APPLE'S RESPONSE TO
CMA WORKING PAPERS 1 TO 5

1 August 2024

Apple responds below to the CMA's emerging thinking on mobile browsing as set out in working papers 1 to 5. Apple considers it beneficial to address these in a single submission because they all relate to mobile browsing, raise common themes, and rely on similar evidence. Sections I and II outline overarching themes that Apple believes are important to understanding and properly analyzing its approach to mobile browsing. The remaining sections address issues unique to each working paper.

I. INTRODUCTION

1. Browsing on iOS and across platforms is well-functioning, and Apple faces robust competition from a wide variety of differentiated browsers to attract and retain users. Unlike many of its mobile browsing competitors, Apple's business decisions are principally driven by its incentives as a device manufacturer. When the iPhone was launched in 2007, Steve Jobs described it as three separate devices in one: an iPod with touch controls, a revolutionary mobile phone, and a breakthrough internet communications device.[1] The iPhone ushered in the era of desktop-class browsing on a mobile device and kicked-off a revolution in mobile computing.[2]

2. To deliver desktop-class browsing within the stringent battery and power demands and touch-native interface of a smartphone, while protecting the privacy and security of its most private and personal device, Apple integrated WebKit deeply into iOS.

3. In 2008, Apple created the App Store and licensed a software development kit ("SDK") to allow developers to make native apps for the iPhone, including third-party browsers. This was no easy decision given the significant security, performance, and other risks posed by third-party native apps.[3] Apple's assessment of those risks informed its requirement that native apps browsing the web use WebKit.

4. The WebKit requirement has been in place since the inception of the App Store in the UK and has contributed significantly to iPhone user satisfaction as well as device and browser competition. Since the App Store's launch, Apple has continued to offer a safe, private and high-quality mobile browsing experience, regardless of which browser a user chooses. WebKit's tight integration with iOS provides all developers with industry-leading performance and protections out-of-the-box. This allows browser developers to focus their efforts on competing more aggressively – including against Safari – without introducing risks to users or the platform. The evidence and data that Apple has submitted to date confirm these realities, and demonstrate that mobile browsing on iOS is feature-rich, highly innovative, and continues to meet users' high expectations and developer needs.

A. Apple's unique approach to mobile browsing on iOS

5. Two critical insights must underpin a proper evaluation of how Apple competes in mobile browsing on iOS: (1) Apple's incentives are driven by competition at the device level and the objective to sell more devices; and (2) WebKit is vital to achieving a high-quality browsing experience on iOS and overall platform privacy, security and performance.

---

[1] See https://www.apple.com/newsroom/2007/01/09Apple-Reinvents-the-Phone-with-iPhone/.

[2] https://www.businessinsider.com/watch-steve-jobs-first-iphone-10-years-ago-legendary-keynote-macworld-sale-2017-6?r=US&IR=T#jobs-took-to-the-stage-in-his-trademark-black-turtleneck-sweater-for-the-now-legendary-presentation-in-january-1.

[3] As Steve Jobs explained in a 2008 interview prior to the SDK's launch, "*I think sometime later this year we will find a way to let third parties write apps and still preserve security. But until we can find that way, we can't compromise the security of the phone. Nobody's perfect, but we sure don't want our phone to crash.*" Peter Cohen and Jason Snell, *Steve Jobs at D: All Things Digital, Live Coverage*, Macworld (May 29, 2007), https://www.macworld.com/article/185837/steveatd.html; *See also*, John Markoff, *Steve Jobs Walks the Tightrope Again*, New York Times (Jan. 12, 2007), https://www.nytimes.com/2007/01/12/technology/12apple.html.

1.   *Apple's incentives are driven by competition at the device level*

6. Apple is focused on providing the best user experience on its devices to make them more attractive and enhance device sales. Apple's approach has thus always been to enhance the value of its device through tight integration of hardware and software, as well as a diverse offering of first-party and third-party services. Apple knows that poor user experiences would be fatal to the core of its business – as nearly 80% of Apple's revenue is generated from device sales.[4] Apple thus has a very strong incentive to protect its users and developers by providing robust security, privacy, and performance capabilities for its device platform. That is why Apple has invested billions of dollars in innovations to improve the processing speed, functionality, and quality of its mobile devices.[5] To encourage a rich variety of apps and services on its platform, Apple also developed a rich platform and resources – including over 250,000 APIs, more than 170,000 technical documents and sample code, and extensive support, consulting, and App Review resources – to allow millions of third-party developers to build and market 1.8 million apps.

7. As a device maker, Apple is committed to maintaining the integrity of the platform and it devotes significant resources to combat risks to the user experience. For instance, from 2020 through 2023, Apple prevented a combined total of over $7 billion in potentially fraudulent transactions on the App Store, including more than $1.8 billion in 2023 alone. Also in 2023, Apple rejected more than 1.7 million app submissions for failing to meet the App Store's basic requirements to protect user privacy, security, and content.[6]

8. Other market participants do not always share Apple's incentive to focus on the integrity of the platform as a whole. Developers, for example, do not bear the costs of harms occurring outside their apps. Frequently, harms like poor performance and over-consumption of device battery life or memory are attributed to the device, even when they may, in fact, be caused by an app. In any scenario, Apple directly bears the risk that poor performance of complementary apps or services will impact user satisfaction and consequently diminish demand for Apple devices.

9. Apple differentiates its devices from its rivals through a relentless focus on privacy, security, and performance, which are key dimensions of device-level competition. Apple's approach also differs markedly from its rivals because it does not rely on monetization via user data collection and advertising. Device-focused business models like Apple's have been shown to yield enhanced investments in key determinants of a user's experience, particularly app-level competition, app curation, and privacy and security protections than prevails on ad-funded rivals like Android.[7]

---

[4] See Apple's 2023 Annual Report, available at https://investor.apple.com/investor-relations/default.aspx.

[5] Apple's continuing focus on improving the iPhone has resulted in a product that competes vigorously with Samsung, Google, Huawei, OnePlus, Xaomi, Nothing, Motorola and other smartphone manufacturers, both in relation to price and device features. There are multiple websites, articles and blogposts (such as https://www.techradar.com/uk/news/best-phone) that summarize the many ways in which various smartphones compare and the pros and cons of each.

[6] Over 248,000 app submissions were rejected from the App Store because they violated Apple's policies against spam, blatantly copying other apps, or otherwise misleading users. This is in addition to over 38,000 app submissions that were rejected for containing hidden or undocumented features. Finally, over 375,000 app submissions were rejected for privacy violations. Details of Apple's efforts are available at: https://www.apple.com/newsroom/2024/05/app-store-stopped-over-7-billion-usd-in-potentially-fraudulent-transactions/.

[7] Recent research in the field of industrial organization has established this proposition. The economics literature has accepted that a business model based on monetization mostly through sales of devices induces a platform to: (i) better internalize the interests of users in terms of privacy and security, invest more in app curation, and set the right standards for apps when they risk negative platform externalities; and (ii) promote competition between apps to increase demand of devices. See for

10. Apple's starkly different approach to user privacy and security is also reflected in real-world outcomes. Several recent developments highlight this contrast:

- A recent litigation revealed that Google harvested search data from millions of unknowing Chrome users who had turned on 'incognito' mode — a browsing feature specifically designed to protect against any kind of tracking at all — and used it to measure web traffic and sell ads.[8]

- For at least ten years, Google has been tracking users of its first-party services via a default extension embedded in Chromium.[9]

- Research articles have criticized Google's Topics API for having significant failings in its purported privacy protections.[10]

- Microsoft's Windows operating system suffered a worldwide outage affecting millions of machines as a result of faulty third-party software that was granted very low-level operating system privileges, while Apple devices remained unaffected.[11]

- Microsoft was forced to delay roll out of a new AI feature after heavy criticism from third parties that it would be a security "disaster."[12]

- Meta received significant scrutiny after discovery that it had installed a tracking tool on hospital websites to collect sensitive health information from patients.[13]

> *2. Browsing is a significant threat vector on iOS and a key determinant of device performance*

11. The most serious attack vectors of mobile devices include: (1) app distribution; and (2) web browsers, which have experienced a surge of attacks in recent years.[14] Apple addresses the first vector through App Review, code signing requirements for apps, and designing secure architectures that limit the impact of malicious apps. However, these mitigations only protect users of Apple's App Store. They do nothing to protect users against the threat of unreviewed and potentially malicious content provided directly from the web.

12. It is uncontroversial that the web is a constant and ever-expanding source of security and privacy threats. The very act of web browsing carries significant risks to the OS because it

---

instance Etro, Federico, 2021, Device-funded vs Ad-funded Platforms, International Journal of Industrial Organization, 75 (March), 102711, see https://www.sciencedirect.com/science/article/pii/S0167718721000047, and 2023, Platform Competition with Free Entry of Sellers, International Journal of Industrial Organization, 89 (July), 102903, see https://www.sciencedirect.com/science/article/pii/S0167718722000789; Bisceglia, Michele and Jean Tirole, 2023, Fair Gatekeeping in Digital Ecosystems, mimeo, Toulouse School of Economics, see https://www.tse-fr.eu/sites/default/files/TSE/documents/doc/wp/2023/wp_tse_1452.pdf; and Joshua Gans, 2024, Three Things about Mobile App Commissions, University of Toronto, see https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4780115.

[8] https://www.npr.org/2024/04/01/1242019127/google-incognito-mode-settlement-search-history.

[9] https://simonwillison.net/2024/Jul/9/hangout_servicesthunkjs/; https://x.com/lcasdev/status/1810696257137959018; https://daringfireball.net/linked/2024/07/12/google-chrome-system-monitoring.

[10] https://ieeexplore.ieee.org/document/10579537; https://webkit.org/blog/15697/private-browsing-2-0/.

[11] https://www.wsj.com/tech/cybersecurity/microsoft-tech-outage-role-crowdstrike-50917b90.

[12] https://www.theverge.com/2024/6/13/24178144/microsoft-windows-ai-recall-feature-delay.

[13] https://themarkup.org/pixel-hunt/2022/06/16/facebook-is-receiving-sensitive-medical-information-from-hospital-websites.

[14] See, for example: https://www.zdnet.com/article/theres-been-a-big-rise-in-hackers-targeting-google-chrome-doing-this-one-thing-can-help-protect-you/.

necessarily exposes a user's mobile device to web content: both good (e.g. what a user intended to navigate to or has an interest in seeing), and bad (e.g. external threats).

13. These risks are particularly heightened given how mobile devices are used. Mobile devices are constantly connected to the internet, carried with users everywhere, and contain an array of highly sensitive and valuable information, including banking/financial information, health information, location and contacts, and other personal information. And as the CMA itself has recognised, mobile devices play a "*fundamental role in the lives of UK citizens*" thanks to the "*dramatic evolution in the role and uses of mobile phones over the last two decades*".[15] As a result, mobile devices are particularly vulnerable to theft, loss, or direct attack, and incredibly profitable to attackers if compromised.

14. That is why WebKit and Safari have always been core elements of Apple's mobile platform, and its multi-layered approach to privacy and security — a key part of which has always been WebKit's tight integration into iOS as well as the requirement that all browsers on iOS use WebKit as the rendering engine. The browsing experience on iOS strongly influences both device security and privacy, as well as overall device performance[16] and the tightly integrated nature of WebKit allows Apple to achieve important benefits on iOS devices:

- The WebKit requirement allows Apple to prioritize security at a platform level in a comprehensive, effective, and efficient way, such as by providing a mechanism for the universal adoption of new browser engine mitigations without requiring developers to have a high degree of security expertise.[17]

- Users benefit from the WebKit requirement because they can trust that all apps on Apple's platform provide the same high, baseline level of security and privacy protections across all web browsing experiences, reflecting Apple's "privacy by design" approach.

- Developers benefit because the high standard of security and privacy protections provided by WebKit gives them the freedom to invest in developing other innovative features and content. Because of WebKit, iOS developers do not need to actively monitor threats and implement mitigations on their own, or constantly update the versions of the browser engine they use.

15. Security and privacy protections are key areas of differentiation for Apple on its mobile device platform, and Apple has invested heavily to provide continuous improvements to Safari and WebKit. In its February 2024 supplemental response to the Statement of Issues, Apple highlighted examples of the literally hundreds of recent feature developments, including features like push notifications, offscreen canvas, and screen orientation, as well as APIs for storage, user activation, and screen wake lock. Recent releases have garnered very positive feedback from developers, as exemplified by Open Web Advocacy's tweet: "*iOS 16.4 is a massive step forwards towards feature parity between Web Apps and Native Apps. Big thank-*

---

[15] See Mobile Ecosystems Market Study (MEMS) Final Report, paragraphs 2.1 and 2.2.

[16] For example, as regards performance, it is well-established that some browsers such as Chrome on Mac (which is based on the Chromium browser engine) consume significant amounts of memory, leading to materially worse battery life when they are used.

[17] See, for example https://www.diffen.com/difference/Android_vs_iOS, which notes in relation to security that: "*Android software patches are available soonest to Pixel device users. Manufacturers tend to lag behind in pushing out these updates. So at any given time a vast majority of Android devices are running outdated OS software*" whereas on iOS "*Security threats [are] rare, because iOS is locked and downloading apps out of the App Store is complicated*".

you to the Safari/Webkit team for all their hard work".[18] Numerous commentators have also recognised Safari's leading role in feature development.[19]

16. Apple's role as a platform provider means that it has an incentive to provide users with the widest set of mobile browsing options possible, while maintaining high standards of safety and privacy on the device. And while Safari competes with third-party browsers, Apple has no incentive to dilute third-party browsing experiences, as doing so would ultimately harm Apple's device sales. Apple's business model requires the availability of a rich variety of services on its platform in order to attract users. This approach does not hinder competition, it enhances it.

B.   The CMA's own evidence reflects well-functioning markets for browsers and browser engines on iOS

17. Contrary to the conclusions drawn in the working papers, the evidence before the CMA shows that competition among mobile browsers on iOS, as well as across iOS and Android, is already robust.

1.   Users have many diverse mobile browser offerings to choose from

18. With respect to iOS devices, there are currently over 100 browser apps available on the UK App Store, each eligible to be selected as a user's default browser. These browsers actively differentiate themselves on a wide array of features, as set out in detail in Section IV below. Examples include: (1) Arc browser, which differentiates on the basis of its AI-based offering; (2) Ecosia, which offers a "green" choice for environmentally conscious users; (3) Opera, which leans heavily on privacy-enhancing features; (4) Chrome, which differentiates on the basis of speed and customizability; and (5) Firefox, which emphasizes its customizable user interface features and privacy. The significant extent of differentiation amongst browsers is illustrated in further detail below.[20]

19. As AI becomes a more important dimension of competition for browsers on iOS, Apple expects that differentiation amongst browsers will continue to increase in the future.[21]

20. Notably, browsers that operate across iOS and Android tend to market similar (and in many cases, identical) features on iOS and Android, as can be seen from Figure 1 below.[22]

[18] https://twitter.com/openwebadvocacy/status/1640492875245920256?s=61&t=Ee4BlnZrrdOUTbth TvXUfg

[19] See for example: https://shoptalkshow.com/598/#t=56:31, a podcast about building websites from Dave Rupert and Chris Coyier in January 2024 reacting to Apple's recent releases and noting "*It's cool to see Safari updating regularly and even paving the way, leading on new features*"; similarly, in a 2 February 2024 episode of the Syntax podcast (https://syntax.fm/show/725/safari-is-the-new-chrome-jen-simmons-of-apple), the hosts noted that "*Safari has been absolutely killing it lately .. Safari has been first to a lot of this stuff ... I feel like there's been crazy momentum lately … what's going on at Apple that Safari is doing so well.*"; and in a January 2024 episode of Kevin Powell's channel (https://www.youtube.com/watch?v=nn3vYS_msc0), the discussion centred on "*Safari and how it's been leading the way on a lot of the new big features like Has and container queries amongst a bunch of other ones*".

[20] See, in particular, Section IV(B)(2) and Annex_Apple WP Response_01.

[21] See, for example, https://www.computerworld.com/article/2520807/how-ai-will-change-the-way-you-use-your-browser.html; https://fastcompanyme.com/technology/ai-is-taking-over-your-web-browser-heres-what-youll-love-and-hate/ discussing the impact of AI on web browsing across various browser offerings.

[22] See further Section IV(B)(3) and Annex_Apple WP response_03.

Figure 1: Browsers market similar features across platforms
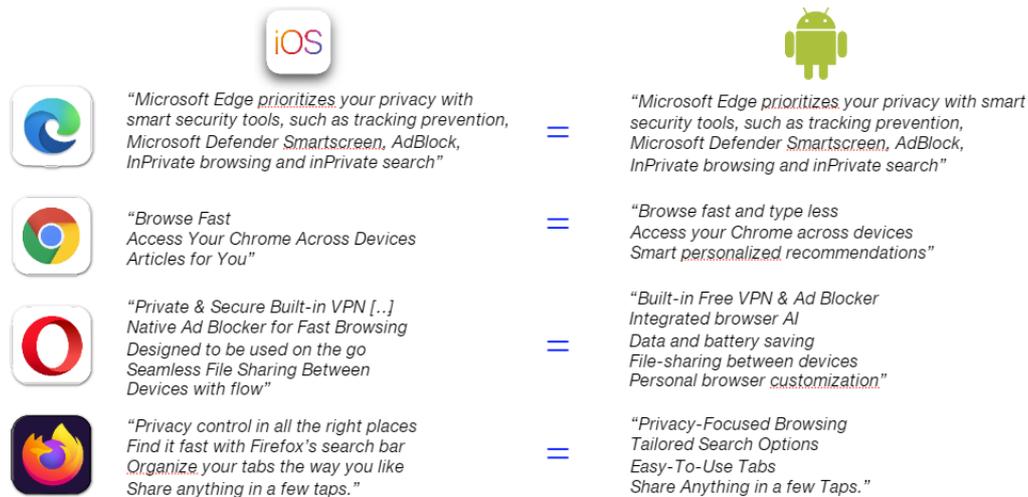
21. Together, this evidence demonstrates that: (1) within iOS, third-party browsers are not hindered from differentiating themselves from Safari or other competing browsers; and (2) the extent of similarities between cross-platform browsers' iOS and Android offerings suggests differentiation on iOS has not been impeded by the WebKit requirement.

2. High pace of innovation

22. Apple continues to invest significantly in WebKit and Safari and has brought many recent improvements to the mobile browsing experience on iOS. Apple has led the way in new feature development, a fact which has been recognised by third parties and commentators. For example, Apple has led efforts to implement switch control, masonry, text box trim, <model>, WebXR, declarative web push, <hr> in <select>, and third-party private tokens. The high pace of recent feature development is inconsistent with the notion that the market for mobile browsing on iOS is not well-functioning and that Apple does not compete strongly with other browsers on iOS or across iOS and other platforms.

3. Lack of switching barriers

23. Apple facilitates competition by making it very easy for users to switch default browsers, remove Safari from the home screen, and add other browsers to the home screen or dock. It also allows third-party browsers to surface prompts and push notifications to prompt users to switch default browsers, features which browsers are utilizing in practice. Furthermore, the CMA's own user survey shows that iOS users not only have confidence in navigating Apple's user interface, but also that their actual experience is consistent with their expectations of a seamless user journey. Most notably:

- 78% of users reported that they could definitely or probably change the default browser setting in their iOS device without assistance.

- 89% of iOS users that had changed their browser said they found it very easy or fairly easy to do so, with only 4% finding it difficult or very difficult.[23]

24. The CMA cannot downplay this evidence by pointing to low levels of switching between platforms. The extent of switching barriers, and hence meaningful switching activity, between platforms cannot be properly evaluated without understanding the underlying levels of

---

[23] These findings are consistent with the findings of the CMA's user survey for MEMS, which showed that a vast majority of iPhone users are fairly or very confident in smartphone technology and using different apps (MEMS Consumer purchasing behaviour in the UK smartphone market, CMA Research Report, Figure 17) as well as in changing settings on their smartphone (*Ibid,* Figure 19).

satisfaction among users. The very high levels of iOS user satisfaction suggest that observed levels of switching are, as would be expected, not artificially low. The evidence indicates that users are aware of, and feel comfortable with, their ability to switch browsers. This is consistent with a well-functioning market and robust competition amongst browsers.

4. *Limited concern from developers*

25. Whilst the working papers spend considerable time iterating individual concerns raised by a small number of developers, the weight of the evidence shows that, overall, developers are satisfied with mobile browsing on iOS. Many of the "concerns" raised in the working papers are outdated and/or factually incorrect (discussed further below in Section V), which gives them no probative value. Further, the survey evidence of web developers shows that they have little concern about the WebKit requirement. And with respect to in-app browsing, app developers (the intended users of in-app browsing technology) are happy with the variety of options that Apple provides.

5. *Users are satisfied*

26. Finally, survey evidence shows that users are highly satisfied with their mobile devices. The CMA's own findings during the Mobile Ecosystems Market Study ("MEMS") show that overall user satisfaction with iPhones is high, with over 9 in 10 satisfied with their device and 74% reporting particularly high satisfaction. Further, Apple users are generally more satisfied than Android users, as can be seen from Figure 2 below.

Figure 2: High user satisfaction rates on iOS

Source: Accent survey commissioned by CMA during MEMS

27. Such positive results are simply not consistent with a poorly-functioning market, particularly a market for high-value and dynamic consumer goods where customers are very well informed, interact with their devices on a daily basis, and have the ability to easily switch devices in response to frequent releases of innovative products.

28. Properly assessed, the evidence set out above shows clearly that there is no basis on which WebKit, or any other aspect of the mobile browsing experience on iOS, could be viewed as having anything but a positive impact on competition.

II.   THE CMA'S APPROACH ACROSS THE WORKING PAPERS RAISES CONCERNS

29. Apple has significant concerns with the CMA's approach to the market investigation thus far, and its substantive assessment of the evidence as addressed below.

A.   The CMA has failed to apply the appropriate framework in its AEC assessment

30. Despite explicit guidance that in an MIR there is "*no presumption that there are market features that harm competition,*"[24] the CMA *presumes* that Apple's WebKit requirement was designed to harm competition by viewing the WebKit requirement in isolation as a "per se" exclusion of competition.[25]

31. Further, the CMA has required that Apple provide evidence to show that its 'justifications' for the WebKit requirement *outweigh* any purported negative effects on competition, in effect reversing the burden of proof. But rather than considering the evidence that Apple provides in support of its legitimate security and privacy justifications, the CMA summarily dismisses it, concluding that "*any such impact may not be sufficient to counteract the negative effects on competition in the browser market*" without evidence to the contrary.

32. For the reasons set out in Section I, the CMA's presumption that competition is excluded in this market is unwarranted. First, Apple's overarching incentives as a device manufacturer are to promote the widest range of high-quality third-party browsing experiences on the platform. As such, it would be entirely self-defeating to integrate WebKit into iOS for the purpose of excluding competition from third-party browser vendors and web app developers. Second, the evidence before the CMA shows positive outcomes for users and developers which are consistent only with a finding that mobile browsing on iOS is well-functioning and reflects competition.

33. The CMA's presumption is also at odds with the documentary evidence – of the more than 50,000 internal documents submitted to the CMA during the investigation and MEMS thus far, the CMA has not identified a single document that demonstrates an intent on the part of Apple to exclude competition. To the contrary, the overarching observable trend is that Apple has steadily *increased* access to its platforms and technologies to the benefit of third-party competition. Important examples for the purposes of this investigation include: (1) open-sourcing WebKit in 2005,[26] after which Google implemented WebKit for Android;[27] (2) supporting web apps on iOS since 2007;[28] (3) launching the App Store in 2008 to third-party apps, including browser apps;[29] (4) releasing the Speedometer performance test in 2014, which would later become an industry-wide benchmark for browser performance;[30] and (5) enabling the default browser setting in iOS 14 to allow users to change their default browser easily and efficiently.[31]

---

[24]   CC3 revised, paragraph 155.

[25]   By so doing, the CMA implicitly adopts as its counterfactual an idealized open platform without any restriction on third-party access. However, this ignores both the harms that could arise from such an approach and the ways in which, as set out above, Apple's approach in fact increases competition by addressing the need for security, privacy and performance regardless of the browser used on the platform. The counterfactual implicitly proposed by the CMA would decrease user choice at the platform level and harm developers who rely on and benefit from Apple's approach.

[26]   https://webkit.org/blog/5718/10-years-of-web-inspector/

[27]   https://om.co/gigaom/webkit/

[28]   https://www.apple.com/newsroom/2007/06/11iPhone-to-Support-Third-Party-Web-2-0-Applications/

[29]   https://www.apple.com/newsroom/2008/07/10iPhone-3G-on-Sale-Tomorrow/

[30]   https://webkit.org/blog/15131/speedometer-3-0-the-best-way-yet-to-measure-browser-performance/

[31]   https://support.apple.com/en-us/118390

34. Ultimately, the CMA's approach reveals that it has not engaged properly with determining the appropriate assessment framework, including the requirement to identify an appropriate counterfactual against which to assess conduct and outcomes on the iOS platform.[32] Without an adequate counterfactual, the CMA has no basis to establish causation between features of the market and any findings it may make about the functioning of competition in the affected markets.

B. The CMA's consideration of the evidence is inadequate

35. Regardless of the deficiencies in the framework applied by the CMA, even on its own terms, the analysis contained in the working papers is inadequate. These inadequacies are addressed in more detail in Sections III to VI. A brief overview is set out below.

36. First, the working papers frequently rely on information gathered and assessed two or more years ago from MEMS. Such information is outdated and not probative of current market realities, particularly in a highly dynamic market such as mobile browsers. In its supplemental response to the Statement of Issues, Apple highlighted a huge number of developments from the past year alone.

37. Second, the working papers rely to a large degree on unsubstantiated assertions made by a subset of developers rather than objective evidence. The CMA appears to take at face value complaints made by selected developers about the WebKit requirement, despite significant evidence provided by Apple to the contrary. By repeating unsubstantiated claims without qualification, the working papers indicate that the CMA is continuing to give credence to such claims over facts and evidence provided by Apple. This is exacerbated by the CMA's failure to analyze how important these features are to developers' ability to compete.[33] Thus, there is no correlation shown between the complaints listed out by the CMA and any effect on competition. In the absence of this, the CMA's emerging thinking that the WebKit requirement has "*significant implications*" for browser vendors[34] must be reassessed.

38. Third, there are many instances throughout the working papers where the CMA highlights concerns that it has never raised with Apple before. This is most problematic in Working Paper 3 on access to features, where more than a third of the 'concerns' raised relate to features that were not previously identified to Apple (and some remain redacted and unidentifiable) and claims regarding documentation and engagement that were not raised during the investigative phase. A further example is the Google security report which the CMA cites to in Working Paper 2. While Apple obviously disagrees with the headline conclusion of that report, it has not been provided with access to the substance of the report to enable it to provide meaningful comments on the reasoning or evidence purportedly underpinning its conclusion.[35] Whilst Apple appreciates that the CMA has not yet reached a substantive view on these issues, it is nonetheless surprising that the CMA has seen fit to publish its emerging thinking on such a range of new concerns without first testing these with Apple or providing Apple an opportunity to comment. Given the overall conclusions reached in the CMA's

---

[32] Published guidance states that the CMA "*has to find a benchmark against which to determine how the market may be judged to be performing*". CC3 revised, paragraph 320. The reforms to the markets regime made in 2017, which require the CMA to consider remedies at the same time as the issue of AEC, reinforce the need for the CMA to identify a robust 'counterfactual' upon which to base its substantive assessment. The counterfactual is a critical part of the assessment framework because it ensures that the CMA's analysis is grounded in market realities rather than being a comparison against an 'idealized' version of the market with perfect competition. "*In the absence of a statutory benchmark, the CC defines such a benchmark as "a well-functioning market" …—ie one that displays the beneficial aspects of competition … but not an idealized perfectly competitive market.*" (Id.)

[33] For example, the CMA lists as a concern background upload and download functionality, even though no developer raised concerns related to this or pointed to any impact on their business.

[34] Working Paper 2, paragraph 6.1.

[35] Apple similarly considers that the citations to the RET2 security advice provide insufficient detail and context to enable it to comment meaningfully on the findings and conclusions of that advice.

emerging thinking, this is a worrying indication that the CMA's analysis is insufficiently impartial and objective.

39. Fourth, the CMA has failed to properly vet the self-interested submissions of some third parties. The CMA is aware that some of the complaints received, particularly from larger participants, have a commercial foundation and are driven by a desire to take advantage of the regulatory process to further their own interests. This is particularly concerning where those interests cut across those of users. For example, complaints from developers who monetize data are clearly at odds with user privacy and Apple's incentives to afford users control over their data. Apple is therefore disappointed to see that the working papers make no attempt scrutinize the motivations behind the concerns raised and to appropriately weigh those against the claims being made.

40. Fifth, the CMA's working papers fail to account for nuanced realities that affect the competitive analysis. For example, the working papers make no attempt to assess what features of concern are Safari features versus WebKit features, and whether they are necessary to support browser functionalities. This is important because Safari features can easily be replicated by third-party browsers using WebKit functionalities, like reader mode, passwords and keychain, and context menus. Apple has pointed the CMA to several examples of this, which clearly demonstrate that competition is not being restricted on this basis. In another example, the CMA bases its analysis of bugs and security issues on a simplistic quantitative comparison, and wholly ignores Apple's representations that prioritization must guide a reasonable assessment of risks from bugs and security issues. Not all issues require remediation or require urgent remediation – Apple focuses on addressing issues by priority and need, not by simply rushing out fixes that may not be warranted or that themselves may pose risks.

41. In Apple's view, the CMA's analysis of mobile browsing is flawed and insufficient. If the deficiencies are not remedied, the CMA risks reaching findings and potentially imposing remedies that ultimately lead to material harm to users (including vulnerable consumers), developers, and competition more broadly.

III. WORKING PAPER 1: NATURE OF COMPETITION IN THE SUPPLY OF MOBILE BROWSERS AND BROWSER ENGINES

42. Contrary to the emerging thinking set out in Working Paper 1, competition in relation to mobile browsers is robust and effective on iOS and across platforms.

A. The CMA significantly underestimates the impact of ecosystem level competition

43. As explained in Section I, Apple's product design decisions, including those with respect to WebKit, are driven by its overarching goal to provide a differentiated customer proposition at the platform level that is anchored in high standards of security, privacy and performance. Apple's incentives in relation to mobile browsing must therefore be viewed in the light of its overall incentives to attract users to purchase devices in the context of robust competition at the ecosystem level.

44. The CMA's finding in MEMS that the Android and iOS ecosystems form a duopoly grossly understates the extent of competition at the ecosystem level. That view is misleading and entirely contrary to Apple's experience.

45. Apple must constantly innovate its hardware and software to keep up with rival OEMs and to attract users to its platform. This includes, for example, recent innovations in AI that it has announced for integration into iPhone. Apple's investments have specifically promoted competition in relation to browser engines and browsers.

46. WebKit and Safari have pioneered innovation, enhanced user choice, and prompted responses from competitors. For example, in 2005, Safari was the first browser to offer a private browsing mode, which is now ubiquitous among browsers. In 2017, WebKit introduced Intelligent Tracking Prevention to limit cross-site tracking; Mozilla later adopted a similar feature in their Gecko engine. Apple continues to introduce new features, innovations and performance enhancements each year to ensure Safari remains an attractive browser option for iOS users. Examples of significant Safari browser innovations include Start Page, Translation, AutoFill, Reader mode, and Private Browsing.

47. Apple competes in this way because the threat of user switching is real and constant. The CMA understates this primarily because it focuses on what it describes as low levels of switching and the previous MEMS finding that "*users typically purchase one 'personal smartphone' which they use as their primary mobile device and this purchase is relatively infrequent*".[36] This is a significantly flawed approach:

    - Apple must constantly compete to attract iPhone buyers. The CMA fails to recognise that UK consumers buy new smartphones on average every 2-3 years. The CMA's own MEMS user survey found that 82% of respondents on iOS and 79% of respondents on Android purchased their current smartphone less than two years ago.[37] Given the ubiquity of smartphones and their role in everyday life, users are constantly assessing and receiving new information on the performance of their device as compared to alternatives. As a result of the high pace of innovation described above, users are attracted to purchase devices with the latest hardware and software, rather than sticking with a single device. The CMA's narrative of users being locked in is contradicted by the frequency with which UK consumers upgrade their mobile devices.

---

[36] Working Paper 1, paragraph 3.20.
[37] MEMS, Consumer purchasing behaviour in the UK smartphone market, CMA Research Report, Figure 10.

- Further, evidence rebutting that narrative is found in the CMA's MEMS user survey, which shows that users are highly satisfied with their mobile devices. In particular, that survey found that overall user satisfaction with iPhones is high, with over 9 in 10 satisfied with their device and 74% reporting particularly high satisfaction. These findings are consistent with survey evidence available to Apple in the ordinary course, which not only confirms that users are on average highly satisfied with their iOS devices, but also shows that the small minority who are not satisfied have a higher propensity to switch.

- Finally, the CMA's approach fails to recognise that, as described above in Section I, low levels of switching do not imply low levels of competition. Users stick with Apple devices not because they are locked-in and face barriers to switching but because the products and services that Apple provides are well designed and meet their needs.[38] This is what one would expect in a competitive market where differentiated products are offered, and consumer preferences are relatively stable over time.

48. To the extent that the CMA relies on other findings of MEMS[39] for its view that competition between the Android and iOS systems is limited, it is also inconsistent with the evidence Apple has submitted:

- The CMA's finding that Apple and Android devices are clearly segmented on price is inconsistent with the evidence before the CMA. This includes IDC data relied on during MEMS, which shows that various manufacturers of Android devices, including Samsung, OnePlus, and Huawei, are active in the high-end segment with products as expensive or more expensive than Apple's. Further, Apple has previously illustrated in submissions that there are competing Android devices at each price point at which Apple devices are sold in the UK. Apple has also stated that it constantly monitors and benchmarks its performance in the UK against rivals, on a weekly basis, including in terms of "total cost of ownership", with that monitoring prompting it to respond with its own campaigns and initiatives to ensure that carriers promote attractive offers for Apple devices.

- The CMA also misinterprets evidence that customers switch more often from Android to Apple than the other way around. The clearest and most intuitive explanation is simply that Apple offers more popular products, so users have little reason to switch back to Android. This is certainly the view that accords with the survey evidence showing very high satisfaction rates for iOS device owners.

49. On balance, it is clear that the extent of competition at the ecosystem level between iOS and Android platforms is significantly higher than the CMA appreciates,[40] which means that the CMA has considerably underestimated the impact that this has on Apple's incentives and conduct in relation to mobile browsing.

---

[38] The CMA's finding in MEMS that users would be deterred from switching by perceived switching barriers is based on an implausible assumption and is thus not robust. The MEMS final report (paragraph 3.92) recognises that "Actual barriers faced by those switching…were lower than the perceived barriers among users that did not switch when purchasing a new smartphone…." As the CMA itself notes (paragraph 3.88), it is "*relevant to consider actual costs because they are likely to reinforce perceived barriers to switching if or when users learn of them, from personal or second-hand experience.*" It is therefore implausible to assume that perceived barriers to switching could be persistently higher than actual barriers to switching.

[39] See Working paper 1, paragraph 3.20 and Working paper 5, paragraph 5.44.

[40] See also evidence that switching is continuing to increase: https://9to5mac.com/2024/07/24/android-users-switching-to-iphone-hits-5-year-high/.

B.   Apple's approach to mobile browsers reflects user demand in a well-functioning market

50. The CMA's qualitative user survey reveals that on the whole users are satisfied with their browsers and browser engine options on iOS. Most notably, the CMA reports that "*respondents felt that there is adequate choice of browsers available to them*" and that "*respondents were typically able to find and download alternative browsers*".[41] These findings demonstrate that the broad range of browser options available on iOS is meeting user demand and that to the extent users see value in switching browsers they are able to act on that desire successfully. These positive outcomes are consistent with the reality set out in the previous section that mobile browsing on iOS is well-functioning and is characterized by healthy competition.

1.   Low salience does not equate to a lack of competition

51. The CMA draws an unwarrantedly negative inference from the survey finding that browsers are a 'low salience' component of the ecosystem for users.

52. The findings referred to above, namely that "*respondents felt that there is adequate choice of browsers available to them*" and that "*respondents were typically able to find and download alternative browsers*" means that a lack of salience cannot be assumed to reflect a lack of competition. On the contrary, users are aware of competing browsers (on average, the CMA finds that users recognise around 5 different browsers when provided with a list of the top 15[42]) and are able to switch to them should they wish to do so.

53. In this context, the fact that users do not think frequently about browsers is more likely to be a further indicator of positive market outcomes. The primary aim of browser usage is to enable users to browse the web easily and seamlessly; to a large degree a lack of user engagement with browsers indicates that they are working exactly as intended. To this end, the CMA's qualitative user survey finds that users tend to switch browsers when they experience specific compatibility or performance issues,[43] indicating that users are more likely to engage with alternative browsers when they have negative, rather than positive, experiences with their existing browser. And, when they do so, as has been illustrated above, they can easily choose from competing alternatives.

2.   Users demand security and privacy

54. Apple's approach to browsers, including its product design decisions for WebKit and Safari, is specifically intended to meet user demand for devices that are secure, private and perform well.

55. Apple's ordinary course survey evidence shows that the vast majority of users place a high priority on the parameters of privacy and security. Factors such has "Security and privacy of your information" are consistently ranked as "extremely important" by the majority of iPhone buyers and iPhone users.

56. The CMA relies heavily on findings from its survey that appear to suggest that security and privacy are of significant importance to a much smaller proportion of respondents. Apple considers that the difference in results is likely to reflect differences in the survey design more than in actual consumer preferences. For example, unlike the ordinary course Apple surveys where users are afforded the opportunity to rank all features as important or otherwise, the CMA surveys require users to select up to five features as important. This artificially reduces

---

[41]   Working Paper 1, paragraph 2.16.
[42]   Working paper 5, paragraph 2.38(a).
[43]   Working Paper 1, paragraph 2.16(d).

the ranking given to features that are important but are less eye-catching or front-of-mind.[44] Apple notes that, during MEMS, the pilot survey carried out by the CMA did not limit respondents' options in this way and found that security and privacy were ranked at a level consistent with the Apple surveys (89% and 91% of respondents stated that privacy and security were important or very important, respectively).

57. Additionally, the CMA's survey finds that 33% of iPhone buyers stated that brand was by far the most important factor in driving their purchase decision. The CMA itself recognizes in its working papers that users may treat brand as synonymous with certain factors that are associated with that brand.[45] Given Apple's privacy-by-design approach and reputation as the most secure platform, it follows that consumers may be associating the Apple brand with these factors.

58. In any event, regardless of whether the CMA ultimately chooses to favour its own survey findings over Apple's ordinary course survey findings, it is undisputable that Apple's surveys provide it with longstanding and consistent feedback as to the importance of security and privacy for users. It is on the basis of this feedback (not the CMA's findings) that Apple has made, makes, and will make decisions on how to compete. Thus, it is clear that Apple's approach is driven by the need to meet what it sees as key consumer demands and parameters of competition. The CMA cannot ignore this when considering Apple's incentives and conduct, as well as market outcomes (not least given that Apple's success as a device manufacturer demonstrates that it does, in fact, know what its users want).

3. Apple's integrated approach is important to competition

59. The CMA's analysis fails to consider whether and on what parameters users may value competition between browsers and browser engines in the context of an integrated platform such as iOS. This stems from the CMA's failure to recognise the appropriate framework for its assessment and instead to proceed on the assumption that the WebKit requirement excludes competition to the detriment of developers and users.

60. In Apple's view, the key survey findings set out above reveal that the current level of competition between browsers on iOS, built upon the trusted platform of WebKit, is already leading to positive user outcomes. There is no suggestion in the survey evidence (or indeed from users otherwise) that they would value greater "competition" at the browser level if that meant a "trade off" against the levels of security and privacy afforded by Apple.[46]

---

[44] This might especially underestimate the role of features that are of high (but not extremely high) importance to a wide range of buyers. Imagine a survey design that only asks for the most important purchase influencing factor and suppose that half of the population values the camera extremely highly and the other half values the display extremely highly. All consumers have a high valuation for privacy, which is slightly below the valuation of the camera and the display. In such a study half of the survey respondents will name the display as the top feature and half the camera but, despite the high overall value of privacy, no one would name privacy features. In an alternative design, where participants are asked about each feature separately (like in Apple's studies), everyone would respond with a high valuation of privacy. This is consistent with users having a general baseline expectation of security and privacy, meaning that in the context of assessing their purchasing decision they choose to prioritize other factors.

[45] See, for example, Working Paper 2, paragraph 5.47(c).

[46] The CMA also fails to consider the impacts of its implicit objective to open up the iOS platform at the browser engine level. As noted in Section II, such an objective would undermine competitive differentiation between the two ecosystems meaning wider consumer choice on a high-salience would be reduced, in favour of the theoretical possibility of increased browser competition, which the CMA's own survey finds to be a low-salience or low value issue for users. By contrast, the harm at the ecosystem level is clear. In particular, consumers who prefer the overall performance and quality of the ecosystem would lose that choice.

C.   The CMA overstates the challenges for web designers and browser vendors

61. A key finding of the CMA's developer survey is that web development is becoming easier over time. Notably, developers report that increased functionality in browsers has played a role in facilitating the development process. These findings are consistent with Apple's submissions throughout this investigation that WebKit is characterized by a high pace of development, provides access to cutting edge feature support without compromising performance, privacy, or security, and is, wherever practicable, responsive to the demands from developers for additional feature support. They are further consistent with the reality that Apple has every incentive to provide developers access to the latest software functionality and features, ensuring they can create great apps, bolstering the quality and robustness of Apple's app ecosystem.

62. Notably, the evidence base does not support many of the claimed "concerns" that developers have raised. As discussed in detail below in Section V, in many cases the concerns have already been addressed or do not arise because developers can, if they so wish, create equivalent features. Whilst Apple is committed to providing as wide a range of functionality as possible to developers in order to facilitate browser competition on iOS, it is not Apple's role to offer to third parties every single feature that developers may wish to have but do not want to invest in creating in-house, nor should it be.

63. Therefore, as Apple has noted in Section II above, the CMA should recognize that where the subjective views of particular developers are inconsistent with technical/objective evidence, the latter should be afforded greater weight as the former may be driven by commercial interests.

D.   The CMA misunderstands Apple's role in standards bodies

64. The CMA's cursory assessment of web standards does not take account of the benefits that Apple's contributions to such standards bring to developers in terms of security, privacy, and functionality. While the CMA has chosen not to focus on the role and impact of web standards for browser functionality Apple's significant contribution to this material factor should not be overlooked:

- WebKit has pioneered various new features and demonstrated leadership in supporting web standards. For instance, Apple has recently announced support for WebXR, a ground-breaking new feature which allows developers to provide immersive experiences on the web utilizing virtual reality capabilities.

- Apple takes great care when contributing to new standards, given the significant implications that can arise for stakeholders across the web community when implementations are sub-optimal. Apple has for example stated that "*if we find that features and web APIs increase fingerprintability and offer no safe way to protect our users, we will not implement them until we or others have found a good way to reduce that fingerprintability. We continue to have open discussions with other browser makers through the web standards process, many of whom share these concerns.*"[47] This is an important function, which is markedly different from a '*launch first; test later*' approach that others may employ, which can subject developers and users to additional harms and costs when problems arise. Notably, Apple is not alone in taking this approach; indeed, Mozilla has typically taken a similar stance to Apple in relation to the implementation of features.

- Apple has also improved industry benchmarking generally by creating testing suites and then providing them to third parties. For example, the Speedometer suite was

---

[47]   https://webkit.org/tracking-prevention

developed by Apple and is now widely relied on by third-party browsers to test and improve the performance of their own offerings.

65. The contributions described above are significant and merit further consideration, in particular in circumstances where web developers generally have to work based on the lowest common denominator, and so best practice is to design on the assumption that not all APIs are present. This means that Apple's efforts to improve web standards have had a material, positive impact on developers' ability to innovate in practice.

E.   Apple's comments on market definition and share of supply

66. Apple submits that the CMA's indicative markets should be expanded or, in the alternative, the CMA must not underestimate the extent to which "out of market" competition is a constraint.[48]

67. First, as is apparent from the comments above in Sections I and II, mobile browsers compete strongly across iOS and Android devices. Safari competes most directly with other dedicated browsers that offer open-ended web searching and browsing experiences, including those on the Android platform. The importance of competition across platforms cannot be underestimated or ignored.

68. Second, there are important competitive interactions between mobile and desktop browsers. As the CMA recognises, despite their differences, mobile and desktop browsers offer many of the same features to users, and cross-device syncing reduces barriers to substitution between mobile and desktop browsers. As a result, even if the two are sufficiently different that they do not form part of the same market, competition from desktop browsers should be adequately taken into account in the CMA's assessment of mobile browsing. The working papers appear to pay no attention to this.

69. With respect to the share of supply data presented in working paper 1, Apple is concerned that this is presented on an inaccurate platform-specific basis. As a result, it does not recognise the full extent of competition between iOS and Android platforms in the area of mobile browsing, and provides an incomplete and misleading picture of the extent of competition.

F.   Conclusion

70. As the above sections show, competition in relation to mobile browsers is robust and effective on iOS and across platforms. Apple therefore urges the CMA to rethink its emerging findings in relation to the nature of competition with respect to mobile browsing.

---

[48] As noted at paragraph 3.4, "*market definition is a tool and not an end in itself…The boundaries of the market do not determine the outcome of our competitive assessment of a market in any mechanistic way. The competitive assessment takes into account any relevant constraints from outside the market, segmentation within it, or ways in which some constraints are more important than others*".

IV. WORKING PAPER 2: THE REQUIREMENT FOR BROWSERS OPERATING ON IOS DEVICES TO USE APPLE'S WEBKIT BROWSER ENGINE

71. Contrary to the emerging thinking set out in Working Paper 2, the WebKit requirement fosters competition between mobile browsers, supports the development of web apps, and leads to positive user outcomes overall on the key parameters of security, privacy, and performance.

A. The WebKit requirement ensures a high standard of security, privacy, and performance, intensifying competition at the ecosystem and browser levels

72. In Sections I and III above, Apple has explained that users demand high levels of security, privacy and performance from their devices. In the context of robust competition at the ecosystem level, Apple has developed a differentiated approach to its platform which leverages tight integration of hardware and software in order to meet these user demands. The WebKit requirement is essential to Apple's ability to deliver that customer proposition and compete effectively. This is demonstrated by the following:[49]

   • The WebKit requirement is a key pillar of iOS platform security that goes back to the original iPhone. WebKit harnesses a range of powerful technologies and strategies to secure the iOS platform (and all mobile browsers on that platform) against web-based attacks and to limit the device attack surface and scope of risk.[50]

   • From a privacy perspective, WebKit enforces privacy protections across all apps through a suite of technologies and strategies, including third-party cookie blocking by default, storage and service worker partitioning, requiring user permission for websites to access the device orientation or motion APIs, and preventing fingerprinting of device microphones or cameras via the WebRTC API. In striking contrast to Apple's approach, Google recently dropped its equivalent protection against third-party cookies, further emphasizing its lower prioritization of user privacy;[51] its move to do so has been criticized by other regulators in the UK.[52]

   • With respect to performance, WebKit is also optimized for high performance on mobile devices.[53] Apple has pointed the CMA to significant features introduced in WebKit that have been instrumental to performance enhancement, such as multi-process architecture which improved responsiveness because it allowed multiple tabs to run and allowed users to interact with tabs even if a website was slow. The

---

[49] For more detail, see also Section IV.D. below.

[50] See paragraph 115 below. The benefits of a security architecture which is designed to minimize risks have been brought sharply into focus by the recent widespread outages experienced by devices running Microsoft Windows, which has been called "the world's biggest IT failure". Those crashes were caused because the security software in question, Crowdstrike, had access to the deepest levels of the OS, meaning that when a faulty update was pushed to 8.5 million devices it rendered them unusable. It is notable that Microsoft has stated that it cannot legally wall off its OS in the same way Apple does because of commitments it made to the European Commission in 2009 to give makers of security software the same level of access to Windows that Microsoft gets. See: https://news.sky.com/story/crowdstrike-chaos-what-are-the-lessons-from-the-worlds-biggest-it-failure-13181065; https://www.wsj.com/tech/cybersecurity/microsoft-tech-outage-role-crowdstrike-50917b90.

[51] https://www.forbes.com/sites/zakdoffman/2024/07/25/new-google-chrome-warning-microsoft-windows-10-windows-11-3-billion-users/.

[52] See Information Commissioner's Office, ICO statement in response to Google announcing it will no longer block third-party cookies in Chrome (23 July 2024), https://ico.org.uk/about-the-ico/media-centre/news-and-blogs/2024/07/ico-statement-in-response-to-google-announcing-it-will-no-longer-block-third-party-cookies/ ("*We are disappointed that Google has changed its plans and no longer intends to deprecate third party cookies from the Chrome Browser…[I]t has been our view that blocking third party cookies would be a positive step for consumers.*").

[53] See further paragraphs 92 to 93.

WebKit requirement ensures that all browsers benefit from this high level of performance.

73. WebKit also fosters competition at the browser level by providing: (a) the basis on which Safari continuously innovates and brings new features to the market; and (b) a trusted platform on which third-party browsers can build and compete.

74. Apple has provided multiple examples of innovations enabled by WebKit. These include Start Page, Translation, AutoFill, Reader mode, and Private Browsing. Other WebKit innovations include Intelligent Tracking Prevention, Private Click Measurement, and Dark Mode. Apple often introduces new features that are later introduced in competing browsers. For example, following Safari's introduction of Reader mode, a similar feature was launched by Mozilla Firefox and by Google Chrome. Similarly, after Apple's introduction of Private Browsing, Google Chrome followed with Incognito. Other browsers, including Opera and Microsoft Edge, also introduced private browsing features.

75. WebKit provides third-party browser developers with the functionalities and features necessary to bring competitive user-facing features to their browsers on iOS. As set out further below, it provides a transparent and effective platform on which browser developers can create differentiating features and innovations. Apple has provided the CMA with multiple examples of functionalities that it introduced into WebKit that enable such development. These include functionalities specifically requested by developers, such as Web Push and Badging, Add to Home Screen, Service Workers support, Web Assembly, HTML5 Media support, Offscreen Canvas, Managed Media Source API, Wide Gamut Color in CSS (P3), modern tools for layout, tools for complex CSS architecture, and modern typography.

76. At the same time, the requirement to use WebKit as the sole rendering engine for browsers on iOS ensures a stable and consistent level of security, privacy, and performance for all browsers. Browser developers do not have to expend time and resources ensuring that they stay on top of the latest security threats, malware issues, and fraud concerns. Nor do they have to concern themselves with ensuring that the version of the browser engine they are using will not negatively impact how their browser performs. This saves browser developers (particularly smaller developers) significant cost and engineering effort.

77. The CMA's implicit objective to open up the iOS platform at the browser engine level would undermine competitive differentiation at the ecosystem level, reduce overall performance and quality, and deny users choice. It would also reduce the significant benefits that WebKit brings to browser developers and to competition at the browser level, as noted above and described further in Section IV.B below.

B.   WebKit brings significant benefits to browser vendors

78. There can be no serious doubt that Apple, through WebKit, provides browser developers with the tools and functionalities they need to develop competitive browser offerings on iOS. Browser vendors have access to more than 250,000 APIs on equal terms, as well as an array of other functionalities that allow them to build highly performing and differentiated browsers for iOS devices. WebKit permits for substantial differentiation between browsers, allowing developers to build features and interfaces on top of WebKit, while upholding Apple's privacy and security protections. Although Apple requires that all browsers use WebKit to render web content, it does not dictate what features ship on third-party browsers. Other developers that control third-party browsers are free to build features into their browsers that are not available in Safari. The benefits that WebKit brings (which, thanks to the WebKit requirement, are available to <u>all mobile browsers on iOS</u>) are sufficient to remove any concerns regarding an Adverse Effect on Competition (AEC) in mobile browsing, regardless of the link between mobile browsing and device competition. While consumers may not directly attribute enhanced security to mobile browsers (or the browser engine), there is ample evidence confirming that security and privacy are very important factors in

consumer choice of device and that the browser engine is a key determinant of device security and privacy.

1.   WebKit's open source nature provides a transparent basis on which to develop competitive browser features

79. The CMA's analysis of the impact of the WebKit requirement on developers appears to be premised on a fundamental misunderstanding of the open source nature of the project and Apple's role as WebKit's steward.

80. As Apple has explained, WebKit has always been open source, and in fact Apple's approach to maintaining the infrastructure for that project is a collaborative one:

- WebKit began as a volunteer-supported community project, benefitting from input from a variety of contributors from a range of organisations, and it has been adopted by a range of vendors to render web content. Apple has expended considerable effort to expand and support the project. In 2023, WebKit had approximately 140-150 third-party developers contributing to the WebKit browser engine.

- Third parties have full access to review the technical details of the WebKit open source repository for the WebKit Open Source Project. Apple builds the WebKit framework from the code in the open-source repository, meaning that anyone can see what changes are being made to WebKit and anyone can contribute. Any contributor can publicly discuss new features, contribute code, and file bugs.

- WebKit's model involves two levels of developers, many of whom are non-Apple developers: committers and reviewers.[54] These are selected by the set of existing WebKit reviewers through a nomination process, based on their experience and ability to collaborate with other reviewers. Which features are ultimately adopted as WebKit code in the open source project is generally decided by a consensus of reviewers.

- To the extent contributors including third parties maintain a port of WebKit, they are responsible for their individual ports of WebKit and each will have final say on what features are enabled on the port for which they are responsible. For example, Igalia is responsible for the GTK port, and Sony is responsible for the PlayStation port.

81. Apple has further explained that its role as steward of the WebKit project facilitates rather than constrains the ability of third parties to contribute and request new features. Apple does not unilaterally dictate the features supported by the project, nor does it dictate which features ship on third-party ports of WebKit or WebKit-based browsers. Apple conducts the critical functions of the 'steward', ensuring that new code introduced into the source code repository is properly tested and vetted to the standards of a commercial enterprise before being shipped.

82. As the above demonstrates, Apple has always supported WebKit as an open source, community-led project as that is the best way for it to encourage developers onto the platform as a means of promoting a rich app ecosystem. Indeed, it would be entirely self-defeating for Apple to use WebKit as a way to control or stymie innovation by developers, as that would likely drive away browser vendors and web app developers. It would also be contrary to the CMA's own view that browser engines compete for browsers and web developers.[55]

---

[54]   See https://webkit.org/commit-and-review-policy/.
[55]   Working Paper 1, paragraph 2.43(b).

83. From a competitive perspective, therefore, the WebKit framework is fully transparent and freely available to developers.

> *2. Browser vendors can build highly performing and differentiated browsers for iOS devices*

84. Contrary to the views expressed in the working paper, third-party browsers on iOS in fact achieve a significant degree of differentiation, enabling them to compete effectively with Safari. Apple has conducted an analysis of mobile browsers' product pages on the UK App Store—where developers market their apps directly to users—to assess the extent of differentiation between mobile browsers on iOS. The results of this analysis very clearly show significant differentiation, illustrating the capacity of the tools Apple provides to developers and the benefits of the WebKit platform.

85. Table 1 below provides a summary overview of the different unique selling points (or 'USPs') highlighted by mobile browsers on iOS in their marketing and taglines, including in particular the use of AI, which is already an important dimension of competition. Notably, contrary to the CMA's analysis that the WebKit requirement precludes active competition on enhancements to privacy and security, this is one of the principal parameters on which browsers choose to compete on iOS. A detailed overview of each of the parameters that browsers call out in their marketing and the associated features is attached at Annex_Apple WP response_01.

Table 1: Focus of differentiation by mobile browsers on iOS

USPs / Browser	Customis-ability	Privacy/ Safety	Speed	Ease of use/ "Smart Features"	Ad-blocking	Gaming capabilities	Sustain-ability	AI
Arc				X				X
Brave		X	X		X			X
Chrome	X	X	X					
Dolphin	X		X	X				
DuckDuckGo		X						
Ecosia		X	X		X		X	X
Firefox	X	X		X				
Firefox Focus		X	X		X			
Microsoft Edge	X	X		X				X
Opera	X	X	X		X			X
Opera GX	X					X		X
Phoenix		X	X		X			
Via Browser			X	X	X			
Vivaldi Browser	X	X	X		X			
Yandex		X	X	X				

86. The CMA's analysis of the impact of the WebKit requirement on security and privacy relies heavily on the view that WebKit somehow limits the ability of browser vendors to improve their browsers by adding additional security or privacy improvements or features to their browsers on iOS devices.[56] This is not borne out by the evidence.

---

[56]   Working Paper 2, paragraphs 3.13 and 3.15.

87. Annex_Apple WP response_02 presents an overview of security and privacy features offered by browsers on iOS that highlight security and/or privacy in their marketing. This reveals a material and diverse array of features, which provide even higher protections than those offered through WebKit. As noted above, this reflects only a subset of all browsers on iOS, indicating that the total universe of security and privacy features available on iOS is likely to be richer.

88. In light of the above, it is unsurprising that allegations raised by individual complainants[57] that they are not able to implement certain security, privacy and user-facing features on iOS are either inaccurate or misleading. For example, it is misleading to suggest that Microsoft's Enhanced Security Mode would bring better protections to users on iOS. The features contained within Microsoft's Enhanced Security Mode, which does not appear to be available on Android but is limited instead to Microsoft's desktop browser, effectively overlap with the protections of Lockdown Mode by disabling JIT, which Apple has already implemented within WebKit. Consequently, all browsers (including smaller browser developers that do not have the same resources as Microsoft) on iOS benefit from such protections, without the need to expend additional resources.

89. The remaining issues raised by Microsoft appear predominantly to relate to enhancements to content security policy. Apple already supports multiple variations of the content security policy standard.[58] As with all developer suggestions, if Microsoft has enhancement requests relating to content security, Apple would welcome such feedback.

3. Innovation by browser vendors on iOS is comparable to that on Android

90. Not only do browser developers on iOS succeed in differentiating themselves from Safari and from each other—dispelling the notion that the WebKit requirement hampers browser competition on iOS devices—they also compete across platforms on essentially the same basis (with some differentiation between platforms). The headline features browser developers market on iOS are similar and in many cases identical to the features they offer on other platforms, including platforms where they use non-WebKit engines. Highlights of this have been provided in Section I above. A detailed overview of each of the features highlighted in product pages across the UK App Store and Google Play Store is attached at Annex_Apple WP response_03.

---

[57] While Brave claims that it cannot implement various privacy features on iOS (Working Paper 2, paragraph 3.14(a)), Brave's own public statements indicate that it has, in fact, done so and that the difference between iOS and other platforms are fewer than it suggests. For example, in an article from August 2022, Brave stated that "*Starting in version 1.38, Brave iOS included the kinds of fingerprint randomization defenses Brave includes on our other platforms.*" (https://brave.com/privacy-updates/20-grab-bag-ios/) It also noted more generally that its latest update "*brings the privacy features on Brave iOS closer into alignment with our other platforms.*" As regards Vivaldi's allegation that it cannot implement its own tools for ad tracker blocking due to ITP being embedded within WebKit (Working Paper 2, paragraph 3.14(b)), recent statements by Vivaldi confirm that it has, in fact, released upgraded ad-blocking tools on its iOS browser. For example, one article notes that "*The latest update to Vivaldi brings new ways for you to personalize the browser, with a customizable button in the address bar and easier Start Page customization. We have also made several improvements to our Ad Blocker and enabled Bookmark nickname matching.*" (https://www.neowin.net/news/vivaldi-68-for-ios-is-out-with-tab-improved-ad-blocker-and-new-personalization/#google_vignette) Apple has also explained that third-party browsers are free to build and implement web extension functionality on top of WebKit, including Ad blockers. With regards to Vivaldi's concerns about access to Reader Mode, Apple would note that Reader Mode is a Safari feature and that third-party browser developers have the ability to implement their own version of Reader Mode if they so choose via WebKit APIs, as reflected in the working paper (paragraph 3.14).

[58] https://webkit.org/blog/6830/a-refined-content-security-policy/.

91. If the allegations raised by browser vendors that they are less able to innovate and improve on iOS than they are on Android[59] had any basis in fact, this would not be the case.

4. *WebKit provides a high performance baseline for all browsers*

92. The CMA has provided no testable evidence and only high-level assertions that "*the WebKit restriction limits the ability of browser vendors to improve the performance of their browsers on iOS, and may therefore decrease the ability of browsers to compete on performance features on iOS*".[60] In several cases, the CMA redacted relevant details. Without the ability to test detailed assertions raised with the CMA, Apple cannot provide important relevant information to the CMA, and the CMA cannot reasonably base findings on mere assertions.

93. Apple has provided the CMA objective technical evidence showing that WebKit-based browsers perform significantly better than browsers based on third-party engines.[61] As other browser engines improve their performance over time, Apple invests significantly to ensure that WebKit-based browsers remain the highest performing browsers available to users. This offers further support that WebKit provides a high baseline level of performance, not just for Safari, but also for all browser apps on iOS.

5. *WebKit promotes accessibility*

94. Mozilla's assertion that the WebKit requirement restricts accessibility features is included in the working paper without supporting detail. As above, in the absence of any ability for Apple to test assertions raised with the CMA, the CMA cannot reasonably base findings on them.

95. In any event, Apple disputes the notion that WebKit restricts accessibility. Accessibility features are generally iOS-based, meaning that all apps receive the benefits of such features whether they are first-party or third-party apps. Examples of such features which WebKit supports include the "prefers-reduced-motion media feature", which can be used to serve alternate animations that avoid motion sickness triggers experienced by some site visitors, such as those suffering from vestibular disorders.[62] WebKit also implements Accessible Rich Internet Applications (ARIA), which is used by major web applications to improve accessibility and general usability.[63]

96. Apple has previously pointed to third-party evidence submitted to the CMA stating that accessibility apps function better and are easier to design on iOS. In particular, it highlighted[64] the submission of a developer of assistive technology in response to the CMA's consultation on the market investigation reference proposal who noted the significant benefits of Apple's integrated approach for developers of accessibility apps.[65] Apple respectfully requests that

---

[59]   Working Paper 2, paragraph 3.10.
[60]   Working Paper 2, paragraph 3.17.
[61]   See https://www.apple.com/safari/, Footnote 1.
[62]   https://webkit.org/blog/7551/responsive-design-for-motion/.
[63]   https://webkit.org/blog/3302/aria-and-accessibility-inspector/.
[64]   See paragraph 67 of Apple's response to the Statement of Issues.
[65]   "*The advantage of the locked-down nature of iOS is that it is consistent and under greater user control. Apple can mandate that all applications work a certain way, and their assistive technology tools can then work with the applications more effectively. For example, a blind person using the VoiceOver program to make their iPhone talk will only ever encounter web content on iOS displayed by the Safari engine because Apple forbids all other browser engines on iOS (for example, Google Chrome). That means that VoiceOver can do a better job than (say) a similar program on Android or Windows because the web content is consistently presented: if a button is always a button then VoiceOver can always say "button" and the blind user can use the webpage. If a different browser engine comes along and displays something that looks like a button but is, under the hood, something quite different, then the blind VoiceOver user might hear not "button" but "picture" or "link" or nothing at all. The consistency of iOS is beneficial to assistive technology and assistive technology users with disabilities ... if a more open alterative was better for people with disabilities then Android might have been the leading platform for people with disabilities - but it is not better, and locked-down iOS has remained the leading platform for people with disabilities.*"

the CMA place greater weight on the submissions of developers such as the developer referred to above, whose business is the provision of assistive technologies and apps for the benefit of vulnerable users, rather than those of large browser vendors whose submissions may be influenced by differing incentives and commercial aims.

6. *Apple routinely ships bug fixes and features in a timely manner*

97. Apple focuses on adding features that will improve the user browsing experience and on refining existing features, without resulting in a net cost on system performance or posing significant risks to security or privacy. To the extent that Apple is able to identify a safe and secure way to implement new features without compromising device performance and security or user privacy, it will do so within as short a timeframe as is reasonably possible.

98. More generally, Apple has explained to the CMA that the implementation of features is not a race to develop the longest list. It involves the consideration of a number of complex, technical issues, as well as an array of possible user risks and other potential unintended consequences. It is therefore necessary to look beyond crude quantitative metrics such as the number of features or the date of implementation.

99. Apple has highlighted the issues that may be caused by the failure to ensure high-quality implementations of new features. A paper by Karami, et al., on the implementation of Service Workers find that all major implementations, save WebKit's, were lacking.[66]

100. The CMA has provided no testable evidence that Apple delays implementation of features or bug fixes. The limited 'evidence' the CMA cites is redacted.[67] As above, since Apple has not had an opportunity to review or scrutinize meaningfully the assertions referred to, they can have no probative value. It is, in any event, contradicted by the evidence of another developer comparing iOS favorably with Android when it comes to engagement with respect to bugs.[68]

7. *WebKit reduces browser development costs*

101. The CMA's analysis in the working paper appears to take at face value claims made by browser developers that the WebKit requirement has caused them to incur additional costs, leading to delays in the implementation of new products or features.[69] Apple strongly contests any such claims.

102. First, the CMA fails to recognize that developers' perceptions around costs may be driven by reasons unrelated to the WebKit requirement. It is indisputable that software development of any nature gives rise to some degree of costs and that these may stem from decisions on resourcing and approach taken by the developers themselves. Other costs can be driven by developers' seeking to develop for multiple operating systems, rather than the use of WebKit specifically.

---

[66] See Karami et al "Awakening the Web's Sleeper Agents: Misusing Service Workers for Privacy Leakage", Network and Distributed Systems Security (NDSS) Symposium 2021, submitted to the CMA on 3 November 2021 ('*Furthermore, all Chromium-based browsers and Firefox are vulnerable to the timing-based attack. Our attacks are not applicable against Safari as it correctly isolates SWs (i.e., a SW cannot be activated by an iframe on a third-party website). Interestingly, since iOS restricts browsers to use the Webkit browser engine, they all behave the same as Safari. While browsers on iOS are not vulnerable to our attacks, they are on MacOS.*' Karami et al further observe: '*Overall, our work presents a cautionary tale on the severe risks of browsers deploying new features without an in-depth evaluation of their security and privacy implication*s.').

[67] Working Paper 2, paragraph 3.29.

[68] Working Paper 2, paragraph 3.30.

[69] The CMA also misleadingly includes in relation to WebKit complaints such as that from Gener8, a desktop browser, even though Gener8 notes that the concern it raises means it does not currently offer a mobile browser on Android either.

103.   Second, Apple already invests significant resources to lower the costs of development on its platform. Notwithstanding the significant existing level of support, developers have an obvious self-interest in seeking to have Apple build for them even more technology and offer even more support for free. The CMA should scrutinize such claims accordingly.

104.   In any event, and contrary to the CMA's emerging thinking, the evidence before the CMA reveals that Apple has provided developers with an array of technologies, tools, and documentation, all of which lower costs for browser development on iOS:

- WebKit offers developers access to Web Inspector, a development tool that allows any developer to view a webpage source, DOM hierarchy, script debugging, and other rules in the source code for any Apple website. This tool shows how much battery life the website consumes, as well as how much memory, or CPU, the website uses.

- Apple offers Xcode, which provides app developers with everything they need to develop, test, and distribute apps on iOS. App developers can leverage the Swift programming language and SwiftUI framework, which provide enhanced editor features. They are also able to test and deploy from Xcode Cloud to TestFlight and the App Store.

105.   To ensure developers capture the benefit of these efforts, Apple performs extensive outreach through a variety of channels, including blog posts, iOS and Safari release notes documentation, WWDC homepages, developer sessions and forums, webpages, and the Beta Software Program.

C.   The WebKit requirement supports competition by web developers

106.   The CMA's analysis shows that web developers on balance do not have concerns about the WebKit requirement. The general view expressed by respondents to the CMA's qualitative web developer research showed little awareness or concern around the WebKit requirement.[70] The CMA is correct to recognize the contrast between this evidence, which is drawn from a wide body of third parties, and the allegations raised by individual complainants.

107.   Further, many developers did not consider that the WebKit requirement was an issue with respect to compatibility[71] or that the cost of ensuring compatibility was burdensome or linked to issues involving WebKit.[72]

108.   Importantly, the CMA's developer research found that for respondents, iOS was perceived as being "*more secure than Android*".[73]

109.   The CMA nonetheless devotes considerable space to complaints by a small group of respondents. This includes concerns that Apple 'lags behind' in terms of feature support. As set out above in section IV.B, the evidence Apple has provided demonstrates such concerns to be unfounded. In particular, Apple has provided evidence showing significant new and expanded feature support as well as improved performance. This evidence shows that WebKit is in fact characterized by a high pace of development.

110.   Similarly, the assertions raised by 'some' developers that Apple is slow to resolve security issues and other bugs are either generic, inaccurate, misleading, or rebutted by objective evidence which Apple has submitted to the CMA:

---

[70]   Working Paper 2, paragraph 4.33.
[71]   Working Paper 2, paragraph 4.18.
[72]   Working Paper 2, paragraph 4.24.
[73]   Working Paper 2, paragraph 4.41.

- Apple typically ships a software or security update around every six to seven weeks but can also issue Rapid Security Responses or "point release" updates on a significantly faster cadence – within days of learning of a significant security threat. These vehicles provide ample flexibility for addressing security concerns of varying levels of severity.

- Apple actively monitors requests to fix bugs and security issues to ensure that they are screened and actioned in a timely manner and to allocate resources to address unscreened requests. The sequence in which Radars are resolved depends on a number of factors, including the seriousness of the issues at hand, the complexity of designing and shipping a solution, and the other demands on the engineering teams. A simple quantitative comparison against Android does not reflect the actual security benefits of Apple's qualitative approach. Importantly, it is ultimately irrelevant how frequently Google Chrome or the Chromium engine ships updates/bug fixes; the relevant consideration is whether the many browsers that run on Chromium patch the engines in their apps on a regular cadence, and Apple has demonstrated that this is not the case.[74]

- OWA's analysis about CVE figures is contradicted by the CMA's own analysis of CVEs, with the latter finding that Chrome and Firefox have had significantly more CVEs than Safari in recent years. OWA also asserts that Apple does not update iOS on older devices, which is demonstrably inaccurate. Third-party sources confirm that Apple supports older devices for many years longer than Android, "*with frequent software updates even years after its release. Companies that develop Android phones are making their phones obsolete in just two to three years*".[75] Indeed, a recent article by statista.com describes Apple's iPhone 6 as "*the first generation to see a seventh iOS version, making it the technological equivalent of a tortoise outlasting generations of humans*".[76]

111. Whilst the working paper indicates that the CMA has concerns that "*WebKit has lagged other browser engines in support for web apps*" and that "*this has created additional costs for developers, particularly those with specific business models reliant on exploiting web apps*",[77] it is clear that these concerns are not widely shared. In particular, the CMA recognises that Apple's support for web apps "*has improved recently*" and that "*not all developers were concerned about it, and some either raised security concerns around web apps or considered web apps would not be effective substitutes for native apps for broader reasons (in particular, the easier discoverability of native apps which are catalogued and discoverable through a curated app store)*".[78]

112. The CMA should give due weight to these views. In particular, third parties (including Mozilla) echo the concerns of Apple regarding the security risks that web apps may pose, including that allowing web apps to access more APIs increases the 'attack surface' and can therefore degrade the current level of security or privacy for those apps and the device at large. Mozilla has 'actively decided' not to provide web apps with access to certain APIs used by native apps due to security and privacy concerns, in similar fashion to Apple. The similarity between the approach taken between Apple and Mozilla, refutes the notion that Apple's approach is somehow indicative of a wider incentive to 'protect' the App Store business.

---

[74] See paragraph 120 *et seq*.
[75] https://www.techtimes.com/articles/263017/20210717/android-vs-apple-why-phones-limited-support-compared.htm. Additionally, Which? has provided detailed information on the duration which Android and iOS phones, which corroborates this finding. See: https://www.which.co.uk/reviews/mobile-phones/article/mobile-phone-security-is-it-safe-to-use-an-old-phone-a6uXf1w6PvEN.
[76] https://www.statista.com/chart/5824/ios-iphone-compatibility/.
[77] Working Paper 2, paragraph 4.37.
[78] *Id*.

113.   In light of the above, Apple submits that the CMA's analysis shows, on balance, that the WebKit requirement does not impede competition between web developers on iOS.

D.   The CMA's conclusions on the impact of the WebKit requirement on security, privacy and performance are incorrect

114.   The WebKit requirement is an essential element of iOS platform security, privacy and performance, and has been applied to all apps in the UK App Store since its launch in 2008. Apple strongly endorses the CMA's emerging view that there are benefits (particularly security benefits) arising from Apple's integration between hardware and software. The WebKit requirement allows Apple to achieve much higher levels of security, privacy and performance than it would otherwise be able to achieve.

115.   Despite this, the CMA's emerging conclusions are that the evidence does not show WebKit requirement results in improved security, privacy and performance. These emerging conclusions are incorrect and fail to take into account the very substantial body of objective evidence, including from third party sources, which demonstrates that the requirement leads to significantly better outcomes on iOS on those parameters.

1.   The WebKit requirement allows for better security outcomes

116.   Despite Apple highlighting to the CMA that security is a critical issue in any assessment of the WebKit requirement, the CMA's analysis of that issue remains strikingly superficial, with the CMA continuing to rely on crude quantitative metrics relating to the number of updates and bugs, which it has previously recognized are of limited probative value. This does not meet the requirement set out in the caselaw, which states that the "*depth and sophistication called for in relation to any particular relevant aspect of the inquiry needs to be tailored to the importance or gravity of the issue within the general context of the [CMA's] task.*"[79]

117.   The WebKit requirement grants Apple "*more control over iOS compared to Google's control over Android, and allows it to limit the addition of browser features that might compromise security*".[80] Apple notes, however, that Apple's approach goes far beyond just vetting new features that might compromise security:

- One of the key benefits of WebKit is that it enables Apple to distribute important security updates to all apps rendering web content on the platform in a single update. In contrast to the centralized model of WebKit updates on iOS, Android's fragmented model abdicates browser engine security to individual developers.

- Apple's engineers work collaboratively across functions during the design, development, testing, and post-release support to detect and prevent potential security vulnerabilities. The tight coordination between different engineering functions plays a critical role in preventing, mitigating, and detecting security vulnerabilities. WebKit forms part of a cohesive model of platform security and supports an array of other methods and tools which Apple uses to secure iOS at every layer: kernel and core runtime, firmware, the hardware on which the operating system runs, applications, and more.

- Apple's security teams also conduct ongoing security analyses designed to find and resolve potential security vulnerabilities before attackers. For example, Apple identifies potential targets for fuzz testing often before development begins, enabling a fuzzer to be built in parallel to development, which would not be feasible in the case

---

[79]   *Barclays Bank plc v Competition Commission* (2009), CAT 27 (paragraph 21); citing *Tesco v Competition Commission* (2009), CAT 6 (paragraph 139).
[80]   Working Paper 2, paragraph 5.18.

of third-party engines, meaning that the onus to conduct such testing would be left to third-party developers, which may not prioritize security to the same extent as Apple.

118.   As set out above, the reason for the WebKit requirement and its success in achieving better security outcomes on iOS is because of the differing incentives between Apple (as platform operator) and mobile browser developers (who offer their apps on that platform). There are enormous negative impacts on the platform, on other developers, and on users resulting from security or privacy incidents, which individual browser developers do not have the incentive to address. Apple, on the other hand, does have this incentive, and the vehicle to do so in WebKit. The WebKit requirement enables Apple to secure a high level of security protection across iOS as a whole.

119.   Contrary to the CMA's emerging thinking that it has not seen clear evidence WebKit leads to improved security outcomes, Apple has pointed to a considerable body of evidence which demonstrates exactly this:

a)   The Android "Patch Gap" problem

120.   Apple has submitted the results of analyses undertaken in February 2023, January 2024, and March 2024 to assess whether mobile browsers available on the Google Play store incorporated browser engines that were up-to-date. [81] Each of those analyses produced similar results showing that many popular browsers rely on outdated browser engines. Most strikingly, the March 2024 UK analysis found that 30 of the 38 most downloaded UK browser applications (78.95%) used an out-of-date engine version. Apple has further provided data showing that as a browser goes increasingly out-of-date, the number of known vulnerabilities and the number of confirmed exploited vulnerabilities rises substantially. For example, a browser running Chromium version 61 in March 2024 (i.e., over six years out of date) would be subject to 1781 known vulnerabilities, 48 of which are confirmed to have been actually exploited by malicious actors.[82] By requiring browsers to use a single instance of WebKit that ensures all browsers benefit equally from security updates, iOS completely avoids this patch gap problem.

121.   The working paper refers to a 2020 Exodus Intelligence study as a reason to downplay the importance of the patch gap issue. It is wrong to do so. The CMA claims that, according to the study "*when Chrome went from releasing security updates every six weeks to every two weeks or less, n-day exploits on Chrome had become impractical*".[83] It fails to recognize that the reason the increase in update releases has an impact on Chrome is not because of the increase in release frequency itself, but because patches to security issues were developed in the open. As a result, increasing the update frequency reduced the time the fix was visible in public before users received it. This is not a valid basis of comparison with WebKit or Gecko, as both Apple and Mozilla do not publish fixes in open source until an update is released. It is also the case that the study only covers a single category of n-day exploits, and it does not cover scenarios where users do not install updates or cannot receive updates and therefore remain vulnerable. Accordingly, this study offers little probative value.

122.   In contrast, the United States Department of Homeland Security's Cybersecurity and Infrastructure Security Agency (CISA) publishes a catalog of known exploited vulnerabilities which currently lists significantly more known vulnerabilities for Chromium/Blink/Chrome than for WebKit/Safari.[84]

---

[81]   The first two analysis assessed browser applications available in the US, with the third specific to the UK Play Storefront.

[82]   See data provided to the CMA in Apple_Confidential Annex_CMA_MI_0038.

[83]   Working Paper 2, paragraph 5.23.

[84]   See https://www.cisa.gov/known-exploited-vulnerabilities-catalog, showing that WebKit and Safari have 2627 known exploited vulnerabilities as compared to 3795 known exploited vulnerabilities for Chromium/Blink and Chrome.

b) Bounty data

123. Third-party exploit acquisition programs also reflect that mobile exploits via browser engines are in high demand. For example, the Crowdfense research hub and acquisition program for high-quality zero-day exploits and advanced vulnerability research indicates that exploits targeting Chrome and Safari on mobile devices are in high demand, with Safari bugs more expensive likely because they are viewed as being more difficult to develop because of the stricter security protections on iOS:

- The bounty for Chrome exploits granting remote code execution ("RCE") and local privilege escalation ("LPE"), which are among the most severe attacks that can be perpetrated on a device, ranges from USD $2 million to USD $3 million.

- Crowdfense offers bounties for Safari RCE and LPE exploits ranging from USD $2.5 million to USD $3.5 million.

c) The CMA's arguments do not rebut the evidence showing the security benefits of the WebKit requirement

124. The CMA appears to point to Apple's DMA compliance efforts as a basis for its emerging thinking that the WebKit requirement is not required for Apple to achieve high standards of security on the platform. This is wholly misconceived.

125. Apple has clearly explained that its DMA compliance efforts introduce risks that cannot be eliminated,[85] and thereby create a materially less secure experience for EU users. The CMA should not wish to extend such risks to UK users.

126. Finally, the CMA asserts that other vendors could implement better security features in the absence of the WebKit requirement. Apple has addressed these arguments.[86] Even if it were correct that some individual developers could implement even stricter security features, this would not cast doubt on the effectiveness of the WebKit requirement:

- First, those benefits would apply only to the individual browsers and their users, whereas the WebKit requirement allows for a high standard of security for all browsers on iOS and consequently for all users irrespective of the browser they choose.

- Second, this argument ignores the threats that would apply if additional access and flexibility were opened to _all_ browser vendors, including those with very different incentives or capabilities when it comes to security. The risks identified above (in particular the patch gap problem) would be brought to iOS and would undoubtedly result in multiple browsers on iOS shipping browsers based on out-of-date engines with malware and other vulnerabilities as well as with lower adoption rates of new platform security mitigations.

- Third, it ignores the vital importance of differentiation at a platform level. Users who choose iOS devices do so with the knowledge and expectation that it is the most secure platform. This security is based, to a significant degree, on the WebKit requirement. Removing that requirement would remove Apple's ability to differentiate from Android on this basis and thus frustrate customer choice.

---

[85] See https://www.apple.com/newsroom/2024/01/apple-announces-changes-to-ios-safari-and-the-app-store-in-the-european-union/ ("_Inevitably, the new options for developers' EU apps create new risks to Apple users and their devices. Apple can't eliminate those risks ...._").

[86] See above at paragraph 87.

2. The WebKit requirement allows for better privacy outcomes

127. Despite the importance to users of privacy and the key role that privacy plays in Apple's differentiated offering, the CMA's analysis of privacy outcomes in Working Paper 2 is under-developed and relies almost exclusively on assertions that third-party vendors could offer different or additional privacy features. [87] As noted above, Apple has addressed these arguments.[88] Even if it were correct that some individual developers could implement more privacy features, this would not cast doubt on the effectiveness of the WebKit requirement because (a) those benefits would apply only to the individual browsers and their users; (b) it ignores the threats to privacy that would apply if additional functionality were opened to <u>all</u> browser vendors, including those with very different incentives or capabilities when it comes to protecting users' privacy; and (c) it ignores the vital importance of differentiation at a platform level.

128. With respect to privacy, the risks of opening up iOS to competing browser engines that do not implement privacy-enhancing features at the engine level are significant. As the CMA itself recognises, different browsers take different approaches to privacy and its importance. Some browser developers would be incentivized to minimize privacy protections for their own commercial interests. Users could find that their data is being tracked, used, and sold to third-parties without their knowledge because Apple could no longer limit browser features that might harm user privacy, for instance by enabling tracking or monitoring of location data.

3. The WebKit requirement allows for better performance

129. The CMA finds that it has "*not yet seen clear evidence that the WebKit restriction improves performance of browsers on iOS (or iOS devices more generally) overall*".[89] As with security and privacy, the CMA's analysis (however brief) is flawed.

130. There is no supporting evidence cited for the CMA's assertions that third-party browser vendors could improve performance beyond the level enabled by WebKit,[90] other than the vague and inaccurate claims raised in the context of Working Paper 3.[91] Without having identified and tested with Apple the performance features which developers claim to be unable to introduce on iOS, the CMA can have no probative evidence on which to base its conclusions.

131. Indeed, such a conclusion is directly contradicted by the evidence Apple has provided that competing browsers perform better on iOS than on Android devices.[92] It is also inconsistent with the results of the CMA's developer survey, which show that a majority of developers did not consider that WebKit's performance was worse than that of alternative browser engines.[93]

132. The CMA appears to suggest that users have less need for platform design decisions to enhance performance.[94] This misses the point entirely – not only does WebKit provide a high standard of performance across all browsers, but also it prevents individual browsers from implementing features that would downgrade device performance generally (such as battery-draining features). Users would not necessarily associate a reduction in device performance

---

[87] Working Paper 2, paragraph 5.37.
[88] See paragraph 87.
[89] Working Paper 2, paragraph 5.39.
[90] Working Paper 2, paragraph 5.41.
[91] Apple addresses these claims in Section V below.
[92] For a summary, please see paragraph 93.
[93] Working Paper 2, paragraph 4.45.
[94] "*Performance is likely to be a parameter that users are reasonably well placed to evaluate and express a preference on, including by potentially trading it off against others. Users are therefore likely to be better placed to make informed choices on browser performance and have less need for platform level restrictions that ensure a given performance level*" (Working Paper 2, paragraph 5.42).

with an individual browser. Rather than resulting in users switching between competing browsers on iOS, a removal of the WebKit requirement is more likely to result in users being dissatisfied with their iPhones overall and reducing Apple's ability to compete with Android devices.

133.   Apple therefore respectfully requests that the CMA give greater weight in its emerging thinking that it is reasonable to expect the integration between hardware and software could result in performance advantages.[95]

E.   Conclusion

134.   As the above sections show, the WebKit requirement fosters competition between mobile browsers, supports the development of web apps, and leads to positive user outcomes overall on the key parameters of security, privacy, and performance. Apple therefore urges the CMA to rethink its emerging findings in relation to the WebKit requirement.

---

[95]   Working Paper 2, paragraph 5.40.

V.   WORKING PAPER 3: ACCESS TO BROWSER FUNCTIONALITIES WITHIN THE IOS AND ANDROID MOBILE ECOSYSTEMS

135.   The emerging thinking set out in Working Paper 3 suggests that Apple provides to Safari, but not third-party browsers, numerous features and functionality of importance to attracting users[96] and that this may adversely impact third-party browsers' ability to attract users[97]. That is not the case.

136.   As already explained in detail to the CMA, Apple has developed and rolled out over 250,000 APIs for app developers to use in building better apps more effectively and efficiently, and Apple remains incentivized to continue extending additional technology to developers so they continue to build browsers that provide high satisfaction to users. Apple also supports numerous ways for developers to learn more about what APIs are available and how they can be used, ranging from sessions at WWDC designed to guide developers on how to incorporate Apple's APIs into their apps, to publicly-available materials presented on developer.apple.com, and even personalized support, including through Apple's Worldwide Developer Relations team ("WWDR"), whose purpose is to work with and support developers and who can and have directed developer questions to Apple engineering and other internal resources.

137.   The CMA does not give due regard to the significant extent to which browser developers are able to differentiate their offerings from Safari on the basis of the existing (and constantly increasing) levels of technological and other support provided by Apple.[98] This is a critical contextual factor and the CMA's failure to give it sufficient credence undermines the working paper's emerging thinking.

138.   As discussed further below, when identifying and building new features, Apple balances the goal of expanding and providing new feature offerings with its commitment to providing the levels of security, privacy, and performance that users demand and expect on iOS. Additionally, certain features that have particular security and privacy implications can only be provided subject to necessary guardrails. This approach takes time and cannot be rushed.

A.   Many of the concerns raised in Working Paper 3 are new and were not raised previously during the investigation

139.   The working paper includes in its list of 'concerns' functionalities that the CMA has not previously raised with Apple. Apple has had no ability to address these alleged concerns. This is no small issue; there are eight functionalities identified in the working paper in relation to which the CMA has not engaged with Apple. Compounding this, the working paper recites a number of 'concerns' in such redacted form that they provide no information at all about the substance of the issue being raised.[99] As noted above,[100] it is surprising that the CMA has seen fit to publish its emerging thinking on such a range of concerns without first testing these with Apple and providing Apple an opportunity to comment. By presenting one-sided, conclusory statements without providing Apple the opportunity to test and rebut them, the CMA presents a misleading picture of the issues at stake. Unless the CMA addresses this, it risks following an approach that is both procedurally unfair and bound to result in irrational findings.

140.   To the extent possible, given the limitations of time and the CMA's lack of specificity, Apple addresses some of these new 'concerns' in the sections below. Where it has not been possible to address them in the time available, Apple will do so in a separate submission.

---

[96]   Working Paper 3, paragraph 3.64.
[97]   Working Paper 3, paragraph 3.68.
[98]   See paragraph 84 *et seq*.
[99]   See, for example, Working Paper 3, paragraph 3.21.
[100]   See paragraph 38.

B.   **Apple provides parity of access to the majority of features listed in the working paper**

141.   The working paper expends a tremendous amount of space setting out a long list of features about which third parties have complained to the CMA. But the list is wholly misleading. As is clear from Apple's submissions (and indeed from some of the 'complaints' themselves), third parties in fact do have access to the vast majority of these features in a number of ways:

- First, third parties have equal access to many features that are built into WebKit.

- Second, where the feature in question is a Safari feature (rather than a WebKit feature) and there are different ways to implement the feature at the browser level using WebKit, WKWebView, or other available resources or tools, third-party browsers can build their own versions of the feature. Whilst Apple is committed to providing as wide a range of functionality as possible to developers in order to facilitate browser competition on iOS, it is not Apple's role to build for third parties every single feature that developers may wish to have but do not want to invest in creating in-house, nor should it be.[101]

- Third, for some of the features, neither Safari nor third parties have access, which precludes any possibility of Safari being granted a competitive advantage.

142.   The CMA's one-sided list of features about which it has 'concerns' is also out of date. Many of them appear to have been collected several years ago during MEMS. Since that time, as the working paper recognizes, Apple has added or enhanced support for many of the features in question, just as it has more generally supported WebKit and created developer APIs since the launch of the first SDKs for iOS. Indeed, the ongoing level of development in these areas underlines the dynamic and competitive nature of the browser market. This demonstrates Apple's strategy and commitment to expanding the developer APIs it offers in a way that maintains security and privacy protection.

143.   Further, there is no support for the CMA's emerging conclusion that the very limited functionality that was available to Safari before third-party browsers is material or has had an adverse impact on competition.[102] Indeed, the CMA has provided no evidence (whether user survey evidence or otherwise) to suggest that this is the case or to show that third-party browsers have in any way been disadvantaged if Safari obtains access at an earlier stage to certain limited functionalities.

C.   **The few functionalities that are not available to third-party browsers give rise to significant security and/or privacy concerns or have no demand from third-parties**

144.   Contrary to the picture painted in the working paper, there are very few functionalities that Apple has not made available to third parties. Indeed, of the 22 features listed in the working paper which Apple has been able to consider, _only three_ are functionalities that are available to Safari but not to third-party browsers. We address each of those in turn.

145.   <u>Smart App Banners</u>: Apple understands the CMA's reference to "<u>Universal linking</u>" to refer to Smart App Banner functionality in Safari.[103] A universal link in a browser offers a way of linking to content on a specified website or in a designated app—this functionality is available

---

[101] See above at paragraph 62.

[102] See Working Paper 3, paragraph 3.64, stating that Apple's approach "_may be particularly significant [for] newly developed or innovative features, which can be important for attracting users, meaning even a small time advantage for Safari could have an impact_".

[103] https://developer.apple.com/documentation/webkit/promoting_apps_with_smart_app_banners.

to third-party browsers and other apps. [104] Smart app banners, however, would pose significant risks if enabled for third-party browsers. To enable this feature, third-party browsers would be given the opportunity to identify all of the apps downloaded on a user's device, which is data that could be manipulated to track and fingerprint the user, thereby creating significant privacy risks.

146.   JIT compilation: As Apple has explained, JIT compilation presents very significant security considerations because it entails the generation of self-executing code. Apple restricts access to JIT compilation to WebKit because it can ensure that the JIT compiler only emits code corresponding to compiled JavaScript and remains limited to optimization of the runtime of the web platform. Free access to be able to modify or replace the JIT compiler by third parties would place a high security burden on developers to minimize the vulnerabilities that arise from such access and create a significant risk that developers would introduce novel vulnerabilities.

147.   Background upload and download: The working paper indicates that this functionality has not been raised as a concern by third parties. Accordingly, it does not provide a basis for any adverse findings.

148.   The CMA cannot reasonably consider that such a limited number of functionalities forms a proper basis on which to conclude that Apple's approach to functionality access impedes competition. Further, even if the CMA were to consider that access to these functionalities impacts competition, it is both reasonable and proportionate for Apple to restrict access to these functionalities because to do otherwise would present disproportionate security and privacy risks.

D.   Virtually all the features or functionalities identified as "concerns" are available in some form to third-party browsers today or are equally unavailable to Safari

149.   Nineteen of the twenty-two features or functionalities with which the CMA takes issue in the working paper are things that are either (i) WebKit features to which third parties and Safari have equal access; (ii) features that third parties can build using WebKit and/or WKWebView or other resources at third-parties' disposal; or (iii) features to which neither Safari nor third parties have access. The features which fall into each of these categories are listed in Table 2 below.

Table 2: Features that do not raise concern on the evidence

(i) WebKit features to which third parties and Safari have equal access	(ii) Features that third parties can build using WebKit/WKWebView or via other resources	(iii) Features to which neither Safari nor third parties have access
• Add web apps to home screen • Service Workers • WebRTC • UserMedia • Process separation • Authentication of users to a network • Apple Pay	• Password keychains and autofill in browsers • Reader Mode • Context menus • Safe browsing • iCloud Private Relay (an iCloud feature, not Safari feature) • "Hide my email" and iCloud+ (an iCloud	• Tracking of default browser selection • Device management certificates

---

104   https://developer.apple.com/ios/universal-links/

• Previous WebKit versions[105] • Intelligent Tracking Prevention[106]	feature, not Safari feature)[107] • Browser extensions • Contacts banner link from Messages app (third parties can build using third-party messaging apps)	

150. Clearly, the vast majority of the features listed in the working paper are already as available to third-party browsers as they are to Safari, and, in many circumstances, have already been implemented in third-party browsers. There is no basis on which to conclude that Apple's approach restricts competition or could result in an AEC.

C. Apple does not restrict competition in relation to browser extensions

151. The working paper focuses to a large degree on browser extensions. The evidence, however, does not support the CMA's emerging thinking that Apple's approach to browser extensions raises competition concerns.

152. As Apple has explained, while third-party browsers do not utilize Safari Extensions, third-party browsers are free to build and implement web extension functionality on top of WebKit; and indeed they do so. Orion, for example, allows users to install and use browser extensions on iOS.[108]

153. Apple creates guardrails around the use of browser extensions because they present an additional dimension of security, privacy, and reliability risk.[109] A browser extension, which

---

[105] The working paper lists a vague 'concern' that in the past third-party browsers were only able to access a "slower" version of WebKit, with this noted as no longer being the case (Working Paper 3, paragraph 3.25). Irrespective of the fact that this 'concern' is not extant, Apple strongly disagrees with the premise that Apple has ever limited third-party browser developers to a "slower" version of WebKit than the one used by Safari. Legacy versions of WebKit, such as the version deprecated roughly a decade ago, were also relied upon by Safari; when new versions of WebKit, featuring better features and performance, were released, they were made available to Safari and to third-party browser vendors alike. Apple has never created and reserved a "faster" version of WebKit for Safari's use only.

[106] When Apple first developed Intelligent Tracking Protection ("ITP") at a platform level it did not see demand among third parties, including browser developers, for the feature. It also observed that certain third parties, particularly those reliant on advertising, were resistant to the feature at launch. Subsequently, other browsers sought to compete with Safari on this basis, and in response to that demand Apple sought to build the tools and APIs necessary for them to do so in a way that preserved user privacy and security. As Mozilla notes, Apple has made ITP available to third-party browsers since 2020 (Working Paper, paragraph 3.50).

[107] Brave submitted that the "hide my email" and iCloud+ features are not available to third-party browsers even though they would add significant consumer value (Working Paper, paragraph 3.31(b)). This is incorrect. Hide My Email is a feature built in to Sign in with Apple and iCloud+, and is available to apps and websites supporting Sign in with Apple. See https://support.apple.com/en-us/105078. Likewise, iCloud+ is a premium service that provides features like enhanced storage for users that work with their apps, including third-party apps. See https://support.apple.com/guide/icloud/icloud-overview-mmfc854d9604/icloud. More to the point, third-party browsers are free to create equivalent functionalities. Indeed, there are multiple apps currently available on the App Store that offer anonymized email services. Similarly, many services offer server-side content storage and related features, including but not limited to Google and Box.

[108] https://apps.apple.com/in/app/orion-browser-by-kagi/id1484498200

[109] The value of the data accessible through a web extension creates incentives for attackers to deploy malicious extensions. Google's Chrome Web Store, for example, was shown to host multiple malicious extensions that had been downloaded a combined 87 million times

may be provided by an entity other than the browser vendor, introduces another entity that can access a user's sensitive information while browsing. As a result, the user must also trust the browser extension vendor with their data. This presents both a security and privacy risk depending on the implementation of the extension model. Extensions may seek to modify a webpage, access cookies, or modify tabs in the browser itself. Many extensions request access to every site that a user visits, and many require that the user grant the extension access to all websites in order to use the extension at all within the browser.[110] Other extensions may seek device permissions to access the display screen, storage, or a feature on the device.

154.   In addition, native browser app extensions could be implemented to allow untrusted executable code from the web. This would open a significant vector for malicious actors to attack the iOS platform.

155.   In light of these concerns, Apple has taken a cautious approach to supporting the use of browser extensions on iOS. It has balanced developer requests for web extensions with the potential risks of such functionality, by implementing appropriate protections before exposing the access needed. Such an approach is entirely proportionate and reasonable. It has allowed Apple to allow third-party browsers to ship extensions, and thus differentiate to the extent they wish to use extensions to do so, without risking user privacy and security. Accordingly, the protections and affordances that Apple has put into place cannot and do not impede browser competition, nor has the CMA pointed to any evidence suggesting such an impact on competition.

E.   The CMA's concerns with respect to documentation are unwarranted

156.   The working paper finds fault with the amount and quality of the documentation Apple makes available to developers.[111] These concerns are wholly unfounded.

157.   Apple offers multiple mechanisms for developers, and users, to learn about available features. This includes documentation on developer.apple.com and during WWDC, as well as direct communications with Apple's WWDR team, engineers, and workshop personnel. Apple has provided more than 170,000 technical documents and sample code,[112] and Apple strives to increase the volume and quality of its documentation over time. As demonstrated to the CMA, Apple has received recent feedback calling out and praising its approach to documentation.[113]

158.   The working paper lists only vague assertions by Opera, Mozilla and one other unidentified browser vendor that Apple's documentation is 'incomplete'. Such assertions provide no basis upon which the CMA can draw conclusions regarding either (i) the validity

---

(https://usa.kaspersky.com/blog/dangerous-chrome-extensions-87-million/28561/). Moreover, the particular access needs and functionality of a browser extension may not be fully understood by a user, as many ship with a browser and therefore the user may not fully appreciate what additional functionality will come with that browser.

[110] As an example of the privacy risks this presents, a user might therefore need to give a web extension access to all of the information input into their banking website, amazon.com, and a health form that the user's doctor requires before a health check-up.

[111] Paragraph 3.65 states that "*there are credible concerns that the visibility and documentation of APIs that can be accessed by third-party browsers on iOS by Apple is poor. This may increase the cost or difficulty of implementing a feature to third-party browsers or result in third-party browsers not being aware that a given functionality is available*".

[112] See https://www.apple.com/app-store/developing-for-the-app-store/.

[113] See for example, '*I'm so glad the Safari team at Apple is now doing a great job documenting the new APIs with each version.*' https://x.com/firt/status/1800271385643913292. Apple has also received a significant amount of more general positive feedback on the WWDC announcements. Examples of such feedback include '*Safari did it again! Look at all of those updates for CSS!' and 'View Transitions + more is coming to Safari*

of any concern or (ii) whether, if valid, such a concern impacts competition. Nor does it provide any support for the CMA's conclusion that this could increase the cost or difficulty of implementing a feature in third-party browsers or result in third-party browsers not being aware that a given functionality is available.

F.   Conclusion

159.   Apple provides widespread access to the features and functionalities that developers seek in order to build and improve their mobile browsers. It does so in a reasonable and effective way, balancing the needs of developers with those of users as well as the integrity of the platform. Apple works continuously to expand functionalities to third-party browsers in a manner that preserves security and privacy. Apple therefore urges the CMA to rethink its emerging findings in relation to access to functionalities on iOS.

VI. WORKING PAPER 4: IN-APP BROWSING WITHIN THE IOS AND ANDROID ECOSYSTEMS

160.   Apple has invested heavily to provide developers with different methods of accessing the Internet within their apps, which developers may tailor to the app experience they desire. Apple's offerings provide developers with choice, convenience, and control over the experience that they want to give users in their apps, while continuing to offer critical protections against the risks and vulnerabilities inherent in accessing web content.

A.   The in-app browsing use case

161.   As Working Paper 4 recognizes, in-app browsing refers to the scenario where a developer seeks to present web content within their native app, as opposed to directing a user to a separate, dedicated browser app. Typically, the use case for in-app browsing is fairly narrow and specific, where developers utilize in-app browsing to access web content that complements the native app experience they seek to provide.   For example, a social networking app may want to allow users to open links to web content — for example to shop — from within an advertisement inside their app, without having to redirect the user to their outside, default browser app. Others may rely on an in-app browsing solution for a very limited purpose, such as for displaying a privacy policy or for completing a sign-in flow.

162.   The protections afforded by the WebKit requirement are vital for in-app browsing.[114] Notably, there are over 20,000 apps that call on SFSafariViewController, which is far larger than the over 100 browser apps on the UK App Store, and represents a very significant potential attack surface that is defended via WebKit. Further, for developers who use in-app browsing, access to the web is more likely to be tangential to the purpose of their apps and they may be less equipped, less aware of the need, or less incentivized to provide privacy and security protections for user interaction with web content than a dedicated browser app developer would be.

B.   App developers are the relevant 'consumers' of in-app browsing functionality

163.   The working paper addresses how different stakeholders interact with in-app browsing, including browser vendors, app developers, web developers, and users. From the perspective of assessing how competition works in relation to in-app browsing, however, the key stakeholder group is app developers. These are the 'consumers' of in-app browsing functionality.

164.   The CMA's focus on end-users, in particular on "*the extent to which users of apps are able to make effective choices about IAB implementations on iOS devices*"[115] is wholly misplaced. It is app developers that decide whether and how they want to incorporate web content in their apps and, having made that determination, app developers that choose their preferred mechanism for accessing that web content during an in-app browsing experience. Indeed, most end-users are likely unaware when they are engaged and in-app "browsing," and this is largely by design.

165.   Many apps incorporating in-app browsing do not offer users a choice of browser for the in-app browsing experience because they want to make it a seamless experience where web content enriches their native app experience rather than detracting from or removing the user from that overall native app experience. This is exemplified by the evidence provided to

---

[114] As discussed above in Section IV.
[115] Working Paper 4, paragraph 4.33.

the CMA by TikTok, which states that the decision to use WKWebView on iOS comes from its willingness to create a "*more seamless experience for customers*".[116]

166.   App developers, not users, thus choose the level of choice that a user may have for their in-app browsing experience and, consequently, are the relevant consumer group for purposes of assessing whether in-app browsing competition is well-functioning.

C.   App developers are satisfied with Apple's in-app browsing functionalities

167.   App developers who want to use in-app browsing are by no means a homogenous group – they cover all app categories, fall on a spectrum from large to small, and have different areas of focus and expertise. Examples of apps that incorporate in-app browsing include: TikTok, BBC iPlayer, Etsy*, PayByPhone Parking*, Boots, Ryanair, Ticketmaster UK*, NYT Games: Word Games & Sudoko*, and the NHS App (those with an * incorporate SFSafariViewController only).

168.   With such a divergent group of consumers, developers offering in-app browsing functionalities must be capable of meeting diverse needs in terms of ability and willingness to develop code, requirements for security and privacy protections, and customization of the in-app browsing experience. Apple's offerings meet these divergent developer needs.

- <u>WKWebView</u>: is designed to let developers create their own browsing experience and can be used both for dedicated browser apps and apps that incorporate in-app browsing. WKWebView allows developers to embed web content into an app, and also offers developers a highly customizable way to enable internet browsing or display web content in their apps. WKWebView gives developers greater access to and control over the browsing experience of users, and therefore carries with it the requirement for developers to use the security and privacy mitigations available through WebKit.

- <u>SFSafariViewController</u>: For developers who do not want to build their own browsing view and instead prefer an "out-of-the box," or "plug-and-play" option for presenting web content in their apps, Apple created SFSafariViewController. Contrary to the CMA's understanding, SFSafariViewController is not a remote tab implementation. And despite its name, it does not invoke or involve Safari but rather is a view controller built on WebKit. SFSafariViewController does, however, build in important privacy and security benefits. For example, if a user navigates to a page during an in-app browsing experience, click-through advertising would be measured in a privacy-preserving way with Private Click Management.

- <u>Default Browser</u>: Apple also supports developers who choose to take users out of the app and direct them instead to a dedicated browser app, regardless of whether that browser is Safari or a third-party browser. Indeed, Apple does not prevent third parties from pointing users to an alternative browser such as Chrome or Mozilla.

- <u>Custom SDK</u>: Finally, Apple also supports browser developers who seek to offer a SDK wrapper around WKWebView for app developers who want to incorporate that particular browser's interface, together with any of the browser features that rely on WebKit, into their apps.

169.   The app developer feedback reflected in Working Paper 4 indicates that app developers understand the differences between Apple's in-app browsing functionalities and choose which to deploy depending on their needs:

---

[116] Working Paper 4, paragraph 4.42.

- One developer "*prefers to use a webview IAB on open links that are not promoted content because it can personalise the user interface more than it can with SFSafariViewController. For advertising links, however, this app developer uses SFSafariViewController.*"[117]

- A different app developer apparently chooses WKWebView over SFSafariViewController because "*SFSafariViewController does not provide any information to the host app about the in-app browsing session*".[118]

- One developer apparently prefers not to use in-app browsing at all, because "*better outcomes can be achieved from ads that send users to remote tab IABs or out of the app to a dedicated browser*".[119]

- Another app developer apparently indicated that it uses both SFSafariViewController and WKWebView "*because the app developer's advertising business customers prefer SFSafariViewController for opening ad links, but the app developer has more customisability over its WKWebView IAB, which is used for all other in-app weblinks.*"[120]

170. The working paper identifies very little concern on the part of app developers with respect to SFSafariViewController,[121] with customization being the only concern relating to in-app browsing itself. To the extent that app developers want the ability to customize, as set out above, WKWebView allows them to do so and many developers are aware of this and indeed use it for that reason. Indeed, as the CMA observes *"[t]he evidence available to date suggests that [Apple may restrict the customisability and functionality of IABs] … is a relatively minor concern for app developers.*"[122] We therefore agree with the CMA that it should de-prioritise this aspect of the market investigation.

171. The working paper also finds that "*the majority of app developers would not be interested in developing a bundled engine IAB on iOS. This includes large app developers that are relatively engaged in developing their IAB. These app developers are largely satisfied with the functionality of WKWebView*".[123]

172. The evidence clearly shows, therefore, that app developers (the relevant 'consumers' for purposes of in-app browsing) are broadly satisfied with Apple's in-app browsing functionalities and do not have concerns that would either amount to a significant competition issue or require remedy.

D. Apple's approach strikes an appropriate balance between the needs of developers and users

173. Apple's approach to in-app browsing places app developers in charge of the in-app experience that they want to offer users in their apps. Developers can choose how to present web content in their apps, decide what level of customization they want, select the user interface to display, and even decide whether they want to display a toolbar – all within a framework that provides high baseline levels of privacy, security, and reliability protections. Moreover, SFSafariViewController and WKWebView protect against risks arising from "patch

---

[117] Working Paper 4, paragraph 2.50(b).
[118] Working Paper 4, paragraph 2.50(c).
[119] Working Paper 4, paragraph 2.50(a).
[120] Working Paper 4, paragraph 3.8(b).
[121] Even OWA in its paper https://open-web-advocacy.org/files/OWA%20-%20DMA%20Interventions%20-%20In-App%20Browsers%20v1.2.pdf states that "*In many aspects, SFSafariViewController is vastly superior to all webview and bundled engine in-app browsers. It does not share all of their myriad of problems*" (page 38).
[122] Working Paper 4, paragraph 4.2.
[123] Working Paper 4, paragraph 4.29.

gaps" because they benefit from Apple's uniform approach to security updates on iOS. And, as noted below, developers also have the option of choosing to link out to a default browser via an easy-to-locate icon.

174.   By contrast, if control over the in-app browsing experience were wrested from developers, they would lose control over an important aspect of their in-app experience, and also have to code for the 'lowest common denominator' browser functionality because the in-app experience would be wholly dependent on user settings. The CMA's own research reflects that users have little appetite or interest in specifying the in-app experience.[124]

175.   This is unsurprising given that, as the CMA survey also finds, users prefer to stay within an app for browsing content in which they have only a surface level interest, as this allows them to return to their original task more easily and not open an unnecessary number of browser windows.[125] Perhaps the lone exception, also reflected in the CMA's survey, occurs when users interested in purchasing an item prefer to be taken to a browser outside the app. [126]   Here again, Apple's approach to in-app browsing serves that very need; SFSafariViewController provides users with the option to punch out to their default browser via an easy-to-find icon in the bottom right-hand corner of the SFSafariViewController UI. In this way, users maintain control over whether to continue using SFSafariViewController or to view the content within their default browser (whatever that browser may be).

176.   With such limited interest in the mechanics of in-app browsing on the part of users, it would not make sense for Apple to disadvantage developers' ability to create the in-app browsing experience that they seek to build. Apple has struck an appropriate balance between honoring the user choice of default browser and the developer choice for in-app experience. And underlying it all is WebKit, offering developers the benefit of — and meeting user needs for — high baseline levels of privacy, security, and performance.

E.   Browser vendor concerns are significantly overstated

177.   Whilst app developers are broadly satisfied with Apple's in-app browsing implementations, and users are generally unconcerned by, or even unaware of, in-app browsing, the majority of 'concerns' raised in the working paper come from mobile browser vendors. These concerns, and their impact, are significantly overstated.

178.   The working paper assumes that Safari is the only browser used on iOS for in-app browsing, based on an assumption that SFSafariViewController calls on Safari. That is mistaken. As noted above, SFSafariViewController is a view controller built with WebKit and does not call upon or use Safari.

179.   It is therefore factually incorrect to claim that Apple preferences Safari or otherwise requires developers to use Safari instead of third-party browsers. Third-party browser developers have several paths to compete effectively to supply in-app browsers on iOS. As Apple has explained, a browser vendor could release a SDK wrapper around WKWebView for app developers to incorporate the interface of that browser, together with any of the browser features that rely on WebKit, into their apps.[127] App developers could use such an SDK, which would in effect be their equivalent version of SFSafariViewController, to create a more "consistent" experience for those users who have selected that browser as their default on iOS.

---

[124] As the CMA's research shows, "*overall users have very low levels of awareness of in-app browsing. Respondents had not thought about in-app browsing before or whether they were using a browser. Respondents also did not normally think about in-app browsing or what was happening operationally 'behind the scenes' when they viewed web content within an app.*" (paragraph 4.49)
[125] Working Paper 4, paragraph 4.44
[126] Working Paper 4, paragraph 4.49
[127] This is acknowledged by the CMA at Working Paper 4, paragraph 4.5.

180. The CMA confirms that browser vendors are aware of this possibility; it observes that "*some browser vendors have considered offering alternative webviews to app developers*".[128] The CMA also notes, however, that browser vendors generally do not do so for reasons unrelated to Apple.[129]

181. Browser vendors' professed concern that remote tabs can make a browser more "sticky" and that SFSafariViewController limits this option falls similarly flat because of the inherent nature of in-app browsing and the app developer's control over the in-app browsing experience. Any idea that users would be influenced in their browser choice by the browser used in a remote tab in-app browser assumes that users know (and care about) the underlying technology operating the in-app browser. However, users generally have little to no knowledge of or interest in what browser vendor is providing the in-app browsing experience, as the CMA's own evidence confirms.[130]

182. Many of the remaining concerns appear to center around the requirement that WebKit is used as the underlying browser engine for in-app browsing. These are unwarranted for a number of reasons:

- First, the WebKit requirement – both in relation to dedicated browser apps and in-app browsing is necessary, beneficial and pro-competitive. It reflects and enables Apple's prioritization of the security and privacy of its users, including while engaging in in-app browsing. In-app browsing carries the same material privacy and security risks as accessing web content through dedicated browsers and therefore requires the same protections.

- Second, as the working paper itself notes, "*remote tab IABs may exhibit certain security and privacy vulnerabilities that are generally not present in dedicated browsers*".[131] Some browser vendors may tout their intentions to provide more privacy-focused offerings; but even if that were true, clearly not every browser vendor would do so.

183. Finally, bald assertions that Apple's approach is "*bad for competition and there is no inherent security challenge with building a system for remote tab IAB on iOS similar to Custom Tabs on Android*"[132] lack any meaningful evidentiary basis, are of no probative value, and are not amenable to response.

184. The various concerns expressed in the working paper therefore do not demonstrate that Apple's in-app browsing implementations are "*likely to be limiting rival browsers' ability to compete against Safari on iOS because they lack the functionality of displaying web content within an app*".[133] To the extent that a browser vendor wants to offer developers the ability to allow users to access web content through their browsers on iOS, they have been and continue to be able to do so.

185. The evidence presented by the CMA demonstrates extremely limited interaction between the in-app browsing and dedicated browser markets, such that in any event there is likely to be limited to no network effects arising from a browser vendor's presence in both of those markets. There is no basis to find that third-party browser vendors are disadvantaged in the

---

[128] Working Paper 4, paragraph 2.33.

[129] For example, the CMA notes that "browser vendors generally do not interact with webview or bundled engine IABs because these implementations are controlled by the app developer and do not rely on or link to a dedicated browser" (paragraph 2.33).

[130] See paragraph 175 above. This is true across both iOS and Android (in other words, the availability of remote tab in-app browsers on Android has no impact on user awareness or engagement with in-app browsers).

[131] Working Paper 4, paragraph 2.9.

[132] Working Paper 4, paragraph 4.13.

[133] Working Paper 4, paragraph 4.14.

wider browsing market, nor to find that they have suffered from lost monetization as a result of developer use of SFSafariViewController or WKWebView.

F.   Conclusion

186.   As the above sections show, Apple provides effective support for browser vendors who wish to supply an in-app browsing implementation and app developers who wish to implement an in-app browsing experience. Apple's implementations provide developers with choice and control over the experience that they want to give users in their apps, while continuing to offer to users critical protections against the risks and vulnerabilities inherent in any access of web content. The evidence cited in Working Paper 4 demonstrates that app developers, the real 'consumers' of these implementations, have little to no concerns in this context. App developers have a variety of options to create the in-app browsing experience they want to provide for the users of their apps, and Apple has provided a way for users to use their default browser in the one circumstance where the CMA's survey evidence suggests they might. Apple therefore urges the CMA to rethink its emerging findings in relation to in-app browsing on iOS.

VII. WORKING PAPER 5: THE ROLE OF CHOICE ARCHITECTURE ON COMPETITION IN THE SUPPLY OF MOBILE BROWSERS

187. Apple's choice architecture [134] is pro-consumer and pro-competitive. Rather than impeding competition, Apple's UI design and policies actively support third-party browsers and promote user choice.

A. Apple's choice architecture is pro-competitive and pro-user

188. Apple has explained in previous submissions to the CMA that its overarching design objective for iPhone and iPad is to deliver a premium consumer experience and ease of use out of the box. The CMA's analysis is characterized by a failure to take account of these benefits, leading to a one-sided and unbalanced position that ignores its own key survey findings.

> *1.   Pre-installation provides necessary out-of-the-box functionality and cannot reasonably be achieved with a different approach*

189. The CMA relies on evidence from its user survey as a basis for the speculative proposition that, as a result of the pre-installation of Safari, users "*may be less aware of alternative browsers and less likely to make active choices between browsers*".[135] The CMA's conclusion entirely overlooks the reality that such a result is also consistent with positive user outcomes, such as user preferences in favour of the pre-installed browser (or indifference between browsers).

190. It is also clear from the CMA's survey that users are aware of their ability to switch browsers, with 60% of iOS users saying they are "definitely" able to download alternative browsers, and 27% saying they "probably" can do this without help. Only 9% said they "probably" cannot and only 2% said they "definitely" could not download alternative browsers.

191. Aside from the survey evidence, Apple has explained that its approach to pre-installation is pro-user, consistent with its overarching incentives as a device manufacturer: it chooses to integrate apps that it believes users would expect to have available when turning on a new smartphone or tablet, that will enhance the overall consumer experience, and that will differentiate Apple products from its competitors. Reflecting iPhone's marketing at launch as a breakthrough internet communications device, Safari has formed part of the set of integrated apps shipped with each iPhone since 2007. This means that users have long-standing expectations that they will be able to browse the web seamlessly as part of the out-of-the box experience. Moreover, integrating Safari into Apple devices protects against the risk that third parties elect to stop making their apps available on iOS. The findings of the CMA survey, which indicates that 46% of Safari users prefer or chose to keep Safari, and 41% saw no reason to change, support the view that Apple's decision to pre-install Safari leads to user benefits in practice.

192. Moreover, pre-installation of third-party browsers would carry a number of harmful impacts and risks for users, such as:

---

[134] Apple uses the term "Choice Architecture", as this is the term used in the CMA's working papers. The issues under consideration relate more broadly to Apple's design of the user interface on Apple devices, including but not limited to how Apple's design decisions offer users choice and promote ease-of-use.

[135] See Working Paper 5, paragraph 5.5(a). The CMA cites in particular to the finding that Safari was used by 72% of respondent iPhone owners (Working Paper 5, paragraph 3.16). The survey data, in fact supports this more benign interpretation: 46% of Safari users prefer or chose to keep Safari, and 41% saw no reason to change, meaning that 89% of users display a degree of preference for or satisfaction with Safari. Only 6% didn't know there were other options and 5% didn't think about it. Far from remaining with Safari due to a lack of awareness, users are actively choosing to do so out of preference.

- Delays in shipment of devices, compromised storage capacity, performance, and longevity of the device: pre-installing apps could lead to significant security risks, as Apple cannot guarantee that each browser does not have security vulnerabilities before shipping, or relevant updates, without logistically complicating Apple's manufacturing process.

- A worse user experience out-of-the-box: pre-installing numerous browser apps and requiring the user to select one while setting up their iPhone would create a jarring and confusing user experience. With over a hundred browser apps available on the UK App Store, pre-installing them all is simply not feasible. And determining the set of third-party browser apps to pre-install would raise difficult questions about the criteria and process used to determine which to include and which to exclude.

193. Apple's pre-installation of Safari therefore brings about significant user benefits in terms of ease-of-use and, as the survey evidence shows, does not cause users to believe that Safari is the only browser available to them or that they cannot switch.

2. Placement on home screen provides ease of use and is entirely customizable by users

194. Apple's placement of apps is intended to provide the best user experience. Apple seeks to deliver that experience by placing apps on the first screen (including the 'dock' at the bottom of the first screen) that users are most likely to find helpful and by ordering apps in a way that benefits discoverability and the aesthetics of the device. Apple's placement of Safari in the dock on the first screen reflects the importance of browsing to the iPhone experience[136] and the need to provide users with an easy and convenient way to access such functionality. For the majority of users who prefer Safari, it is also convenient to have Safari present on the homescreen.

195. Apple has made the home screen fully customizable and has made it extremely easy for users to tailor the home screen, reflecting well-established demand for such flexibility. Apple has explained in this regard that users have the ability: (i) to reposition Safari from its initial position on the home screen; (ii) to move, and group Safari with other apps; and (iii) to place another app in the original location of Safari. It has also clarified that there are no restrictions on the types of Apps that can replace Safari's position on the device home screen. Apple considers that, in light of the ease and extent of the ability to customize, it is most likely that users who have not moved their browser have not done so because their preferred browser was already on the home screen.

196. The CMA fails to recognise any nuance in making high level assertions that placement of Safari on the home screen contributes to users' low awareness of and propensity to make active choices about browsers.[137] In fact, the majority of users display some degree of preference for the use of Safari so it is extremely convenient for Safari to be placed in the dock on the home screen from initial device set up. The CMA overlooks this equally likely alternative explanation for why some users have not moved their browser (i.e. users see no

---

[136] According to https://www.statista.com/statistics/1285168/uk-daily-mobile-activities-by-usage/, browsing was one of the main mobile activities by UK users as of August 2023.

[137] The CMA's analysis of the home screen highlights the findings of the quantitative user survey that '*36% of respondents chose the location of the browser app, while 48% had not changed the position of the app from when it was first installed on their phone*' and that '*for the majority of iOS users the browser app they most frequently used was located on their home screen (60%), with only 6% indicating that it was in a location other than their home screen.*' (paragraphs 3.27 and 3.28). The fact that 24% of iOS users mostly used Google Chrome (Mobile Browsers Quantitative Consumer Research, Verian, p. 40) suggests that users frequently move their preferred browser to a prominent location (home screen or dock).

reason to move the browser because their preferred browser was already conveniently placed on the home screen).

197.   The CMA analysis also relies on the misunderstanding that, after a third-party browser app is downloaded, "*it is placed on the minus one or minus two screen rather than the default home screen*".[138] When users download new browsers (or indeed any new app) from the UK App Store, provided there is availability on their home screen and depending on their device's Settings,[139] the icon for the downloaded app can appear in the next available position on the Home Screen, as shown in Figure 3 below.

Figure 3: Placement of a new browser app on the home screen
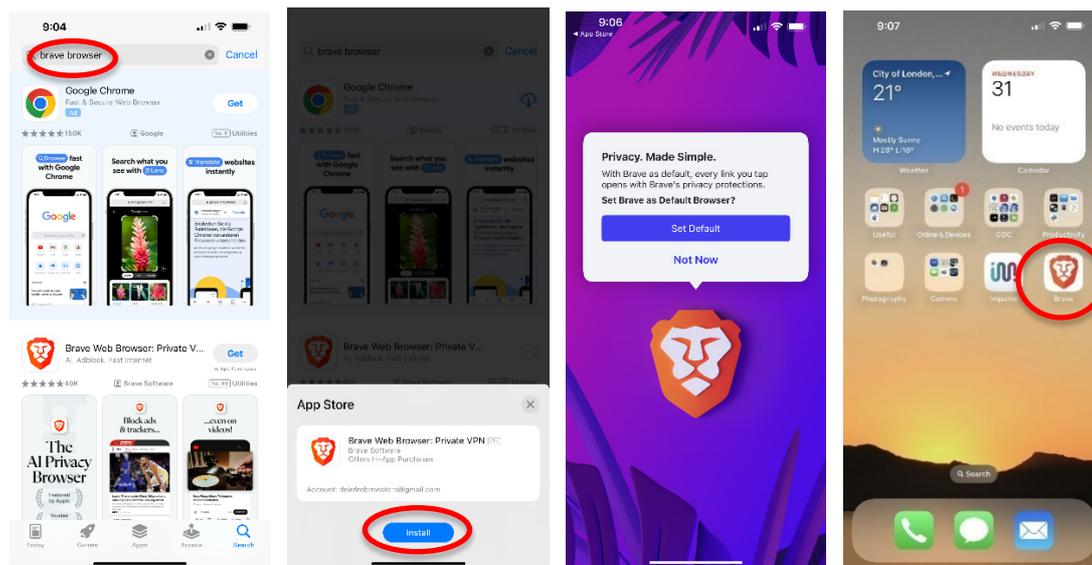
> 3.   *Default settings and the ability to change the default do not favour Safari or make it difficult for users to switch*

198.   The CMA's analysis of the user journey for switching the default browser focusses on: (i) evidence from its user survey which shows that 81% of iOS users have Safari as default and 76% of iOS users have not changed default browser; (ii) the fact that there is no central point available in the settings menu to select the default browser; and (iii) that users cannot search in the settings menu to find how to change default browser.

199.   The CMA's analysis is in clear tension with various findings which together provide strong evidence that the user journey on iOS is easy and is not impacted by 'frictions'. This includes findings that:

• switchers on iOS report changing the default browser was very/fairly easy (88%);

• when iOS users who did not find it "very easy" to switch were explicitly asked for issues experienced in switching default browsers, 48% identified none;

• among iOS users in the survey that had not changed their default browser, the most common reason (34%) was that their current default browser was their preferred browser. For 24%, the web browser used is simply not considered important;

---

[138]   Working Paper 5, paragraph 3.4.
[139]   At Settings > Home Screen & App Library, users have the option to have newly downloaded apps added to the Home Screen, or instead added exclusively to their App Library.

- iOS users on average could spontaneously name 2.5 browsers and 69% of respondents could name more than one browser, suggesting that most users are aware that alternatives exist; and

- 53% of iOS users have more than one browser installed, showing that iOS users frequently download and use alternative browsers.

200.   Furthermore, The CMA's analysis is undermined by significant flaws in its survey design, in its approach to interpreting the survey evidence, and in its consideration of the facts relating to the user journey.

201.   First, the CMA downplays evidence showing that 78% of iOS users state they would be confident in changing the default browser by arguing that this figure may be skewed by certain psychological biases which lead users to be over-confident or overly positive about recent experiences. The CMA has gathered little evidence to support this, instead referring only to limited qualitative research in an unrealistic setting.[140] The CMA's position is thus no more than speculation and cannot serve to undermine the explicit findings of its survey.

202.   Second, and as noted above, the fact that a low share of users have changed the default browser is equally likely to be explained by high satisfaction with the default browser.

203.   Third, the CMA's analysis of the switching journey fails to recognize that it is most likely that the user journey to change the default browser would involve users downloading an alternative browser, trying it out and then deciding they want to use it as a default browser. The natural starting point for the user journey is thus the alternative browser app, with the app able to provide a shortcut to the relevant settings page for the user to change the default. The CMA's focus on an alternative and less likely user journey starting from the general iOS settings is unrealistic.[141] The more likely journey would involve only three steps, as set out in Figure 4 below:

Figure 4: User journey to switch defaults from a prompt

---
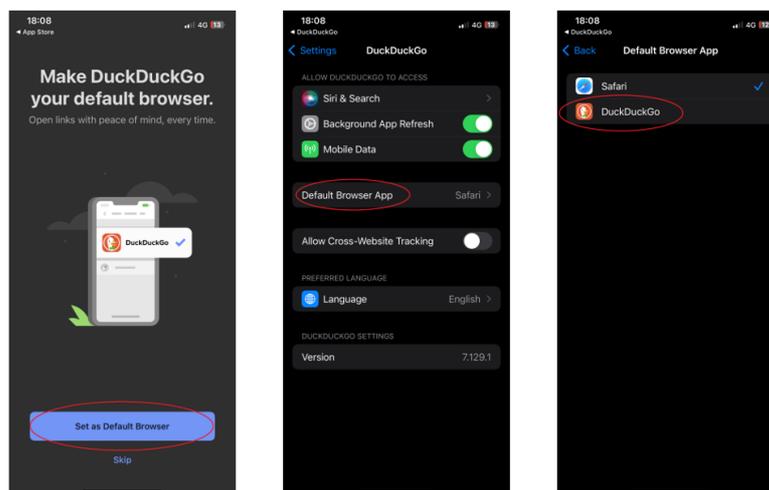
[140]   Respondents were asked to "show […] how [they] would download a new browser and then set it as the default browser on [their] phone". This framing discourages users to search for solutions online, which might be a more conventional approach. ("Mobile Browsers Consumer Research 2024: Technical note on Public Voice panel survey 26, conducted as part of the web browsers market investigation for the Competition and Markets Authority (CMA)", p. 64).

[141]   In any event, even on the CMA's approach, the user journey is intuitive and simple, with users only needing to navigate to Device Settings, scroll down to the browser app, tap the app and then the Default Browser App buttons, before selecting a web browser to set as the default. In light of this, it is clear that the concerns raised by browser vendors about the 'complexity' of this process (paragraph 3.46) are not credible.

204.   Further, once a user has changed the default browser, Apple seeks to preserve that choice even when users make changes to their device, including but not limited to device reboots, iOS updates, updates to Safari or other apps, and device transfers. Any prior behaviors that failed to preserve a user's choice of default browser in such scenarios reflected bugs that have been resolved.[142]

> ### 4.   *Prompts and push notifications are not restricted for third-party browsers even though Safari itself does not use them*

205.   The CMA acknowledges that users may see prompts and push notifications displayed on their device encouraging them to change their default browser.[143] However, it remains concerned that Apple "*does not provide third-party browser vendors with an API that would enable them to target users more effectively when using prompts to switch to alternative browsers (eg target users at the right time to avoid intrusiveness of repeated prompts)*".[144] This concern is unwarranted.

206.   First, Apple's approach in not using prompts or notifications to promote Safari is user-friendly in that it facilitates users in making a choice when they wish to do so but does not interfere with their browsing experience in order to force a choice to be made at an inopportune or inconvenient time. As noted, most browsers prompt users to switch defaults when the user has downloaded and is engaging with that browser. This makes sense from both the browser and user perspective and does not interfere with the user's browsing experience. Allowing browsers to track when they are set as the default with a view to enabling them to prompt users more frequently to change browsers would invariably result in a far worse user experience. For example, if a user had set Chrome as the default browser and then switched to Opera to try it out, an API of the kind sought here would allow Chrome to immediately prompt the user to switch back, interrupting the experience of trying out a new browser, with consequent annoyance for the user and possibly disadvantaging Opera.[145]

207.   Second, as Apple has explained, neither Safari nor third-party browsers are able to detect the default browser setting of specific users, consistent with Apple's broader approach at the platform level to ensure that user privacy is protected. In this way, Safari is treated in exactly the same way as third-party browsers.

> ### 5.   *Uninstallation is not necessary for users to make effective choices or switch browsers*

208.   The CMA asserts that not allowing iOS users to uninstall Safari may restrict users from exercising effective choice as it forces them to keep Safari on their devices, even if they never intend to use it.[146] According to the CMA, the inability to uninstall Safari may lead to an 'endowment' effect. Users may also believe that there may be a functional reason as to why they cannot uninstall Safari, that might give the impression that the pre-installed browser is

---

[142]   The CMA analysis also appears to rely on an OWA report concerning an alleged 'dark pattern' involving the use of different UIs in order to preference Safari is set as the default browser (paragraph 3.48) by not displaying the default browser setting in the Settings app's Safari tab where the default browser is Safari. This is not correct. The default browser app setting in the Safari tab is clearly visible when the user has set Safari as the default.

[143]   Working Paper 5, paragraph 3.54.

[144]   Working Paper 5, paragraph 3.55.

[145]   This type of user experience occurs in desktop browsing and has been widely criticised. See for example: https://www.theverge.com/23935029/microsoft-edge-forced-windows-10-google-chrome-fight - "*I thought I had malware on my main Windows 11 machine this weekend. There I was minding my own business in Chrome before tabbing back to a game and wham a pop-up appeared asking me to switch my default search engine to Microsoft Bing in Chrome. Stunningly, Microsoft now thinks it's ok to shove a pop-up in my face above my apps and games just because I dare to use Chrome instead of Microsoft Edge.*"

[146]   Working Paper 5, paragraphs 3.69 and 3.70.

the endorsed browser by Apple and therefore should be used. These assertions are entirely speculative. Neither third parties nor users have suggested that any of these are likely to occur, nor has the CMA provided any evidence in support of these speculations. The concerns are purely theoretical in nature and of no probative value to a proper assessment of browser competition on iOS.

E.   The CMA's analysis is flawed and divorced from its own findings

209.   The CMA focusses its overall analysis of Apple's choice architecture on two 'hypotheses': (i) that Apple's control of choice architecture on iOS "*may increase barriers to entry and expansion for other browser vendors and further reinforce Safari's very strong position on iOS*"; and (ii) that use of such practices on iOS "*may maintain low levels of consumer awareness and engagement in relation to choice of browsers and reinforce Safari's very strong position on iOS.*"

210.   As set out in Section II above, the CMA fails to test the above hypotheses by reference to an appropriate counterfactual, and thereby fails to ensure that its assessment is grounded in the realities of the market. The issue of the appropriate counterfactual is particularly salient in this context because it is unclear whether any realistic alternative exists for much of the 'choice architecture' considered by the CMA.

211.   The CMA's assessment is also unbalanced, as it fails to take account of various beneficial impacts of choice architecture which it has itself identified. The CMA notes upfront that choice architecture "*can be used to design environments that optimise user experience and help consumers make decisions that are in their best interes*t".[147] The CMA also recognizes the benefits of particular aspects of choice architecture, for example recognizing that pre-installation has benefits to users in "*minimising effort because users do not have to make an active choice at the device set up stage and instead have the option to use their device and additional functionality out-of-the-box*".[148] However, the CMA's analysis effectively disregards these points, instead treating Apple's choice architecture practices as if they were presumptively harmful by asserting that they "*mean that consumers make less active and effective choices about which browser to use on their mobile device*", without any supporting analysis.[149]

212.   In taking this approach, the CMA fails to consider relevant evidence derived from its own survey, as well as Apple's submissions on the various benefits to users and competition which result from such practices,[150] and on its wider incentives to attract both users and developers to its platform. As a result of the above failures, the CMA's analysis is artificial, one-sided, and untethered from the market realities.

213.   One empirical finding from the CMA's user survey which features prominently in the working paper is that the survey "*demonstrates a low level of consumer awareness of the choices that are available to them or the choice architecture that has been applied relating to the use of mobile browsers*".[151] The CMA notes in particular that only 7% of respondents in the user survey indicated that availability of a particular browser was an important factor in their choice of phone.[152] The CMA appears to link this with Apple's choice architecture as a matter of presumption, suggesting the CMA is predisposed to reach findings consistent with its hypotheses set out above, contrary to an evidence-led process.

214.   Having failed to identify any counterfactual to frame its analysis, the CMA has no basis to presume either that Apple's choice architecture lowers user awareness of or engagement with browsers on iOS, or that there is any causal link between observed low levels of user

---

[147]   Working Paper 5, paragraph 2.4.
[148]   Working Paper 5, paragraph 2.8.
[149]   Working Paper 5, paragraph 3.5.
[150]   See further sub-section C below.
[151]   Working Paper 5, paragraph 5.6.
[152]   Working Paper 5, paragraph 2.40.

awareness and/or engagement and Apple's choice architecture. An equally plausible and benign reason for those observations is that browsers are working broadly as intended in providing users with seamless access to the web. In such circumstances, and given they are generally 'low salience' items, one would not expect to see high levels of user awareness of browsers and indeed survey evidence shows that in instances where browsers become salient this is typically because of performance issues. The benign explanation would also be entirely consistent with Apple's submissions on the intended purposes of its choice architecture, its incentives as platform operator to ensure the highest quality browsers are available on its devices, as well as the CMA's own recognition that choice architecture can be pro-user and pro-competition.

215.   Taking account of the above, the CMA has not identified any sound basis on which to establish that Apple's choice architecture lowers user awareness of or engagement with browsers, that that there is any causation between Apple's conduct and low user awareness or engagement.

C.   The CMA's survey dispels the notion that Apple's choice architecture provides a competitive advantage to Safari

216.   The CMA's own user survey provides conclusive evidence that Apple's choice architecture does not prevent users from making effective decisions about their choice of browser in a way that could provide Safari a competitive advantage. Apple draws the CMA's attention in particular to the following CMA survey findings:

- A majority of iOS users (53%) have downloaded at least one other browser. Further, 60% of iOS users said they are "definitely" able to download alternative browsers, and 27% said they "probably" can do this unassisted. These findings reveal that users are aware that they can switch the default and that they do so in practice.

- Users do not perceive or experience barriers to switching, as revealed by the survey findings that: (i) 78% of users reported that they could definitely or probably change the default browser setting in their iOS device without assistance; and (ii) 89% of iOS users that had changed their browser said they found it very easy or fairly easy to do so. This clearly shows that users do not experience difficulty in navigating its system UI in order to switch.[153]

217.   Among iOS users that had not changed their default browser, the most common reason (34%) was that their current default browser was their preferred browser. For 24%, the web browser used is not considered important. The majority of users who continue with their default browser primarily do so for reasons relating to quality rather than any possible 'frictions' in the user journey.

D.   Conclusion

218.   As the above sections show, Apple's choice architecture is pro-consumer and pro-competitive. Rather than impeding competition, Apple's choice architecture design and policies actively support third-party browsers and promote user choice. Apple therefore urges the CMA to rethink its emerging findings in relation to Apple's choice architecture.

---

[153] The CMA attempts to undermine this clear evidence by stating that "*consumer self-assessed confidence in their ability to change default browser may not necessarily reflect their experience*" (paragraph 3.50). However, the CMA has obtained no evidence to support such an assertion, nor is there any basis from the survey findings to consider that this would apply to the users surveyed.

VIII. CONCLUSION

219. Mobile browsing on iOS and across platforms is currently well-functioning and Apple faces robust competition to attract and retain users. Apple's business decisions are driven by its incentives as a manufacturer of devices and its focus on delivering secure, private, and high-performing products that enhance the user experience.

220. This is demonstrated by a considerable body of evidence:

- there is a large and diverse set of differentiated browsers available on iOS, with the level of differentiation observed on iOS equivalent to that on Android;

- there is effective parity in the access to features and functionality between Safari and third-party browsers;

- the majority of developers are happy with the support and options available to them on iOS, including with respect to in-app browsing;

- WebKit and Safari are characterized by a high rate of innovation, which is driven by Apple's significant and long-standing investments in those products;

- users are informed and able to make effective switching decisions; and

- extremely high rates of satisfaction persist amongst iPhone users, reflected in survey evidence.

221. Properly assessed, the evidence shows that the market for mobile browsing on iOS and across platforms is well-functioning and there is no basis on which WebKit, or any other aspect of the mobile browsing experience on iOS, could be viewed as having anything but a positive impact on competition.

****