for 3+ months, due to Apple's total misunderstanding of how IP law works at a basic level. After much public fighting, our updates were eventually allowed."[59]

Typically, app stores provide little in the way of explanation as to why an app might have been rejected beyond a reference to a general rule, without individualized guidance or feedback, or an explanation of why similar apps might have been accepted, but not the one in question.[60] Developers have no formal due process rights, and platform maintainers appear free to apply "rules" that don't really exist, or develop new interpretations of existing rules on the fly, with little recourse other than the court of public opinion.

---

[54] Mansoor Iqbal, *App Download and Usage Statistics (2019)*, BUSINESS OF APPS (April 24, 2020), https://www.businessofapps.com/data/app-statistics/.
[55] For example, Tot for iOS, a new app, costs $20. Federico Viticci, *Tot Review: Collect and Edit Bits of Text*, MACSTORIES (February 27, 2020), https://www.macstories.net/reviews/tot-review-collect-and-edit-bits-of-text.
[56] Jordan Kahn, Apple Making iOS 8 Notification Center a Bit Less Useful by Banning Calculator Widgets (October 29, 2014), https://9to5mac.com/2014/10/29/apple-making-ios-8-notification-center-a-bit-less-useful-by-banning-calculator-widgets.
[57] *See* PCalc, About PCalc for iPad, iPhone, and Apple Watch, https://www.pcalc.com/ios.
[58] Rogue Amoeba, https://twitter.com/RogueAmoeba/status/1273635569617305601?s=20 (June 18, 2020).
[59] Rogue Amoeba, https://twitter.com/RogueAmoeba/status/1273634800751050753?s=20 (June 18, 2010).
[60] While most developers do not publicly document their app store rejection stories, there are numerous examples. This story provides an overview of some of them: Ryan Christoffel, *10 Years of App Store Controversies*, MACSTORIES (July 18, 2018), https://www.macstories.net/stories/10-years-of-app-store-controversies. *See also* TJ Addams, *My Apple App Store Rejection and Approval Story* (August 13, 2019), https://medium.com/swlh/my-apple-app-store-rejection-and-approval-story-cc692a6cb7e3;

Incentivizing ad-supported apps is problematic in a number of ways

Advertisements are one common business model that developers adopt in response to the business model constraints of the app store. An ad-supported app does not have to pay 30% of its sale price, or of an ongoing subscription or of continuing in-app purchases, to the app store. It does not have to worry about the lack of recurring revenue that the lack of a paid upgrade model creates. It does not have to worry about "race to the bottom" pricing.

That said, advertisements are hardly an ideal outcome as a means to support software development. Ads are necessarily intrusive, and on smaller devices, may take up valuable space that would otherwise be devoted to application functions or content. Downloading and displaying ads may degrade the performance or battery life of a user's device. Ads may present a risk to user privacy, and depend on or incentivize intrusive data-collection and the user of personal information in other ways.

These considerations concerning advertisements raise concerns and questions beyond the scope of this paper, but it is at least relevant to observe that plentiful ad-supported "free" apps for users is not necessarily the best market outcome given the externalities that ad-supported business models create, even if users tend to gravitate toward apps with the lowest up-front cost. (One example where this can go wrong is with "free" weather apps. Because weather apps, by their nature, are more likely to gather granular user location data, they are more likely to carry unacceptable privacy tradeoffs that users may not even be aware of.[61])

To the extent that platform maintainers value user privacy and promoting quality experiences, they should not actively promote and favor ad-supported apps over other business models. Instead, ads should be one option among many, instead of the path of least resistance for developers looking to develop a sustainable business model.

App stores encourage subscriptions, which don't always make sense

Some developers have adopted subscriptions as another business alternative. Subscriptions are, of course, the logical and predominant business model for video and music streaming services. The subscription model is ideal for software in some ways, as it creates ongoing revenue that offsets ongoing costs—both development and maintenance costs, as well as infrastructure costs for back-end services that may be associated with software.

---

[61] Jason Koebler, *Stop Using Third-Party Weather Apps*, MOTHERBOARD (January 4, 2019), https://www.vice.com/en_us/article/gy77wy/stop-using-third-party-weather-apps.

But this model has its downsides as well. First, in-app subscriptions are treated as in-app purchases, which means that the developer pays the 30% of the subscription price (less, after a time) to Apple or Google up front. Subscriptions also add complexity to how developers design and monetize their apps. As developer Brent Simmons wrote,

> The best part of the App Store, years ago, from this developer's point of view, was that it was easy to charge money for an app. No need to set up a system — just choose the price, and Apple takes care of everything. So easy!
>
> But these days, in almost all cases, you'd be ill-advised to charge up front for your app. You need a trial version and in-app purchasing (IAP) and maybe a subscription.
>
> Here's the thing: this is a *massive* pain in the ass to implement, test, and support — Apple does *not* make it easy.[62]

Additionally, customers may object to paying for subscriptions for services that are not immediately visible to them. Subscriptions are traditionally seen as a subscription *to* something—in the digital context, perhaps some cloud service or media library. Subscribing to a podcast player or a calendar app is another matter. In general, subscriptions might make the most sense for major app developers like Adobe or Microsoft, not smaller developers, or for software that is more typically sold to businesses.

Increasingly, subscriptions are more like a patronage model that provides ongoing support for a developer to continue work on her project rather than a fee that offsets a specific marginal cost. Patronage models have many benefits and can work well for some kinds of creators, as the success of services like Patreon shows. At the same time, users have little insight to whether their payments truly are going to fund further development and maintenance of the app. In any event, patronage is not how the software industry has traditionally worked, and the terminology and tools that developers have access to are not well-suited to the task.

There are further drawbacks to a subscription approach, as well. Subscriptions to various apps and services can add up and are harder to mentally keep track of than one-time purchases, leading to "subscription fatigue,"[63] or even people paying for subscriptions to things they no longer use. Subscriptions can be abused, as well: App stores have faced a problem with scam apps that prey on unwary users by

---

[62] Brent Simmons, *One Advantage of the App Store That's Gone*, INESSENTIAL (June 20, 2020), https://inessential.com/2020/06/20/one_advantage_of_the_app_store_thats_gon.
[63] *See 'Subscription Fatigue': Nearly Half of U.S. Consumers Frustrated by Streaming Explosion, Study Finds,* VARIETY (March 18, 2019), https://variety.com/2019/digital/news/streaming-subscription-fatigue-us-consumers-deloitte-study-1203166046.

offering a "free trial" for a basic app (say, a weather app, or a voice memo recorder), but that auto-renews at an absurd rate, such as $50/week. Subscription abuse of this kind is an unintended consequence of app store design decisions, and one that, due to the cut off the top that the platform takes, platforms actually profit from.

In-app purchase requirements can limit developer flexibility and revenue

Yet another business model that is enabled by app stores—and also, to an extent, forced on developers, given pricing constraints—is the use of in-app purchases to unlock new features. Under this approach, the basic app can be free or low-cost, and advanced functionality can be unlocked with further payments. This allows for a degree of beneficial price discrimination—only "pro" users need pay for pro features that only they would use—and allows for new features to be presented as new in-app purchases. However, this approach also has many limitations and drawbacks. Software applications sometimes need to be rewritten or rearchitected to make them more expansible, to increase performance or as the result of major platform technology transitions. Such a rewritten app may make sense as a paid upgrade, even if it has no new customer-facing features, but it's not feasible to offer "rewritten app" as an in-app purchase. Determining which, if any, features can even be logically separated out as an in-app purchase may be difficult or impossible, and it is likely not possible to keep selling new in-app purchases to a single customer indefinitely, even if that same customer would happily pay an upgrade every 18 months or so for an important tool.

In-app purchases have also led, unfortunately, to the rise of abusive business models—games with addictive casino-like mechanics like loot boxes, for example, take advantage of human psychology and prey on children to encourage users to continually spend to keep playing. And while in-app purchases may be a good way for players to unlock additional game content that doesn't affect the fundamental nature of the game (such as cosmetic features like new outfits or more levels of play), far too many games employ "play-to-win" mechanisms where they cannot reasonably be completed unless users purchase in-app "gems" or weapons or other content—a fact which is usually not apparent to users before they invest time in playing a game. The abuse of in-app purchases, which unscrupulous developers and platforms alike both profit from, shows that the design and policies of app stores can not only encourage some business models while discouraging others, but can also negatively impact users directly.

In-app purchases are also at the heart of one of the main ways that app stores constrain developer business models—platforms, to varying extents, require that developers use platform-provided payment systems in order to conduct sales, from which they (at least, both Apple and Google) take the same 30% cut as they do from paid apps.

At a high level, this is not necessarily objectionable: If app stores take a 30% cut from the sale of paid software, but allow in-app purchases to be provided by any means, then many apps that are not paid might simply switch to a "free" (to download) model, with most app functionality locked behind a credit card purchase. At the outset, this could annoy customers, as platform-provided in-app purchases use a customers' saved payment information and do not require the inconvenient, and potentially risky from a privacy and fraud standpoint, entry of payment information. Also, while it is fair to argue about the exact rates, it is reasonable for an app store to earn a return, or at least cover costs, instead of simply being a file host for developers. The question then is not the existence of in-app payment rules, but what those rules apply to.

Here, the differences between the leading mobile app stores, Google Play and the iOS App Store, are salient. Both require that developers use the platform payment system for app features such as game levels, additional tools and features, and so forth. Neither of them requires that the in-app purchase system be used for payment for physical goods and services. But Apple, and not Google, does require that the in-app payment system be used for media purchases and subscriptions. Apple views these as more akin to app features than physical goods. Thus, while Amazon can sell you a paper book via its app without giving Apple a 30% cut, it cannot sell you a Kindle book the same way. To buy a Kindle book, you must go to a web browser or some other device without such a restriction. Similarly, Netflix or Hulu cannot sign customers up for new subscriptions in their apps using their own payment systems—they either must pay Apple the commission, or simply allow users with existing accounts that they have created elsewhere (e.g., on the web) to sign in, and hope that new users don't find this confusing.

These rules can seem arbitrary, even confusing, to developers. According to one developer:

> If we wanted to launch, we had to disable a part of the app that allowed Tech Top 10 users to preview The Information articles and subscribe to read them. The reason: our The Information subscription—which is a separate subscription from the Tech Top 10 —doesn't go through the App Store. I knew that to sell the Tech Top 10 app through the App Store, we had to use the App Store payment method. But I didn't realize how far Apple's grip on commerce went. [64]

The enforcement of this requirement can seem quite severe, as not only are developers required to use in-app purchases if they offer any in-app payment system at all, they are also forbidden from informing customers of alternatives (e.g., subscribing on a website).

---

[64] Jessica E. Lessin, *Inside Our App Store Ordeal*, (December 9, 2019), https://www.linkedin.com/pulse/inside-our-apple-app-store-ordeal-jessica-e-lessin.

This developer limitation carries profound competitive implications, as Apple itself competes in many media and digital services markets. It sells music and music subscriptions; TV and movies as well as a video subscription service; ebooks; and even cloud storage. In all of these markets, Apple can offer in-app payments without paying 30% to some third party, and none of its competitors can. (In fact, Apple prevents apps from even directing users to a web browser to purchase a subscription. If they don't offer an in-app purchase, they simply have to be silent on how a user might sign up.) This allows Apple to give its own media apps and digital services an advantage in the marketplace and enables it to achieve success for reasons other than their merits.

Analyst Ben Thompson offered the following analysis of this issue:

> To put it another way, Apple profits handsomely from having a monopoly on iOS: if you want the Apple software experience, you have no choice but to buy Apple hardware. That is perfectly legitimate. The company, though, is leveraging that monopoly into an adjacent market — the digital content market — and rent-seeking. Apple does nothing to increase the value of Netflix shows or Spotify music or Amazon books or any number of digital services from any number of app providers; they simply skim off 30% because they can.
>
> To be clear, Apple absolutely did create the modern app marketplace, and, as the company loves to brag, an entire new economy full of new types of jobs. That, though, is precisely the problem: the App Store is not a fun side diversion; it is one of the largest platforms we have ever seen, on which hundreds of thousands of people are seeking to build real businesses, and that carries different types of responsibilities — and legal limitations — than an OS feature. It is bad for society generally and, I strongly believe, illegal for Apple to have crafted App Store rules such that it can leverage its smartphone share into monopoly profits on digital goods and services that are on iOS not because iOS is anything special, but because that is the only possible way to reach nearly 50% of the U.S. population.[65]

Compounding this issue, Apple has recently begun interpreting its rules even more strictly—instead of just requiring that, if there is an in-app purchase functionality at all, that it use Apple's system, but *requiring* that some kinds of apps offer in-app purchase functionality, when they would otherwise prefer not to. While this policy

---

[65] Ben Thompson, *Antitrust, the App Store, and Apple*, STRATECHERY (November 27, 2018), https://stratechery.com/2018/antitrust-the-app-store-and-apple. From the perspective of this paper whether or not Apple's behavior is "illegal" is irrelevant; the question is whether it would be broadly beneficial if it changed its practices.

change has been in place for some time, it recently has received more attention due to a dispute regarding Hey, a premium email service from Basecamp.[66]

Rule 3.1.3(a) of the App Store review guidelines state:

> Apps may allow a user to access previously purchased content or content subscriptions (specifically: magazines, newspapers, books, audio, music, video, access to professional databases, VoIP, cloud storage, and approved services such as classroom management apps), provided that you agree not to directly or indirectly target iOS users to use a purchasing method other than in-app purchase, and your general communications about other purchasing methods are not designed to discourage use of in-app purchase.[67]

Apple has begun to interpret that rule as *only* applying to the app categories that are specifically listed, and disallowing other categories of app from offering "reader" functionality. Thus, while Amazon is free to allow Kindle users to access Kindle books purchased on the Amazon website inside the Kindle app, Apple has told the developers of Hey that if Hey can only function with a paid subscription, that it *must* offer iOS users the ability to purchase one from within the app, paying Apple 30%.[68] (When pressed on how inconsistently it appears to apply this rule—for example, it is not applied to the Gmail app, even though GSuite, like Hey, is a paid email service, Apple claims to be applying a "business vs. consumer" distinction which is not found anywhere in its app review guidelines.[69]) It goes further than even taking a cut of transactions that happen on a platform to *require* that transactions take place that otherwise would not, or to require apps to offer new features or functionality in order to be listed on an app store. (Hey has since changed its iOS app to offer a free option, meaning that Apple no longer requires it to offer an in-app purchase for subscription plans.[70])

Putting aside the fact that the app store may be the sole way to get software onto a device, it does not seem unreasonable for Apple or Google to require that apps use their billing systems for in-app purchases of genuine app functionality, such as

---

[66] Nick Statt, *Apple Doubles Down on Controversial Decision to Reject Email App Hey*, THE VERGE (June 18, 2020), https://www.theverge.com/2020/6/18/21296180/apple-hey-email-app-basecamp-rejection-response-controversy-antitrust-regulation.
[67] Apple, *App Store Review Guidelines*, https://developer.apple.com/app-store/review/guidelines.
[68] David Heinemeier Hansson, https://twitter.com/dhh/status/1272976901762478080?s=21 (June 16, 2020).
[69] John Gruber, *The Flimsiness of 'Business vs. Consumer' as a Justification for Apple's Rejection of Hey from the App Store for Not Using In-App Purchases*, DARING FIREBALL (June 16, 2020), https://daringfireball.net/2020/06/hey_app_store_rejection_flimsiness. There may be a broad difference between software and services that are usually purchased in bulk by employers for their employees, and ones that are usually purchased by individuals for their own use, but there is likely so much overlap between those two categories that attempting to embody this in app store rules would lead to unintended outcomes.
[70] Jason Fried, Apple, Hey, and the Path Forward (June 22, 2020), https://hey.com/apple/path.

application features, game levels, or feature unlocks of various kinds. That is, if a feature could have been offered via a single up-front purchase, but the app instead is a free download with premium features locked behind an in-app purchase, requiring that those features use the app store's payment system seems like a reasonable way to avoid a form of arbitrage, where apps are "free" but unusable until a user pays using a third-party billing system. Instead, requiring that software features be available via a platform's in-app purchase system allows for apps to provide basic functionality for free, or work analogously to free trials, depending only on consumer demand, instead of a desire of developers to avoid paying a 30% cut of sales.

But this does not mean that digital content purchases in an app, or subscriptions associated with an app, should require a 30% payment to the platform maintainer. In-app purchase rules around software functions simply preserve the integrity of the app store as a retail store, but retail stores do not typically take a percentage of later purchases or subscriptions relating to a product they sell. For example, if Best Buy sells a Kindle, it doesn't require that Amazon pay it 30% of the proceeds of books purchased on that device, or if it sells a Nintendo Switch, it doesn't require that Nintendo use a Best Buy-provided payment system for subscriptions to Nintendo Switch Online.[71]

As mentioned above, Apple's practices in this area are under investigation by both the European Commission, as well as the House Subcommittee on Antitrust. This issue will be discussed further as it relates to Spotify's specific complaint, as well as in the section on policy recommendations.

Censorship and curation

App stores have effectively unlimited "shelf space," but for various reasons, they are selective about what apps they will actually carry. As discussed above, one of the most useful functions that app stores provide is preventing (or trying to prevent) harmful software from being distributed. This can include software that is harmful to a user's device, violates privacy, or is simply fraudulent. However, app store curation can go far beyond that.

One kind of curation is moral. App stores typically do not want to carry apps whose primary purpose is to access or display explicit sexual content and may restrict

[71] In part, app stores being far more restrictive than traditional retail stores may relate to customers, and sellers, have other options. To be sure, alternatives to the app store such as sideloading could be one way of ensuring that the cut taken by the store owner is not excessive. However, even in a world where all the major platforms supported sideloading, competition and consumer protection issues are still at play, since sideloading, even on platforms that support it, is inevitably more difficult for users than just relying on the default app store.

violent and even "controversial" subjects, as well. This is unsurprising; mainstream brick-and-mortar retail stores also typically exclude such content. Even so, this kind of restriction raises some issues. First, of course, much of the content that app stores want to exclude for moral or taste reasons is, in fact, lawful content that adults have a constitutional right to produce and access. This doesn't just mean pornography—app store policies will generally exclude content (at least of a sexual nature, or content touching on controversial subjects) that is commonly found in R-rated films or in literary novels. But Best Buy or Target refusing to stock something doesn't mean that no one can get it elsewhere. These retailers are merely serving as curators, and consumers have other choices. But when an app store that is the only possible way to access certain kinds of content (say, a game) refuses to carry something, it can look more like censorship than curation. That being said, few people complain that app stores have policies designed to exclude hateful or harassing content, even though such content is generally lawful and protected by the First Amendment. The question for most people is not whether there should be some sort of curation, but where to draw the line.

There are practical difficulties, as well. The same platforms that operate app stores often also operate services that sell or stream music, movies, and books. They are typically far less prudish with respect to what media of this kind they will carry than with apps. Why the difference? It could be that a publisher, a record label, or movie studio will be the entity seen as directly responsible for the content — in other words, not the retailer — but nevertheless, the inconsistency in the kinds of material that creators can produce in one medium and not another seems unsustainable.

Additionally, apps that access online content are sometimes held to be responsible for that content, but the basis for this is somewhat confusing, and it is hard to fashion a one-size-fits-all rule. For example, the maker of a dictionary app was rejected from the app store because it could access databases that provided definitions for "urban slang,"[72] which seemed questionable, but it is hard to argue with Apple taking down the Tumblr app from the app store for that service's failure to remove child pornography.[73] Apple requires Reddit clients to block adult content by default, unless users opt into that account in their Reddit account settings, on the Reddit website (apps themselves are not allowed to offer such a setting).[74] At the same time, Apple's own Safari web browser can access any material that is posted online and does not block it by default. There may be no completely straightforward way for a platform maintainer to both preserve a family-friendly

[72] Rik Myslewski, *Apple Denies Censoring App Store Swear Words*, THE REGISTER (August 6, 2009), https://www.theregister.co.uk/2009/08/06/apple_denies_dictionary_censorship.
[73] Jon Porter, *Tumblr was Removed from Apple's App Store Over Child Pornography Issues*, THE VERGE (November 20, 2018), https://www.theverge.com/2018/11/20/18104366/tumblr-ios-app-child-pornography-removed-from-app-store.
[74] Nicolas Deleon, *Apple is Cracking Down on NSFW Content Inside Reddit Apps*, MOTHERBOARD (April 12, 2016), https://www.vice.com/en_us/article/78k8yb/reddit-ios-apps-disappear-nsfw-content.

app store while allowing apps that can access internet and user-generated content, but it does not seem unreasonable to ask app stores to be transparent as to their reasoning and to treat similar apps similarly.

Restrictions on apps that touch on "controversial" topics are, in many ways, more troubling than app restrictions on salacious or sexual content. A few examples will be discussed below.

App stores may also refuse to carry entire app categories. Apple will not carry apps that it feels are used to facilitate copyright infringement, for instance, such as BitTorrent clients and retro game emulators, even though both of those app categories are capable of, in the Supreme Court's phrase, "substantial noninfringing uses."[75] Apps that download and install executable code are also prohibited by Apple. In fact, Apple's rules used to be much stricter, prohibiting even *interpreted* code—a prohibition which made many categories of apps difficult or impossible to implement, such as educational and programming apps, and which severely limited the functionality of advanced apps that use scripting to enhance and automate their functionality. Apple has since relaxed these rules, which now state that:

> [A]n Application may not download or install executable code. Interpreted code may be downloaded to an Application but only so long as such code: (a) does not change the primary purpose of the Application by providing features or functionality that are inconsistent with the intended and advertised purpose of the Application as submitted to the App Store, (b) does not create a store or storefront for other code or applications, and (c) does not bypass signing, sandbox, or other security features of the OS.[76]

However, it is worth considering exactly why Apple would ever put so much effort into what may seem to the outsider like a seemingly arbitrary retraction. It is because, having put certain restrictions on what it carries in the app store, and instituting a process for approving apps, it then had to think of ways its policies might be circumvented. Allowing apps to change their functionality via downloaded code is one such way. But if, as discussed below, the app store was not the *only* way to distribute apps to iOS devices, the incentive for developers to attempt to skirt the rules may be lessened.

Customer ownership, resale, and preservation

One drawback of app stores and related platform rules is a feature of DRM generally: It interferes with basic consumer rights, including the right to resell or transfer property.

---

[75] *Sony Corp. of America v. Universal City Studios*, 464 U.S. 417, 428 (1984).

[76] This user comment from Hacker News gives the history of this. DerekL (July 20, 2017), https://news.ycombinator.com/item?id=14809096.

In the first place, despite what app store terms and conditions (and confused judges) might say, customers purchase copies of apps—not mere "licenses." This is because customers usually own the devices where the apps are saved. Copyright law defines a "copy" as a "material object,"[77] meaning that whoever owns the material object, owns a copy of whatever copyrighted work is embodied on that material object. A person who owns an iPhone owns a copy of iOS, and a copy of the various apps installed on the phone, in the same sense that a person who owns a vinyl record owns a copy of an album, or a person who owns a video game cartridge or disc owns a copy of the game. This ownership is, of course, distinct from ownership of intellectual property rights, and many of the uses that a copy owner can put a copy to do not require permission of the underlying rightsholder. While software industry lawyers attempt to confuse this issue, no flurry of EULAs, terms of service, licenses, and contracts can rewrite the basic terms of the Copyright Act.

When customers purchase a copy of an app, they also receive a license—that is, permission to use the app and make new copies of it in certain circumstances. However, customers do not need permission to run software they own a copy of, just as a reader doesn't need a license to read a book.[78] Additionally, under the first sale doctrine, users do not need permission to transfer copies of software they lawfully own (in the case of mobile devices, the copies are typically stored in the device's memory). However, there are caveats to this. While a license is properly defined as permission, not a contract,[79] some of the terms of something styled as a "license" may function as contract conditions. While a contract cannot redefine legal realities (e.g., transfer a sale into a "license"), people can give up legal rights they would otherwise have via contract. Additionally, apps being tied to specific user accounts via DRM, the anti-circumvention provisions of the DMCA, as well as the fact that a customer's right to transfer a copy of software refers to just that—the right to transfer a copy, not to make a new copy and then delete the old one—puts practical and legal difficulties in the way of any consumers who might want to avail

---

[77] 17 U.S.C. § 101.

[78] Courts have (probably wrongly) held that executing a software program, which copies portions of the program into a computer's RAM from its permanent storage, implicates the reproduction right in copyright. However, 17 U.S.C. § 117, though initially designed to cover the *installation* of software on a device, not its use, also applies these "RAM copies" and states that any "new cop[ies]…created as an essential step in the utilization of the computer program in conjunction with a machine" are noninfringing provided the user owns a copy of the computer program. By attempting to assert that customers do not actually own material objects (copies of software) that they plainly do own, software industry legalese attempts to render 17 U.S.C. § 117 inapplicable, meaning that customers need a license to make RAM copies—to actually use software they paid for—and provides the opportunity to place conditions on that license. Strongly asserting that users do in fact own material objects such as phones thus carries the implication that they don't need a license to merely use software, or to transfer copies, but only to do things that actually would otherwise infringe on the exclusive rights of a copyright owner.

[79] Lawyers and even judges are not generally very clear about this. A contract that grants a user a license may simply be called a "license" or a "license agreement."

themselves of the resale and transfer rights (codified in the first sale doctrine) that are traditional with most forms of media and software. That said, customers do at least retain the basic right to sell or give away their devices—thanks to the first sale doctrine, the fact that those devices embed copies of software does not by itself interfere with this.

None of this changes the fact, though, that customers effectively cannot resell individual apps. Nor can they even transfer entire accounts. It's not even possible to meaningfully leave a digital account to your children or spouse after you die—while it is possible to give them your user name and password, platforms typically provide no way to merge an existing account and its various purchases with another account, meaning that perhaps thousands of dollars of purchased apps (as well as music, movies, books, and other media) simply vanish at the time of one's passing. Imagine if a person's library of books just vanished into a puff of smoke when they died, and couldn't enrich and educate future generations. This is the reality of app stores.

Similar issues stand in the way of software preservation. Software, unlike most other forms of media, is tied to particular platforms (both hardware and software) and these platforms change, become obsolete, and are even forgotten. App store terms, legal restrictions, and practical challenges stand as obstacles in the way of archivists, librarians, avid fans, and historians who might want to preserve apps as cultural and historical artifacts.

Single target for scams

Any time you make a single source the only place where a user can access apps—or anything else—you set up that source as a target for bad actors. This has happened on app stores, as persistent bad actors are able to get harmful apps past app review, or even to change the nature of their app after it has been approved.

One common scam involves apps that use deceptive user interface patterns to trick users into "agreeing" to expensive subscriptions that offer no useful functionality.[80] Even when it comes to apps that are not, strictly speaking, scams, it is common for top-grossing apps to offer functionality that is already built-in to the phone or

[80] *See* John Gruber, *Gaming the App Store*, DARING FIREBALL (December 3, 2018), https://daringfireball.net/linked/2018/12/03/barnard-game-app-store, https://www.forbes.com/sites/johnkoetsier/2018/10/04/app-scams-cheap-utility-apps-are-stealing-260-2500-or-even-4700-each-year-per-user/#5d5f91ee162a.

available for free elsewhere.[81] Platform maintainers are aware of these problems and seek to curb them,[82]even though it is true that they profit from them, as well.

Apart from the financial harm caused by scam in-app purchases or subscriptions, and even though apps are limited in the damage they can cause by sandboxing and other protections, they can still attempt to exfiltrate user data (e.g., by tricking users into enabling location tracking, or entering passwords that they then capture) in ways that are hard to solve through purely technical means. While these problems are not unique to app stores, the lack of user options means that problems on the app store are all the more serious, and a lack of competition between different sources of apps may limit the incentives of the platform owner to improve matters.

Exclusion of certain markets

For any number of reasons, a platform might not offer services, or only a subset of services, in a given country. When app stores are the sole way to install software on a device, this leaves people with nowhere else to turn for what may be basic categories of apps.[83] While the details of national content laws, trade sanctions, and other matters are beyond the scope of this paper, people in the United States and other developed countries sometimes do not appreciate the extent to which the availability of digital services—including app stores—varies globally.[84]

App store tradeoffs may not be the best for all users

App store policies can constrain what apps can and can't do. Even if you assume that the policies that app stores adopt correctly balance security, convenience, and functionality for most users, there will always be a category of user who is willing to make different tradeoffs. For example, it might make sense to prevent apps from over-aggressively tracking a user's location, as this could both be bad for privacy or the device's battery life. But there may be situations where an informed user is aware of these tradeoffs but is still willing to take them. The one-size-fits-all approach of app stores that are the sole source of software on a platform can limit a device's usefulness for some category of user, with switching costs, network effects, and other factors making it difficult or impossible for a user to just switch to

---

[81] Sarah Perez, *Sneaky Subscriptions are Plaguing the App Store*, TECHCRUNCH (October 15, 2018), https://techcrunch.com/2018/10/15/sneaky-subscriptions-are-plaguing-the-app-store/.
[82] Dami Lee, *Apple Adds Extra Step to App Store Subscriptions to Prevent Scams and Accidental Purchases*, THE VERGE (April 12, 2019), https://www.theverge.com/2019/4/12/18307785/apple-ios-subscription-confirmation-scam-accidental-purchases.
[83] Bryan Pon, WINNERS & LOSERS IN THE GLOBAL APP ECONOMY, Farnham, Surrey, United Kingdom: Caribou Digital Publishing (2016).
[84] *See* Apple Support, Availability of Apple Media Services, https://support.apple.com/en-us/HT204411#latam-car.

another platform. Here, as in many situations, the problem isn't the choices the app store makes per se, but the lack of alternatives to the app store, that can create consumer harm. Even if the app store allows for some flexibility, the power of defaults selected by a dominant app store will have outsized influence.

Case Studies

Specific instances of behavior where app store policies have harmed competition, or reduced user choices, can be more instructive than generalizations. This section will list a few illustrative examples. They are not intended to be exhaustive or present the most extreme or egregious cases of app store practices gone wrong. Some of the issues discussed may have even been resolved. These examples, instead, are intended just to provide context and to show that the proposals given here are intended to solve real-world problems.

Duplicating built-in functionality and defaults

Early in the life of its App Store, Apple didn't allow third-party apps that competed with its built-in apps at all. Apple rejected the apps for "duplicat[ing] built-in functionality" of the phone. Thus, third-party email and podcast apps[85] weren't allowed on the store at all. They couldn't compete on the app market, much less compete on an even playing field. Eventually, this restriction was lifted, but other restrictions remain.

For instance, third-party browsers cannot use their own rendering engine—the part of a web browser that actually decodes and displays HTML, CSS, and JavaScript. Instead, they are mere user interface wrappers to iOS's built-in rendering engine. Apple cites security and performance reasons for this, and there likely are concerns in these areas. Nevertheless, the restrictions prevent third-party browsers from competing with Apple's built-in Safari browser through greater compatibility with sites or higher performance, limiting them to offering different user interface choices, different syncing options, and other more superficial features.[86]

---

[85] Chris Foresman, *Apple Rejects Another App For "Duplicating Functionality,"* ARS TECHNICA (September 22, 2018), https://arstechnica.com/gadgets/2008/09/apple-rejects-another-app-for-duplicating-functionality; Robert Palmer, *App Disqualified From App Store Because It 'Duplicates Itunes Functionality,'* ENGADGET (September 12, 2008), https://www.engadget.com/2008/09/12/app-disqualified-from-app-store-because-it-duplicates-itunes-fu.

[86] Additionally, even in its browser, Apple makes some choices it justifies in the name of security and privacy, that some developers see as impeding competition and user functionality—for example, the ability of web pages to create persistent local storage can be seen as an attempt to unfairly privilege native apps over web apps. *See* Aral Balkin, *Apple Just Killed Offline Web Apps While Purporting To Protect Your Privacy: Why That's A Bad Thing And Why You Should Care*, ARAL BALKIN BLOG (March 25, 2020), https://ar.al/2020/03/25/apple-just-killed-offline-web-apps-while-purporting-to-protect-your-privacy-why-thats-a-bad-thing-and-why-you-should-care. Browser competition would allow different

Further, while Apple is reported to be considering changing its policy here, it is not possible to set third-party apps as defaults. This means at least three things: First, third-party apps cannot be set as the defaults for certain kinds of URLs. Web URLs (that begin with http:// or https://) open in Safari. Mail URLs (mailto://) open in Apple Mail or, if that app has been uninstalled, nowhere.[87] (According to Phillip Shoemaker, who oversaw the Apple's app store approvals from 2009 to 2016, this is because Apple is worried that companies like Google or Facebook would promote apps to replace core iPhone functions like calling and messaging.[88])

Second, iOS provides certain functionality of its built-in apps to third-party apps as an API: For instance, a user might be able to send an email via an Apple Mail compose sheet. It is not possible to replace this functionality systematically. Third, Siri—which itself cannot be replaced with a third-party voice assistant, unlike Google Assistant on Android—uses certain apps as its default. For example, if a user says, "Make a note that says, 'Buy some milk,'" the note will be created in Apple's Notes app. If a user says, "Play music by Bob Dylan," Siri will try to use Apple Music. It is possible to use third-party apps but only if you specify the app's name for each command ("Play music by Bob Dylan in Spotify"), which may seem like a small inconvenience but is enough to discourage users from using the feature at all.

Finally, there is the simple issue that Apple preinstalls its own apps and increasingly advertises them to users in various parts of its software—particularly those apps that generate subscription revenue. It is hard, especially for mass market apps, to compete with built-in offerings. It is probably going too far to suggest that there should not be built-in apps at all. Devices should have certain built-in capabilities over time, and user expectations about what those should be might expand over time. But it's one thing to say that a phone should come with a built-in browser, calculator, or email app, and quite another thing to say that there should be a built-in, highly preferenced music streaming app or video service. Where to draw the line (and how to draw it) may be a hard question, but this doesn't mean there shouldn't be a line.

---

browsers to make different choices, with Apple able to make the privacy choices it does without being under a cloud of suspicion of anticompetitive motives.

[87] Apple allows users to "uninstall" certain default apps, and "redownload them" from the app store, but in reality, the code for the apps in question is always on-device, just hidden. So "uninstalling" these apps does not actually free up any storage space.

[88] James Vincent, *Apple's Former App Approval Chief Says He's 'Really Worried' About Company's Anticompetitive Behavior*, THE VERGE (May 29, 2019), https://www.theverge.com/2019/5/29/18643868/apple-app-store-approval-process-antitrust-phillip-shoemaker-interview.

Spotify, and the required use of in-app purchase system for non-app content

As touched on above, one of the most significant controversies over in-app purchases in app stores is Apple's requirement that they be used not just for application functionality (such as extra game levels, additional processing tools in graphics apps, and so on), but for media purchases and digital subscriptions[89] as well. To be sure, these things fall somewhere between real-life goods and services and app features. Calling an Uber or buying batteries on Amazon seem like clear examples of real-life services, and with these, Apple and Google both allow developers to provide their own payment processing.[90] However, Apple categorizes things like music streaming subscriptions as more like app features, while Google categorizes them as more like services like ordering an Uber.[91] But if developers can sell users a music CD without paying the platform 30%, they should be able to sell music downloads the same way—Google's policy seems more developer-friendly and straightforward.

Spotify, in particular, has drawn attention to the ways that Apple's app store policies preference its service, Apple Music, at the expense of consumer choice. Spotify's Daniel Ek has laid out the changes he believes Apple should make:

- *First, apps should be able to compete fairly on the merits, and not based on who owns the App Store. We should all be subject to the same fair set of rules and restrictions—including Apple Music.*
- *Second, consumers should have a real choice of payment systems, and not be "locked in" or forced to use systems with discriminatory tariffs such as Apple's.*

---

[89] Apple requires that apps and services like Dropbox, Microsoft Office 365, and Adobe Creative Cloud use its in-app purchase system for subscriptions. It may not be straightforward to determine whether a service is closer to unlocking actual app functionality or not, but a company in Apple's position should likely err on the side of permissiveness and treat digital services such as cloud storage (or, for example, domain name registration) as not requiring use of its IAP system, since they offer functionality which is useful outside of the app itself. A harder question is whether games that require or allow some form of continuing payment that could not have been offered via a single, up-front purchase should be required to use the in-app purchases system.

[90] Apple does now allow Apple Pay to be used for these kinds of purchases, though it is not required. Apple Pay is just a normal payment service like PayPal or Venmo that can be used in apps and in the Safari browser, or in real-life stores, and is distinct from the in-app purchase system, which uses the billing infrastructure originally designed for iTunes.

[91] Google, Developer Policy Center, Monetization and Ads, https://play.google.com/about/monetization-ads.

- *Finally, app stores should not be allowed to control the communications between services and users, including placing unfair restrictions on marketing and promotions that benefit consumers.*[92]

The first request would be mostly realized by implementing the other two, which are more concrete. Specifically, he argues that Spotify should be able to provide its own payment processing for in-app subscriptions, rather than having to use Apple's own. This practice would bring Apple in line with Google's policies, which allow media apps like Spotify's to bypass the in-app purchase system. He also believes that Apple should not be able to prevent apps from telling users where they can go to sign up for subscriptions.

Spotify's problem is magnified by the business it is in. Music subscription services pay out most of their revenue to rightsholders. They simply cannot afford to lose 30% of their revenue to Apple, and if they simply up their subscription rates to make up the difference, they risk becoming uncompetitive with Apple, whose own music service does not have to pay 30% to a third party.

This conflict of interest applies in other areas, as well. Customers cannot buy ebooks in the Kindle app on iOS, for instance, but they can in Apple's own Books app. Nor can Amazon direct users to the web, where they can purchase books which then appear in the app. The popularity of the Kindle store relative to Apple's ebook store shows that many consumers are aware of how to get around Apple's restrictions, but certainly many are not. And Apple does not even compete in some of the categories where its policies make it difficult to sell media purchases or subscriptions, such as digital comics—even though it is increasingly entering new categories, such as video streaming subscriptions. But even viewed from a consumer protection rather than a competition perspective, Apple's policies, at best, inconvenience users for reasons that are less than compelling. It doesn't even seem likely that they even generate much revenue for Apple, at least not directly, since many apps appear to simply lack in-app purchase functionality altogether rather than paying Apple 30%.[93]

Apple adds to this issue by not even allowing Spotify, Amazon, or any other app developer to provide a link to a web page for sign-up, or even explanatory text describing how to do so. Apps that require some sort of subscription to work, but that don't use Apple's in-app purchase system, are required to be black boxes: They don't work, and developers are not allowed to communicate to customers how to get

---

[92] Daniel Ek, *Consumers and Innovators Win on a Level Playing Field*, SPOTIFY NEWSROOM (March 13, 2019), https://newsroom.spotify.com/2019-03-13/consumers-and-innovators-win-on-a-level-playing-field.

[93] Netflix used to have in-app purchases through Apple, but it stopped offering them to new customers, though existing subscriptions continue to work. YouTube TV recently decided to cancel all in-app iOS subscriptions, requiring users to re-subscribe through alternate means.

them to work. If the app even links to a web page for unrelated reasons (for example, a privacy policy, or a password recovery form), and it is possible to tap on a logo and be taken to the developer's home page, where there is a sign-up form, Apple, will reject the app. Customers might be able to figure out how to go to Spotify's website, or Netflix.com to sign up for a subscription, but it is easy to see how this policy disadvantages smaller apps without the same level of brand recognition.

Fortnite and the Google Play Store

Fortnite by Epic Games shows that it is difficult, even for popular apps, to successfully bypass an app store, even when the platform makes it possible. Specifically, on Android, to install Fortnite users used to be required to sideload—download the app from a web page an install it—rather than installing it from the Google Play Store.[94] Epic Games did this because Fortnite's entire business model is charging users for optional in-app items, such as costumes for their characters,[95] and it did not want to give Google 30% of each transaction. But the difficulty that ordinary users had in sideloading software meant that Epic Games eventually had to start offering the game in the Play Store regardless.[96] These events show that while sideloading remains an important way for certain kinds of apps to bypass app stores, and can be extremely valuable in some cases, by itself it does not provide a reason to avoid other scrutiny of app store policies.

Exclusive access to hardware features

A similar issue to a platform maintainer giving its own apps preferential treatment is when it gives its own hardware accessories access to platform features unavailable to other companies. For example, Apple's AirPods and Beats Headphones can not only set up and pair with Apple devices much more easily than third-party headphones can, but they can be immediately used with any of a user's other Apple devices without cumbersome unpairing/repairing steps. As Nilay Patel of The Verge has written, "There is literally no way for another headphone company to compete with the advantages Apple gives itself[.]"[97]

---

[94] John Callaham, *Here's How to Install Fortnite for Android*, ANDROID AUTHORITY (April 17, 2020), https://www.androidauthority.com/how-to-install-fortnite-for-android-894001.
[95] Akhilesh Ganti, *How Does Fortnite Make Money?*, INVESTOPEDIA (March 27, 2020), https://www.investopedia.com/tech/how-does-fortnite-make-money.
[96] Nick Statt, *Epic gives in to Google and Releases Fortnite on the Play Store*, THE VERGE (April 21, 2020), https://www.theverge.com/2020/4/21/21229943/epic-games-fortnite-google-play-store-available-third-party-software.
[97] Nilay Patel, https://twitter.com/reckless/status/1188964589553143809?lang=en (October 28, 2019).

Tile, a company that makes hardware tracking tags, has similarly sounded the alarm about Apple's rumored competing product, which is likely to have access to iPhone features that Tile cannot match.[98]

Preferential treatment of important apps

At times it can appear that preferred or important developers get treatment from app stores not available to average developers.

For example, when Uber was caught not only violating Apple's privacy policies, but attempting to use location access to hide this behavior from Apple's app reviewers, instead of being kicked out of the App Store, Apple CEO Tim Cook called Uber CEO Travis Kalanick and told him to change it.[99] It is hard to believe that an app less important to Apple's platform would have gotten away with Uber's behavior with a scolding.

Some developers have noticed that the enforcement of rules against major companies and app developers is less exacting than against smaller companies. For example, The Information noted that "the rule that we couldn't require users to enter their email" applied to them, but not to Disney and the New York Times.[100]

A more recent example is Apple exempting Amazon for its in-app purchase rules for some video content because it is a "premium subscription video entertainment provider"—a category of provider that seems only previously to have included cable services.[101] This appears motivated by Apple's desire for video services to support Apple platform features like its video-aggregating TV app and AirPlay—but it remains to be seen whether smaller video service providers will win the same exemption from App Store rules as the major providers thus far covered.

Businesses cut special deals with preferred partners all the time, and certain apps (such as Uber, Netflix, Facebook, or YouTube) may be so important to a platform that they get more leeway than others, or at least more time to comply with rules and more flexibility in their interpretation. It is even rumored that some of them pay less than the 30% or 15% that Apple typically requires, though this has not

---

[98] Adi Robertson, *Sonos and Tile Execs Warn Congress that Amazon, Google, and Apple are Killing Competition*, THE VERGE (January 21, 2020), https://www.theverge.com/2020/1/21/21070812/sonos-tile-basecamp-popsocket-congressional-hearing-amazon-google-apple-competition.
[99] Andrew Liptak, *Uber Tried to Fool Apple and Got Caught*, The Verge (April 23, 2017), https://www.theverge.com/2017/4/23/15399438/apple-uber-app-store-fingerprint-program-tim-cook-travis-kalanick.
[100] Jessica E. Lessin, *Inside our Apple App Store Ordeal*, (December 9, 2019), https://www.linkedin.com/pulse/inside-our-apple-app-store-ordeal-jessica-e-lessin/.
[101] John Gruber, *Amazon and Apple Strike Deal for Prime Video In-App Purchases and Subscriptions*, DARING FIREBALL (April 2, 2020), https://daringfireball.net/2020/04/amazon_apple_prime_video.

been publicly substantiated. If developers that did not get such treatment were able to reach customers without having to go through the App Store, e.g. through sideloading, this would be less of a competitive and fairness concern.

HKmap.live, and government pressure to censor

Last year, Apple removed an app called HKmap.live from the App Store in Hong Kong. This app allowed people to track the location of police during the civil unrest in Hong Kong. Apple attempted to argue that the app violated its policies, and violated local law, but neither claim could withstand the barest of scrutiny.[102] Yaqiu Wang from Human Rights Watch stated that this app removal "is just the latest incident of Apple caving to the Chinese government's political pressure."[103]

An even more recent incident involved Apple removing Pocket Casts and Castro, third-party podcast clients, from its App Store in China.[104] Podcast apps, like web browsers, simply access content that is available on the open web. They do, however, typically provide a directory of available podcasts. The makers of podcast apps, like browser developers, are likely to resist efforts to require that they restrict what content their users can access,[105] either by blacklisting URLs of certain podcasts or removing listings from their directory. But while independent app developers may be willing to simply forgo the Chinese market rather than comply with local laws they object to, large multinational companies like Apple are not likely to make that decision.[106] Indeed, instead of removing its own podcast app from the App Store in China, Apple simply removes "objectionable" podcasts from its directory in that country.

---

[102] Maciej Ceglowski's Twitter thread explains why. Maciej Ceglowski, https://twitter.com/Pinboard/status/1182348757360234497 (October 10, 2019).
[103] David Crawshaw and Reed Albergotti, *Apple Pulls Police-Tracking App Used by Hong Kong Protesters*, WASHINGTON POST (October 10, 2019), https://www.washingtonpost.com/world/asia_pacific/apple-pulls-police-tracking-app-used-by-hong-kong-protesters/2019/10/10/4aad5ebe-eb14-11e9-a329-7378fbfa1b63_story.html.
[104] Sam Byford, *Apple Pulls Podcast Apps in China After Government Pressure*, THE VERGE (June 11, 2010), https://www.theverge.com/2020/6/11/21287436/pocket-casts-castro-china-apple-government-pressure. China is able to block the functionality of another major independent podcast app, Overcast, without having the app removed from the store. Overcast, https://twitter.com/OvercastFM/status/1271122252868784140 (June 11, 2020) ("Overcast's servers have been blocked in China for years, so it already didn't work.") Apple also has removed other categories of app from its store in China, such as Virtual Private Network (VPN) apps that allow users to bypass government internet restrictions.
[105] While most podcast clients, including Apple's, do not host podcasts themselves, some "podcast" clients, such as Spotify, do not actually offer access to podcasts on the open internet, but host audio files themselves in a proprietary system. Apps that choose to forgo the open internet in this way may be required to make decisions about what content to actually host, and where to make it available.
[106] Google did leave the Chinese market in 2010. That said, its business in China at the time was marginal, and it has since explored ways to reenter the Chinese market in ways that are consistent with the requirements of the Chinese authorities, via its now-terminated "Project Dragonfly." Wikipedia, "Project Dragonfly", https://en.wikipedia.org/wiki/Dragonfly_(search_engine).

Again, there are solutions to this. It is not necessary to look to companies to disobey what they view as unjust local laws, to ignore political pressure from local governments or, except in extreme instances, to simply decline to do business in certain countries at all. (Apple's position in China of course is more precarious since it not only sells billions of dollars of devices into the Chinese market, but depends on Chinese manufacturing and supply chains for its global operations.) Rather, a better approach is for companies to avoid putting themselves in a situation where they even can prevent customers from installing an app if they choose. One approach would be for Apple to outsource the operation of its services in countries where it might be subject to local pressure or laws that may be objectionable from its standpoint. But this is merely passing the buck. A better approach, to be discussed below, will be the often-mentioned sideloading, which would allow users to install apps without going through a centralized app store.

Apple, AT&T, and the FCC

When the iPhone was first released, it was available only on AT&T, in the United States. Voice-calling apps were not, at first, permitted to use an iPhones mobile connection—just WiFi. While the FCC generally has jurisdiction over phone carriers like AT&T,[107] it does not have direct authority over information services like the iOS app store. However, it came out that Apple only had this restriction in place under the terms of a secret agreement with AT&T,[108] and AT&T released Apple from this agreement after the FCC signaled its intention to subject wireless networks to some form of net neutrality rules.[109] This early app store controversy shows how the gatekeeper control that app stores enable can be used in ways that are contrary to the public interest, and even due to outside pressure on the platform maintainer itself. Clear rules of the road can not only protect users from platform maintainers but also from pressure put on platform maintainers from their business partners.

---

[107] Leaving aside how the FCC under Chairman Ajit Pai decided to abdicate authority over broadband. *See* Lindsay Stern, *Two Years Later, Broadband Providers Are Still Taking Advantage of An Internet Without Net Neutrality Protections*, PUBLIC KNOWLEDGE (December 20, 2019), https://www.publicknowledge.org/blog/two-years-later-broadband-providers-are-still-taking-advantage-of-an-internet-without-net-neutrality-protections.

[108] Free Press, Press Release, *AT&T-Apple Deny and Confirm Blocking VoIP*, (August 21, 2009), https://www.freepress.net/news/press-releases/att-apple-deny-and-confirm-blocking-voip.

[109] Ryan Singel, *AT&T Relents, Opens iPhone to Skype, VOIP*, WIRED (October 6, 2019), https://www.wired.com/2009/10/iphone-att-skype/. Any reasonable legal regime would not distinguish between AT&T blocking a category of app at the network level, and it directing a third party to block a category of app from being installed on a phone to begin with.

Moral censorship

Apple is frequently subject to charges of censorship, including moral censorship, and it does apply strict standards to the App Store that are not applicable to its other content stores, where it sells movies, books, and music that touch on or feature the same subject matter that will get an app removed.[110] In an earlier version of its app store guidelines, Apple was forthright about this:

> We view Apps different than books or songs, which we do not curate. If you want to criticize a religion, write a book. If you want to describe sex, write a book or a song, or create a medical app.[111]

Some examples of this kind of content-based censorship (sometimes reversed, with the app put back after the developer makes some changes) include the removal of an app that featured the political cartoons of a Pulitzer Prize-winning artist,[112] a game about dealing with the government in a dystopian society,[113] an app that provides news about U.S. drone strikes,[114] and a strategy game about the Afghan war.[115] More recently, Apple removed all vaping apps from its store,[116] including apps that perform functions not possible on iOS without native iOS apps. As John Gruber wrote:

> The stuff about selling cartridges and sharing news — it's fine for that stuff to be out of the App Store because you can get it on the web. But Bluetooth stuff where apps were used as the interface for controlling hardware — web apps can't do that (nor should they be able to). There is no alternative to a native app, and native apps are only available on the App Store. This would be an

---

[110] Jess Bolluyt, *Why Does Apple Censor its App Store but not iTunes?*, CHEATSHEET (April 26, 2016), https://www.cheatsheet.com/gear-style/why-does-apple-censor-its-app-store-but-not-itunes.html/.
[111] Leander Kahney, *Here's the Full Text of Apple's New App Store Guidelines*, CULT OF MAC (September 9, 2010), https://www.cultofmac.com/58590/heres-the-full-text-of-apples-new-app-store-guidelines/.
[112] Ian Paul, *Apple Rejects Pulitzer Prize Winner's App*, PC WORLD (April 16, 2010), https://www.pcworld.com/article/194387/apple_rejects_pulitzer_prize_winners_app.html.
[113] Kyle Orland, *Apple Reverse Decisions, Allows Unedited Papers, Please on iPad,* ARS TECHNICA (December 12, 2014), https://arstechnica.com/gaming/2014/12/apple-forces-nude-immigrants-to-cover-up-in-ipad-version-of-papers-please/.
[114] Louise Matsakis, *Apple's Long History of Rejecting 'Objectionable Content' From the App Store*, MOTHERBOARD (July 17, 2017), https://www.vice.com/en_us/article/a3dwq8/apples-long-history-of-rejecting-objectionable-content-from-the-app-store.
[115] Charlie Hall, *Afghanistan '11, Featuring US and Taliban forces, Removed from App Store*, POLYGON (December 6, 2018), https://www.polygon.com/2018/12/6/18128924/afghanistan-11-taliban-app-store-removed.
[116] Ina Fried, Mike Allen, *Exclusive: Apple to Remove Vaping Apps from Store*, AXIOS (November 15, 2019), https://www.axios.com/exclusive-apple-to-remove-vaping-apps-from-store-8669fd94-e92a-4ce4-a9e2-ce5afa598b67.html.

> easy call to make (and would have been made from the get-go by Apple) if vaping were illegal. But it's not illegal.[117]

There may be good reasons for consumers to criticize where Apple draws the line when it comes to content decisions, and the apparent double standard between apps and other media. At the same time, its decision to not host apps that feature genuine hate speech or pornography raises fewer objections. But, as with politically-motivated removals such as that of HKmap.live, sideloading is an escape valve.

Location data in iOS 13

Another example of where the privacy and security needs of users can come into tension with free and open competition relates to how apps can access location information on current versions of iOS. Previously any app could ask for permission to access this information, and users would be presented with a dialog box where users could allow the app to access location just once, while using the app, not to allow it—or to always allow it.

However, it has become apparent that allowing apps to track users in the background can have serious privacy implications,[118] and that many apps that ask for such access do not truly need it. Thus, Apple has changed iOS to make it more difficult for apps to track users. Developers have complained—in open letters and to Congress—that this, by itself, is anti-competitive. Some of the implementation details—specifically that Apple does not subject itself to the same rules it applies to outside developers—do have competitive effects that must be considered. But the privacy of users is also a legitimate concern, and the interests of developers as a whole should not take precedence over this, either.

Specifically, Apple has changed iOS so that apps cannot directly ask for always-on tracking. Users can activate it on a per-app basis in system settings, but apps themselves present a dialog box asking for location access for just one use, or while the app is in the foreground. The system also periodically prompts users to confirm their choice.

A number of app developers complained about this change, arguing that the new feature would confuse users, who would not be able to figure out how to enable location tracking, or understand why apps that require this functionality weren't working properly without it. Tile has argued that Apple requires users to re-confirm their choice too often, and that Apple has promised to bring back some form of an

---

[117] John Gruber, *Apple is Removing All Vaping Apps from Its Store*, DARING FIREBALL (November 18, 2019), https://daringfireball.net/linked/2019/11/18/app-store-vaping.
[118] Jennifer Valentino-Devries, Natasha Singer, Michael H. Keller and Aaron Krolik, *Your Apps Know Where You Were Last Night, and They're Not Keeping It Secret*, NEW YORK TIMES (December 18, 2018), https://www.nytimes.com/interactive/2018/12/10/business/location-data-privacy-apps.html.

"Always Allow" prompt.[119] One coalition of developers added, "The changes also have the added effect of removing critical geolocation functionality while simultaneously not applying to Apple's own apps, some of which compete with the products we develop."[120] This is because when built-in iPhone functionality requires always-on tracking, Apple bypasses the app-based system of permissions entirely, and gives users instead buttons in a system settings area, which are by default set to allow tracking. For instance, Apple Maps does not even have an "always" option for location services—the most that users can give it is "While Using the App." Nevertheless, Apple tracks users at all times and gathers location information in the background on iPhones to improve Maps other services whether or not users actually use Apple Maps, or an alternative like Waze. Thus, Apple gains telemetry about traffic conditions—which improves Apple Maps—even from users who use a competing app, or who are not using a mapping app at all. The setting that controls this is found in the Settings app, under Privacy, then Location Services, then System Services (which is at the bottom of a very long list, which is otherwise alphabetical) and then, at the bottom of another long page of options, there is an "Improve Maps" button which is by default set to "on."

Certainly, some services require location access—such as "Emergency Calls and SOS," "Cell Network Search," and "Setting Time Zone" that do not raise any competitive implications, and having these services have access to location information by default is beneficial. Even so, this thin wedge should probably not be used as a justification for Apple to privilege any or all of its apps and services over those of competitors, or to rule out ways to limit the competitive harm that legitimate measures to improve user privacy and enhance device performance may cause.

Parental control apps

It is possible to install something called "profiles" on iOS devices that add features or functions not available to normal apps or that apply certain settings. One kind of profile for "Mobile Device Management" was designed for companies that issue their employees phones or otherwise need to have more centralized control over devices than iPhones and iPads allow off-the-shelf. As Apple describes it, "MDM includes updating software and device settings, monitoring compliance with organizational policies, and remotely wiping or locking devices."[121]

---

[119] Diane Bartz, *Tile Says Apple's Behavior Is Anticompetitive And Has 'Gotten Worse, Not Better*,' REUTERS (April 1, 2020), https://www.reuters.com/article/us-tech-antitrust-apple-tile/tile-says-apples-behavior-is-anticompetitive-and-has-gotten-worse-not-better-idUSKBN21J72V?.
[120] Juli Clover, *App Developers Claim Apple's iOS 13 Location Tracking Changes Are Anti-Competitive*, MACRUMORS (August 16, 2019), https://www.macrumors.com/2019/08/16/app-developers-tracking-restrictions-ios-13.
[121] Apple, Support: Mobile Device Management Settings, https://support.apple.com/guide/mdm/mdm-overview-mdmbf9e668/web.

Third-party developers, however, realized that by using MDM profiles, they could offer parents a similar level of control over their children's devices. (As a technical matter, the app developer would become the manager of the child's phone and would then provide tools for the parent to access and control it.) For example, using such software, a parent might restrict whether and to whom a child can send messages.

Apple, however, claimed that it felt that this was a misuse of MDM. It stated:

> Over the last year, we became aware that several of these parental control apps were using a highly invasive technology called Mobile Device Management, or MDM. MDM gives a third party control and access over a device and its most sensitive information including user location, app use, email accounts, camera permissions, and browsing history. We started exploring this use of MDM by non-enterprise developers back in early 2017 and updated our guidelines based on that work in mid-2017.[122]

However Apple's timing raised eyebrows, as after seemingly tacitly tolerating apps that make these kinds of uses of MDM profiles on the App Store for some time, it began cracking down on them only after it rolled out its "Screen Time" feature, which offers similar functionality to many parental control apps.[123] Developers of such apps banded together to ask Apple to reverse course,[124] which Apple eventually did.[125]

These profile features raise several very nuanced issues. The fact is that MDM profiles can be a privacy and security risk if misused. MDM profiles are sometimes used by malicious actors to spy on people, and even by "stalkerware," which is software that is like parental control apps but that is used, for instance, by abusive spouses to track their partner's whereabouts and read their communications.[126]

---

[122] Andrew Liptak, *Apple Explains Why It's Cracking Down on Third-Party Screen Time and Parental Control Apps*, THE VERGE (April 28, 2019), https://www.theverge.com/2019/4/27/18519888/apple-screen-time-app-tracking-parental-controls-report.
[123] Jack Nicas, *Apple Cracks Down on Apps That Fight iPhone Addiction*, NEW YORK TIMES (April 27, 2019), https://www.nytimes.com/2019/04/27/technology/apple-screen-time-trackers.html.
[124] Dieter Bohn, *Parental Control App Developers Band Together to Demand an API from Apple*, THE VERGE (May 30, 2019), https://www.theverge.com/2019/5/30/18646601/parental-control-app-developers-band-together-to-demand-an-api-from-apple.
[125] Sean Hollister, *After Outcry, Apple Carves Out Room for Rival Parental Control Apps to Exist*, THE VERGE (June 4, 2019), https://www.theverge.com/2019/6/4/18653142/apple-mdm-parental-control-screen-time-app-store-guidelines-update.
[126] Charles Osborne, *Apple Reveals Why App Store Parental Control App Crackdown Took Place*, ZDNET (April 29, 2019), https://www.zdnet.com/article/apple-refutes-anti-competitive-parent-control-app-claims-says-removal-was-for-security-user-privacy.

Additionally, Apple's competitive motives are not as straightforward as it might appear, because Apple does not actually *sell* Screen Time—it's included for free with iOS.[127] Rather, it seems like Apple chose to allow apps that may have technically violated the terms of the App Store until it was able to offer alternatives—a bait and switch that puts developer's livelihoods at risk, even if it did not have a straightforward financial motivation in terms of generating new sales of its own apps.

Apple's approach was essentially to continue to allow third-party MDM use for settings such as parental control apps, but to pledge to monitor them carefully. This solution though expedient at the time is not ideal, and there may be better solutions, as discussed below.

Solutions

This paper does not argue against the existence of app stores or controlled software platforms in service of some vision of openness, freedom and creativity. The benefits that app stores and controlled platforms create in terms of user trust, security, and privacy mean that the risks from gatekeeper controls they give to platform maintainers are worth tolerating. However, in the case of dominant platforms, certain policy changes can reduce those risks and benefit users and developers, without significantly undermining either the security and integrity of the platform, or the app store, nor a platform maintainer's incentive to invest in and maintain the store. Many of the solutions discussed here have already been hinted at above or follow straightforwardly from the descriptions of the problems they are intended to address.[128]

Sideloading

It is possible to preserve the benefits of a controlled app store without making it the only way to install software on devices. Alternate ways of installing software on devices would allow users to access software that is not available on an app store

---

[127] You could say that Apple was attempting to entice people with older phones to upgrade to iPhones that support the version of iOS that includes Screen Time, rather than continuing to use older phones that may work with third-party parental control apps. This motivation cannot be discounted but is likely not what drove Apple's decision.

[128] As this paper was being finalized, Astropad, a company that sells software that allows iPads to be used as an external display on Macs, a feature which Apple itself now offers, published a blog post with recommendations that in some ways mirror the ones in this section. It calls for users to be able to set default app preferences, to open up alternate in-app payment options, to allow sideloading, to give third-party developers better equal access to APIs, and to "stop sherlocking third-party developers." Savanah Reising, *Dear Apple: Here's How to Stop the Antitrust Investigations* (June 16, 2020), ASTRO BLOG, https://astropad.com/dear-apple. However, as mentioned earlier, while being Sherlocked is not a pleasant experience for developers, this paper calls for guardrails to ensure fair competition, not an outright ban on platforms integrating new features into their core system.

due to content reasons, government censorship (e.g., podcast and VPN apps in China), or simply because the developer's business model is not a good fit with the app store.

Before the advent of app stores, "sideloading" is how all software was installed on computing devices—through installation disks, web downloads, and the like. On many platforms like Windows and macOS, it remains the predominant method of installing native software. However, for devices where the primary way of installing software is through app stores, these "old-fashioned" ways of installing software go by the name of "sideloading," coined by analogy to "uploading" and "downloading."[129] Sideloaded software may be installed any number of ways—via physical media, an internet download, "package managers" of the sort that are popular on Linux, like RPM and APT, or from another device (e.g., a phone connected to a PC via USB). The question isn't whether sideloading is possible or the best way to do it—it's how to preserve the security and curation benefits of app stores, and to preserve the incentives of platform owners to invest in their app stores, while still giving users and developers more freedom than a fully locked-down platform allows. A few real-world examples of how sideloading works on existing platforms will provide context for the recommendations to follow:

- *Windows*. Windows is a typical "open" platform where the distribution and installation of software was completely decentralized. In recent years, app store models have been introduced to varying degrees of success, however. Interestingly, third-party game stores, such as Steam, have been much more successful than Microsoft's own app store (which is simply called "Microsoft Store"). There are many reasons for this, including the kinds of apps that Microsoft allowed in the store in the first place Though the driving reason for the slow uptake of the Microsoft Store is probably (1) user habits, and (2) developers being unwilling to give Microsoft a cut of the sales of paid apps unless there was a truly compelling reason to do so, which the Microsoft Store does not provide.
- *macOS*. On the Mac, whose operating system has been variously named System Software, then Mac OS, then Mac OS X, and now macOS, has a similar historical situation as Windows. Users traditionally installed software from discs and internet downloads, and user and developer habits are hard to change. Apple, after its experience with iOS Apple also rolled out its own Mac App Store, and introduced new security features that apply for apps installed from outside the store, as well. Its "Gatekeeper" feature encourages apps to be code-signed, and certain features that at first were restricted to Mac App Store apps (such as iCloud connectivity) are now available to all signed apps. At the same time, the Mac still allows users to

[129] Wikipedia, Sideloading, https://en.wikipedia.org/wiki/Sideloading#Historical. "Sideloading" is a form of "retronym," like "acoustic guitar" or "dumbphone"—a new term needed to specify what used to be the default.

run unsigned apps—though they may have to click a checkbox in System Preferences, or right/two-finger click on an unsigned app and select "Open" the first time it is launched.[130]

- *Android*. Android has allowed sideloaded apps from the beginning, and it has received some usage, as the Fortnite situation discussed above demonstrates. However, sideloaded apps have no code-signing requirements, which is a potential security problem.

This paper has had to provide a lot of context and detail to get to specific, fairly straightforward suggestions, which are discussed below.

All general-purpose consumer computing devices should allow sideloading.

Many of the problems that app stores create become problems worthy of a policy response only because of their gatekeeper status—there is no way to install software onto a device except through the app store. While perhaps it may make sense to limit any actual legal or regulatory requirements in this regard to dominant platforms, giving users and developers the flexibility to make the most of their devices is sensible across the board.

This is not to say that *all* "computers" should have this capacity. While a few enthusiasts may desire exactly that, there is no pressing general user or developer need to allow people to install "apps" on their smart thermostats or alarm clocks. While limited-purpose devices such as these, or even gaming-specific devices, may improve by being more open, they do not rise to the level of public concern as dominant, general-purpose computing platforms.[131]

Allowing sideloading is not the only change that dominant platforms should make, nor—as we've seen with Android—does sideloading necessarily reduce the importance and potential abuse of a platform's default app store. Additionally, code-signing, automated malware checks, and even placing more stringent sandboxing restrictions on sideloading will still likely not provide the same security guarantees as app store reviews and will do not replicate the other curatorial functions that app

---

[130] Several years ago, there was an interesting exchange between James Grimmelman and Jonathan Zittrain. Zittrain has been a critic of locked-down computing models and characterized the Mac App Store and its sandboxing requirements as the beginning of the end for the openness of the Mac platform. However, this model—where a single device has both a "safe" way to install software and a more open space—was called for a few years before by Zittrain himself. James Grimmelmann, *Zittrain vs. Zittrain*, THE LABORATORIUM (November 9, 2011), http://laboratorium.net/archive/2011/11/09/zittrain_vs_zittrain.

[131] Though there may even be good reason for game console platforms to allow certain sideloaded apps at a minimum, such as student or hobbyist programming projects. Additionally, consumer devices should not use software locks or contractual restrictions to limit repairs or resale, so even in that context, some level of openness is desirable.

stores perform, such as ensuring that software is useful, accessible, or even non-deceptive.

With these caveats, one can at least think of sideloading as an escape valve—available on an opt-in basis to users, for apps and situations where distribution through the app store or more formal channels is impossible or impractical. Finally, it is worth noting that a dominant platform itself can benefit from sideloading. Not only is a platform more valuable when users and developers are free to use the apps of their choice, the slight lessening of gatekeeper control that sideloading affords takes away some of the impetus to either use antitrust or common carrier-type regulation of app stores as a means of preserving user choice and developer access.

Mainstream devices should require code-signing of sideloaded apps

Many of the security benefits of app stores come from code-signing, the security technique that allows users to be sure that an app is from whom it purports to be from and has not been modified. (A more exacting form of code-signing is app notarization, where apps are first scanned by the platform provider before they can be signed.) As discussed above, Macs and Windows PCs traditionally do not require code-signing, though Macs encourage it. Android does not require it for sideloaded apps. However, requiring[132] code-signing is not just a good idea for platforms in general, it precisely addresses the specific potential shortcomings of allowing software to bypass an app store.

There should be multiple code-signing authorities

Code-signing does, however, reinforce the power of the platform in the sense that certificates still must be issued by some central authority—typically, the platform. To sign an app, a developer must be a registered Apple developer, and pay an annual fee. However, it doesn't have be this way. For example, on the web, multiple certificate authorities can provide the necessary authentication for encrypted websites.[133] A platform vendor does not have to be the only central authority allowing users to install apps and developers to sign them. Apple, for instance, could work with different organizations in the United States and around the world and design its devices to allow apps signed by them, as well. For example, a nonprofit organization like Mozilla might be able to issue certificates to some developers allowing them to sign apps,[134] or a government might allow its own apps to be signed under its own authority. Each one of these authorities could become a

---

[132] At least by default. Devices could have a "developer" mode that allows self-signed or unsigned apps, provided it is not something that unscrupulous malware developers could somehow trick users into enabling.

[133] Wikipedia, Certificate Authority, https://en.wikipedia.org/wiki/Certificate_authority.

[134] The suggestion is not that Mozilla or some other organization allows apps to be signed with its credentials, but rather that it has the same power to issue new credentials as Apple.

chokepoint, and ultimately the platform vendor retains the power to decide which different code-signing authorities to recognize, and under what terms. However, diffusing power in this way can curtail abuse and, while not perfect, may be preferable to a system where a single entity decides what apps can be installed on user devices worldwide.

In-app purchase requirements should be limited to app functionality

The competitive issues with certain in-app payment rules have been discussed at length above. Provided sideloading is available as an option to allow developers with good reason to bypass the platform's default app store entirely, it is not objectionable for platform maintainers to require that developers use an in-app purchase system for legitimate app functionality—that is, application features that could have been offered as a single up-front purchase, but are made available via in-app purchase instead. In-app purchases of this kind can be a means of facilitating free trials or tiered functionality, and do not raise the same competitive concerns, and requiring the use of an in-app payment system in this narrow case may prevent app stores from simply hosting "free" apps that require that users enter a credit card to begin using.

However, extending the requirement beyond that to other kinds of purchases, like those of ebooks, media subscriptions, in-game items, or cloud services crosses the line from ensuring the sustainability of the app store to taking a cut of someone else's business merely because it is possible. (And from a customer perspective, it makes no sense that ebooks and physical books should be treated differently.)

In-app purchases policies also can result in "reader" apps that offer no in-app purchase functionality at all, which benefits no one. How does Apple benefit from Amazon being unable to sell Kindle books directly in the Kindle app? More recently, Apple has started to *require* that some apps offer in-app functionality when the app developer has made the business decision not to—not even permitting them to function as "reader" apps. This is analogous to a broadband provider deciding to block access to an app store unless it gets a cut of the store's total revenue. In short, it is rent-seeking behavior, not a means of ensuring the viability of the app store or protecting consumers.

With dominant platforms, there can be broad anticompetitive and anti-consumer effects from policies like this. Platform maintainers should therefore limit their in-app purchase rules to application features and functions that could otherwise have been offered via an up-front purchase, not to digital media, in-game items,

subscriptions, or cloud services more broadly, and should never *require* that in-app purchases be used.[135]

Developers should be able to truthfully communicate with their customers

App developers should be able, in their app, to provide truthful and accurate information to their users. The main reason that developers currently find themselves prevented from communicating with their users—telling them how to make purchases outside the app—may perhaps be obviated if the other recommendations in this paper are followed. Nevertheless, there are times when a developer may want to inform customers about features available on other platforms, for example, or deals available if cloud subscriptions or other products and services are purchased elsewhere. As a general matter, developers should be free to inform their customers of these options.

It is a good thing, of course, when platform maintainers seek to protect their users' privacy. For example, by preventing apps from requiring that users turn over unnecessary personal information before using an app, or by requiring that apps offer privacy-protecting login options where feasible.[136] At the same time, developers should not be restricted from establishing a direct customer relationship with their users through the app. And, when developers do so, platform maintainers should not be in a position to tell developers exactly what they can say through those relationships.

Allow users to set and change defaults

Consumers expect devices to come with a certain number of pre-installed apps. A device without a web browser is broken. A device without a camera app, or an email client, similarly so. However, there is no reason for users to be locked into system default apps. Just as on desktop operating systems, users should be able to change their default browser, email client, camera app, and so on. Changing defaults does not just entail changing the apps that respond to certain URL schemes, such as mailto:// for email and http:// for the web. It also includes all the ways that an operating system may default to certain apps—notes and music in a voice assistant, and other touchpoints for an app throughout the system, such as embedded music players, web views, mail sheets, and so on. Changing defaults does not eliminate the advantage of being a default or the privilege that a platform's first-party apps get, but it does offer some breathing room for developers while enabling a greater degree of user customization.

---

[135] This applies to cloud subscriptions such as Office 365 or Dropbox storage, to digital media subscriptions like Netflix and Spotify, and to digital purchases such as music from Bandcamp, or in-game items. It should also apply to apps that use a subscription model merely as a form of optional patronage, or to access app features, or to remove ads, on an ongoing basis, as recurring payments of this kind could not have been initially offered as a single, up-front purchase.

[136] E.g., Sign in with Apple, https://support.apple.com/en-us/HT210318.

Limit preinstalled apps to essentials

While devices need to have some set of preinstalled apps, this does not mean that a dominant platform maintainer should be free to integrate just any of its apps and services into a device. As discussed above, default and preinstalled apps have a significant advantage over competing alternatives, even if an interested user can find and install third-party alternatives from the app store. There is no reasonable way to avoid this when it comes to apps that offer basic system functionality, such as taking photos or sending text messages. Additionally, as discussed above, platforms should not be prevented from improving their core product or expanding its functionality in reasonable ways, which is why this paper does not call for a complete ban on "sherlocking."

That said, it should not be fair game for a dominant platform maintainer to bundle any and every app it chooses or to merely assert that a first-party app is part of the "system." At a minimum, services that require an additional subscription, such as cloud storage or music or video streaming services, should not be given any specific preference over alternatives, including being preinstalled on a device.

It may be a judgment call as to exactly which apps a consumer expects to be a basic function of a device, and which should not. It may even change over time. But merely allowing consumers the ability to change defaults and install alternatives should not give a platform vendor free reign to engage in other forms of self-preferencing.

App store search transparency

Preinstalling apps on devices is not the only way that platform maintainers can give their own apps an advantage. The search feature of app stores—searching for app names, or for categories of app—is one of the primary ways that users discover apps. While the curatorial functions of an app store are inherently subjective, users have a reasonable expectation that search results are based on objective criteria. But without knowing what those criteria are, it is difficult to know whether poor app store results are the result of poor design, unscrupulous developers finding ways to game the system, or deliberate preferencing of apps from particular developers, including the platform itself. App stores should therefore make the ranking criteria for the app store search engine public, as well as provide a means for developers to register problems with search results (such as searches for the

exact name of an app not yielding results[137]) or when search results of particular terms are overwhelmed by apps manipulating keywords.[138]

Platforms should avoid using competitors' proprietary data to compete with them

As discussed above, platforms often introduce apps or features that compete with, or replace, offerings from third parties. In some cases, this can be anticompetitive. As a general rule, however, platforms should refrain from using the data that they have privileged access to for competitive purposes. App stores necessarily know how many times an app is downloaded and by whom. Operating systems often report usage metrics, which can be important for security and performance reasons. (For example, if an operating system bug is causing certain apps to crash more frequently, it is good if the platform maintainer is made aware of this so that it can fix the problem.)

However, to the extent that platform maintainers have access to proprietary, non-public information about third-party apps, they should generally avoid using this data in making their own choices about what products to develop, or what features to implement. Restrictions on the use of data of this kind (along with restrictions on bundling) can help limit unfair competition without altogether preventing platforms from competing in general. It can also allow platforms to continue improving core features without making it too risky for outside developers to deliver their own system improvements.[139]

Proactively offer secure APIs to for third-party developers for major new features

Platform owners have found that it is possible to give third-party developers greater power and access to more device features and user data in secure ways when it is in their interest to do so. To give an example: Android has allowed third-party replacements for the system software keyboard since version 1.5, in 2009.[140] This permits device manufacturers or users to have a different software keyboard appear whenever text entry is needed. Different keyboards can offer custom layouts and

---

[137] Complaints of this kind are numerous among developers. *See* discussion at the Apple Developer forums, "App Store search of exact app and company name returns no results," https://forums.developer.apple.com/thread/94204.
[138] *See* David Bernard, https://twitter.com/drbarnard/status/1171382087670280192 (September 10, 2019), for one example of this.
[139] Public Knowledge has broader recommendations on these points in Harold Feld, THE CASE FOR THE DIGITAL PLATFORM ACT, (May 2019), https://www.publicknowledge.org/assets/uploads/documents/Case_for_the_Digital_Platform_Act_Harold_Feld_2019.pdf.
[140] Wikipedia, Android Version History, https://en.wikipedia.org/wiki/Android_version_history.

autocorrect systems, and ultimately were the first to introduce features like swipe typing. By contrast, Apple did not introduce third-party keyboard support until iOS version 8.

There are several issues to consider regarding this feature, which shed light on ways that platforms can offer developers enhanced functionality in other contexts. The details will always be different, though common themes emerge.

First, keyboards are not regular apps. A user does not switch from Twitter to Google Maps to a keyboard. Instead, keyboards are a system-level feature and, once installed, are invoked in any number of apps. Platform maintainers need to develop the functionality to allow apps that work this way to be installed and managed. Keyboards also raise security and privacy concerns—and here, Apple has introduced privacy features that Android has not. For instance, any time that iOS recognizes that a user is entering a password, it defaults to the system keyboard. This removes the possibility entirely of a third-party keyboard recording a user's password and perhaps sending it over the internet for some unscrupulous use. Additionally, third-party keyboards by default do not have what Apple calls "full access," which means they cannot connect to the network at all. Users can override this default, and may want to in the case of keyboards that rely on machine learning in the cloud to provide users with smart autocorrect or suggestions, or simply to update their dictionaries with new words without the need for a full app update. But in the case of a keyboard that offers only, for example, a better interface for selecting emoji, there may be no reason at all for a user to allow the app to communicate with some remote servers. In short, although it took Apple much longer to offer a feature than Google, when it did offer it, it baked in protections that either Google did not think were necessary or did not think of at all.

Similar considerations could allow a platform to continue offering third-party developers the ability to reach consumers with their products, and even compete with the platform maintainer itself, while still protecting user security. For instance, as discussed earlier, Apple continually gathers location information to improve its services by default. This policy makes its decision to make it more difficult for users to grant similar permissions to third-party apps potentially anticompetitive in effect, if not intent. Apple claims that the data it collects "to improve navigation, such as routes and search terms, is not associated with your identity. Instead, that information is based on random identifiers that are constantly changing,"[141] and has even claimed that "your route from A to B is fragmented into scrambled sections on Apple servers because nobody else should know your entire route. Not even us. In fact, we don't even know who requests a

[141] Apple, Privacy, https://www.apple.com/uk/privacy.

route."[142] If this is true, then Apple should share this private data that only it can easily collect with other apps on a user's phone. This data might not entirely replace the kinds of location tracking that developers might want to engage in, but, by offering, at a minimum, the data that it collects would offset competitive harm without threatening user privacy.

Similarly, Apple should allow third-party headphone manufacturers to access the same ability to sync their pairing status across devices,[143] so that users can use the same headphones with multiple devices without tedious pairing/unpairing/repairing steps, and allow third-parties to access the iPhone's new ultra-wideband chip and features to ensure that things like Apple's rumored item-tracking tiles do not have an insuperable advantage over market leaders like Tile.[144]

This approach—engineering APIs and solutions to ensure that third parties can offer functionality either not permitted by or that simply does not fit in the standard model of an app used, and then quit, requires work from the platform maintainer. However, dominant platforms have special obligations to ensure they do not unreasonably interfere with competition.

Obligation to allow archiving / emulation of older system versions

Fundamentally, problems of software preservation and research should be solved through changes to copyright law that recognizes copying, modifying, and distributing copies of software, as well as circumventing copy and access control techniques, as lawful when done for lawful purposes, such as research, preservation, and in some cases public access. However, changes such as these are out of the scope of this paper. Platform maintainers can help the effort, though, by providing technical data and source code for operating system versions, apps, and even device information[145] for products that are no longer commercially viable, but remain of interest to professionals, archivists, and even enthusiasts. With older projects, the same security and privacy measures that can, as an unintended

---

[142] Christian Zebrig, *How Apple Maps Protects Your Privacy When Navigating*, iDownloadBlog (March 13, 2019), https://www.idownloadblog.com/2019/03/13/apple-maps-navigation-privacy.
[143] As Nilay Patel of The Verge has noted, "There is literally no way for another headphone company to compete with the advantages Apple gives itself." Nilay Patel, https://twitter.com/reckless/status/1188970777917448192 (October 28, 2019).
[144] Chance Miller, *Everything We Know So Far About Apple's Tile-Like 'AirTag' Item Trackers*, 9to5Mac (February 22, 2020), https://9to5mac.com/2020/02/22/apple-airtags-features-release.
[145] In the long term, the only way to continue to access older software is through software emulation or through specialized devices like field-programmable gate array (FGPA), e.g. the MIST 1.4 MIDI FGPA computer, AMIGAstore.eu, https://amigastore.eu/en/358-mist-midi-fpga-computer-with-midi-add-on.html, not continuing to maintain older devices. This requires that developers be able to accurately and completely implement device features in software, an effort greatly assisted by first-party technical information.

consequence, restrict worthy efforts like these have simply outlived their usefulness and should be possible to disable.

Allow users to transfer and merge accounts

The restrictions on software resale and transfer that app stores have brought about are a consumer rights problem that may require some fundamental changes to copyright law, as most media is no longer distributed via unique physical copies to which traditional notions of ownership (which is distinct from the ownership of intellectual property itself) apply to.

However, app stores can make it easier for consumers to exercise rights similar to those they traditionally enjoyed with physical property by 1) expressly allowing users to transfer the ownership of user accounts (with all their associated purchases, data, and subscriptions), and 2) allowing users to merge accounts, so that the digital assets of one account can be easily added to another.

The first issue may be less of a practical problem today than it may seem, because while the terms of service of typically forbid account transfer, all it takes in practical terms is handing over a username and password. While doing this may be helpful at times—for example, a person can leave their user name and password in a will to allow their children to access and download their photos and documents after they pass away—there is generally no clean way to add purchased content from one account to another. Since it is typically not possible, or at best extremely cumbersome, to use more than one account with a device at a time, this makes it all but impossible for a person to transfer, bequeath, or sell digital purchases. A parent can leave a library of books, records, and even software (that is on physical media) to their children-—there is no reason why these basic rights of alienability and ownership should cease to exist in practical terms merely because of technological changes.

Thus, while full "digital first sale" consumer rights may be a matter for legislation, dominant app stores can at least more easily facilitate digital transfers through formally allowing account transfers and create account merger capabilities. This does not mean that they have to allow piecemeal digital first sale of individual apps or media purchases or even acknowledge that consumers do in fact "own" what they have purchased. It simply means that, whatever rights consumers do have with respect to a particular user account, should be transferrable in a practical, authorized way.[146] This will help users regain some of the traditional ownership rights they enjoyed, by default, with physical media.

---

[146] Jason Mazzone has argued for a similar policy. COPYFRAUD AND OTHER ABUSES OF INTELLECTUAL PROPERTY LAW 135 (2011).

Due process for developers

Public Knowledge has elsewhere argued that dominant platforms, in general, should offer their users due process.[147] This well-developed legal principle can ensure that rules are being applied fairly, and that mistakes—which are inevitable—can be corrected. In the case of a dominant app store, the right to due process would rest with developers who need to access the store to reach customers or an audience. Due process can ensure that small developers are treated as fairly as large ones, that outcomes are predictable, that precedent is followed, and that different people handling disputes within an organization do not come to arbitrarily different conclusions.

Among other things, due process includes notice of a proposed action and grounds asserted for it, the right to be heard by an unbiased decision-maker, an opportunity to present reasons for the proposed action not to be taken, the right to present evidence, the right to know the opposing evidence, the right to a decision based only on the evidence presented, and the right to written findings of fact including the basis for a decision. While platform maintainers typically have minimal appeal processes (and Apple has recently introduced a mechanism for developers to challenge rules themselves, and not just their application), they generally fall short of the full array of procedural safeguards that full due process rights provide.[148]

Due process can appear different in different situations, and platforms must build in safeguards against abuse, and ways to dispense with trivial disputes. Establishing procedures of this kind can help developers dealing with platforms feel less like they're talking to a black box and can help create both the reality and sense of fairer outcomes.

Greater business model flexibility

The structure of app stores (and retailers in general) always places constraints on how sellers can do business.[149] In many ways, app stores offer more flexibility than traditional retailers do. For example, Apple currently allows app bundles, and both Apple and Google permit sharing of purchases within families, as well as offering in-app purchases and upgrades, as well as subscription plans.

---

[147] John Bergmayer, EVEN UNDER KIND MASTERS (2018), https://www.publicknowledge.org/assets/uploads/blog/Even_Under_Kind_Masters.pdf.
[148] Apple, Press Release, Apple Reveals New Developer Technologies To Foster The Next Generation of Apps (June 22, 2020), https://www.apple.com/newsroom/2020/06/apple-reveals-new-developer-technologies-to-foster-the-next-generation-of-apps ("[D]evelopers will not only be able to appeal decisions about whether an app violates a given guideline of the App Store Review Guidelines, but will also have a mechanism to challenge the guideline itself.").
[149] One example is how Apple's app store requires that apps be listed at certain prices—$29.99, for example, not $30. Jessica E. Lessin, *Inside Our App Store Ordeal*, (December 9, 2019), https://www.linkedin.com/pulse/inside-our-apple-app-store-ordeal-jessica-e-lessin.

A simple pricing model for app stores may have made sense early on. However, these examples show that app stores have already moved beyond the approach of just having apps be free, or available with a single up-front purchase. Innovation is fine—subscriptions and in-app purchases have enabled business models and categories of apps that did not exist before and have made some apps more accessible to users. (For example, an Office 365 or Adobe Creative Cloud subscription, that can be cancelled at any time, can be more accessible to ordinary users than an upfront purchase of hundreds of dollars.) But some tried-and-true software pricing models are tried-and-true for a reason. Upgrade pricing benefits developers by giving them a recurring source of revenue (and keeping customers loyal, if upgrade pricing is cheaper than switching to a competing app), and users, by giving them a discount compared with buying a license up front, and by presenting them with the *option*, not the requirement, of paying to upgrade. (Many users happily continue using old versions of software for as long as they work or skip a few cycles of upgrades. This is not possible with subscriptions, or with software that automatically upgrades without user choice.). Free trials allow users to see if they really want software before paying for it, without necessitating convoluted in-app purchases schemes. This gives developers the opportunity to prove to customers why apps with a higher price might nevertheless be worthwhile. Mature app stores should add them as options.

Conclusion

App stores do not signal the end of the open, competitive world of computing that has brought users so much, and created so many benefits for developers, since personal computing first took off in the late 1970s and eventually supercharged by the advent of smartphones. Instead, app stores and related platform technologies help address many of the problems that have accompanied the rise of these fantastically capable devices, guarding consumers against viruses and malware, and protecting their security and privacy. But, the immense power these technologies give platform maintainers can be used to enhance and fortify the platform maintainers' market power and its bottom line, in addition to protecting users. This paper's recommendations are designed to ensure that app stores continue to serve their pro-user purpose, and can continue to be sustainable businesses for platform maintainers, by creating more flexibility and safeguards for users and independent developers without giving up important security protections.