Vivaldi Technologies AS, Mølleparken 6, Oslo, Norway

Response to Strategic Market Status Investigations into Apple's and Google's mobile ecosystems Invitation to Comment

Vivaldi, launched in 2016, is a powerful, personal & private web browser (for desktop, mobile and in-car) that adapts to its users and offers more features than any other modern browser.

Vivaldi has two ground rules: privacy is a default, and everything's an option. In practice, this means building software that protects users' privacy but also does not track how they use it. Vivaldi believes private and secure software should be the rule, not the exception.

Vivaldi is headquartered in Norway, with satellite offices in Iceland and USA. It has no external investors and is co-owned by its approximately 50 employees.

We currently have 3,100,000 active users world-wide, [redacted] of whom are in UK (as are [redacted] employees).

Q1: Do you have any views on the scope of our investigations and descriptions of Apple's and Google's mobile ecosystem digital activities?

We're very encouraged that CMA is considering mobile browsers and the iOS rendering engine monopoly, the latter of which impedes our ability to develop our products as we would like.

Q2: Do you have any submissions or evidence related to the avenues of investigation set out in paragraph 70-72? Are there other issues we should take into account, and if so why?

Although the Digital Markets Act theoretically allows for third party rendering engines on iOS, Apple’s terms and conditions in its Web Browser Engine Entitlement mean that we don’t anticipate any vendor producing one, as the risks are too great.

Q4: Which potential interventions should the CMA focus on in mobile ecosystems? Please identify any concerns relating to Apple’s or Google’s mobile ecosystems, together with evidence of the scale and/or likelihood of the harms to your business; or to consumers.

We are separating our answers as our competitive problems with each of the two ecosystems are very different.

Google Play Store

Vivaldi depends on Google to approve updates and releases on its Play Store. We have occasionally seen delays of a week, which is a significant lag in a eight-weekly cycle of updates of Chromium, the browser engine used by most Android browsers. (We use Chromium Extended Stable, which has an 8 week release cycle, subject to variations for holiday periods etc.) This can mean that security updates of Chromium are not distributed to Vivaldi users as quickly as they are to Chrome users.

We’re not accusing Google of malice; it seems as if our updates occasionally get stuck in some automated approval mechanism that denies our builds with inscrutable error messages. The main problem is that there is no named contact to speak to in order to resolve any blockers, so we’ve had to ask people we know in Google if they know anyone, etc.

We suggest that Google provides a named management contact to each browser vendor that uses the Play Store to distribute its products so that delays can be easily and swiftly resolved.

We are somewhat surprised this isn't already the case, for the sake of transparency, given that Google controls both the Play Store and Chrome, the dominant browser on Android.

Google Search

Vivaldi does not have any revenue share agreement with Google Search, but many other independent browsers do. We have given our opinions on this ecosystem, and the confidential background that informs them, in our response to 'Strategic Market Status Investigation into Google's General Search and Search Advertising services Invitation to Comment", so will not repeat them here.

Apple App Store

After initial teething problems on launching our iOS product, our experience distributing Vivaldi through the App Store has been trouble-free. Updates are usually approved in around 24 hours.

Apple iOS

Paragraph 85 of the Invitation to Comment notes "Weak competition in native app distribution", but Vivaldi is more concerned how Apple weakens competition on iOS from web applications, self-preferencing native iOS apps which require developers to purchase a license from Apple, a MacOS computer, and to use its App Store with associated payments.

As we wrote in Vivaldi's response to Working Paper 7[1]

> Vivaldi exists to give users access to the Web in a way that they control, while protecting them as far as possible from surveillance and other bad actors …
>
> Except in some very small niches (some gaming, systems very tightly coupled to specific hardware features) we believe the web, built on mature open technologies, should be the delivery mechanism for software, rather than some vendor-controlled proprietary technology. Consumers should be free to access that software on whichever browser and device they prefer.

In order that businesses and developers can take advantage of the cross-platform ubiquity of mature web technologies, and avoid paying for Apple

[1] https://assets.publishing.service.gov.uk/media/673f34d3b3f0df6d2ebaf05b/Vivaldi_-_WP7_response.pdf

Developer licenses, submission to the App Store and any associated fees, the CMA could mandate the interoperability of all hardware features, such as Bluetooth, AirDrop, NFC, with Apple's system-supplied WebKit framework.

While Apple is certainly within its rights to make a business decision not to implement certain device integrations in its own Safari browser, the system WebKit framework could allow developers of Web Apps to interoperate with all APIs that connect to device hardware features, without compromising security; as Apple said in February 2022, in its response to UK CMA's Mobile Ecosystem Market Study interim report,

> "WebKit's sandbox profile on iOS is orders of magnitude more stringent than the sandbox for native iOS apps" https://assets.publishing.service.gov.uk/media/62277271d3bf7f158779fe39/Apple_11.3.22.pdf

This would require PWAs ("home screen web apps", in Apple's parlance) to be managed by the browser that installs them, so Safari-managed PWAs would only have access to hardware features that Apple wants Safari to be able to access, whereas PWAs installed with Vivaldi (for example) would be able, via the newly full-featured system Webkit, to access Bluetooth, NFCs, AirDrop etc.

Enabling PWAs to interact with device features with the greater-than-native protection provided by the web sandbox would lower the cost of entry to the market for businesses, potentially bringing new capabilities and cheaper connected devices to the consumer.

However, the WebKit restriction for all iOS browsers is onerous. Vivaldi is a small organisation of 57 people (including all non-development staff) and the majority of our products are built using Chromium, based on the Blink engine.

The only exception is for iOS, where we are compelled by Apple to code for the system WebKit "black box". We gave a concrete example in Vivaldi's Request for Information (MBCG VI RFI-3),

> "The result is that we need to maintain a complex, separate implementation of our ad blocker for iOS which provides generally worse functionality and user experience, as well as requiring developer resources to cater for the only Operating System that controls the browser engines allowed to run on it."

We could provide a better product for our customers, in less time and for less cost, if Apple were required to allow non-WebKit web engines in third-party browsers to run PWAs that can access all iOS APIs (including currently-private APIs, such as the ability to programmatically add PWAs to the device Homescreen which is currently only available from the system Share menu).

(Note that “access to all iOS APIs” requires documentation of those APIs; developers can’t meaningfully code to APIs that they don’t know exist, or which change regularly.)

Choice architecture

Section 83(b)(iii) of the invitation to comment mentions “Requirements for Apple and Google to make changes to choice architecture… to enable users of mobile devices to make active and informed choices about the product or services they use and/or set as a ‘default’ service.”

Vivaldi believes that the selection of candidates for the choice screen is of paramount importance.

1. Candidates for the choice screen should be general-purpose web browsers aimed at end-users.

We believe that Apple has attempted to erode trust in the EU’s choice screen and exclude other competitors by deliberately including browsers that aren’t useful to end-users, for example, by including

[redacted]

2. Cross-platform browsers should take priority. If it is a browser that is available on all platforms, it is more likely to be a major competitor.

3. Browsers that contain their own compiled code should take priority as a candidate. If their vendors compile the code themselves (rather than simply wrap a third party’s core), they are more likely to receive quick security and privacy updates.

4. Browsers that are updated frequently should take priority, as they are more likely to receive quick security and privacy updates.

5. In the case of Android, the only OEM browser that should be a candidate is that of the manufacturer of the device.

Design and functionality of choice screens

We also believe that the design and functionality of the choice screen is of vital importance in reducing Apple and Google's ability as controller of the Operating System to self-preference their own browsers. Therefore, the order of display should be randomised each time the screen is displayed. Selection should only be possible once the user has scrolled through all the choices, so that users whose font size is large for accessibility reasons are aware of all the options. The choice screen should be very obviously scrollable if there are options "below the fold".

While not part of a choice screen, an additional 'choice' if a user installs a third-party general-purpose browser (e.g., a choice screen candidate as listed above) but has not set it as default replacing Safari on iOS, or Chrome on Android would educate users and offer more competitive opportunity.

So if, for example, a user installs Vivaldi on iOS but did not know how to set it as default, when default Safari is activated, a system message saying "You downloaded Vivaldi; do you want to set that as default?" with a yes/ no button could be shown (with 'do not ask again' choice).

Q5: Are the potential interventions set out above likely to be effective, proportionate and/or have benefits for businesses and consumers?

Yes.

Q6: What key lessons should the CMA draw from interventions being considered, imposed and/or implemented in relation to mobile ecosystems in other jurisdictions?

Vivaldi suggests the CMA specify requirements so that firms designated with strategic market status cannot employ circumvention tactics. The European Commission's recent specification proceedings "to assist Apple in complying with its interoperability obligations under the Digital Markets Act" [2] produced some excellent preliminary findings [3] that could serve as a model for the level of detail necessary to ensure openness, transparency, a mechanism for

[2] https://digital-markets-act.ec.europa.eu/commission-starts-first-proceedings-specify-apples-interoperability-obligations-under-digital-2024-09-19_en

[3] https://digital-markets-act.ec.europa.eu/commission-seeks-feedback-measures-apple-should-take-ensure-interoperability-under-digital-markets-2024-12-19_en

requesting and receiving information in a timely manner, and a concilation process.

We urge the CMA to ensure that firms with SMS status cannot impose restrictive contracts, or fees that make it difficult (or impossible) for competitors to interoperate with Apple and Google's mobile ecosystems.

Finally, we remind the CMA of the importance of Desktop browsers across platforms. For many independent browser companies, mobile products are a gateway to consumers finding our desktop products. For many, Desktop is where the majority of their users are, and therefore generate the revenue needed to keep developing the products (and for the company to remain afloat).

Browsers are a key enabler of other important services such as AI, and productivity tools, and the browser market has great potential, with direct economic impact. Desktop continues to be an important way for users to access the internet and apps, as well as for developers to be found by users. This is especially true in the important educational and enterprise contexts.