

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,

Petitioner,

v.

HEADWATER RESEARCH LLC,

Patent Owner.

IPR2026-00049

Patent No. 9,179,359

PATENT OWNER'S RESPONSE TO THE PETITION

TABLE OF CONTENTS

I.	Introduction	1
II.	Overview of the '359 Patent	2
III.	Level of Ordinary Skill in the Art	5
IV.	Claim Construction	6
V.	The Petition's Ground 1 fails.	6
	A. The Petition's Shell-Cole theory fails with respect to limitation [1e].....	6
	B. The Petition's Shell-Cole theory additionally fails with respect to claim 26.	10
VI.	The Petition's Grounds 2–3 (Shell-Cole in combination with Flack) fail.	16
	A. The Petition's Ground 2 fails for the same reason as the Petition's Ground 1 fails with respect to limitation [1e].	16
	B. The Petition's Ground 3 fails because, like Shell-Cole, Flack does not disclose or render obvious basing an active-application determination on user interaction.	18
	C. The Petition's Ground 3 theory likewise fails because it is not adequately articulated.	21
VII.	Conclusion	25

TABLE OF AUTHORITIES

Cases

<i>Personal Web Technologies, LLC v. Apple, Inc.</i> , 848 F.3d 987 (Fed. Cir. 2017).....	25
--	----

Statutes

35 U.S.C. § 316(e).....	6
-------------------------	---

PATENT OWNER'S EXHIBIT LIST

No.	Description
2001	Notice of Partial Settlement in <i>Headwater Research LLC v. Amazon.com Services LLC et al.</i> , No. 7:25-cv-00286-ADA (W.D. Tex. Nov. 10, 2025)
2002	Joint Motion to Dismiss in <i>Headwater Research, LLC v. Apple, Inc.</i> , No. 7:25-cv-00379-ADA (W.D. Tex.)
2003	Joint Motion to Dismiss in <i>Headwater Research LLC v. Samsung Electronics Co., Ltd. et al.</i> , No. 2:23-cv-103-JRG-RSP (E.D. Tex. Sept. 11, 2025)
2004	Order Granting Joint Motion to Dismiss in <i>Headwater Research LLC v. T-Mobile USA, Inc. et al.</i> , Nos. 2:23-cv-379-JRG-RSP and 2:23-cv-377-JRG-RSP (E.D. Tex. Nov. 13, 2025)
2005	Order Granting Joint Motion to Dismiss in <i>Headwater Research LLC v. AT&T Services, Inc. et al.</i> , No. 2:23-cv-00397-JRG-RSP (E.D. Tex. Nov. 13, 2025)
2006	Declaration of Erik de la Iglesia in Support of Patent Owner's Response
2007	<i>Curriculum Vitae</i> of Erik de la Iglesia
2008	Transcript of the Deposition of Dr. Kevin Butler (June 19, 2026)
2009	Excerpt of January 14, 2025 Trial Transcript in <i>Headwater Research, LLC v. Samsung Electronics, Co., Ltd.</i> , Case No. 2:22-cv-422-JRG (E.D. Tex.)
2010	Excerpts of July 17, 2025 Trial Transcript in <i>Headwater Research, LLC v. Cellco Partnership, d/b/a Verizon Wireless</i> , Case No. 2:23-cv-352-JRG (E.D. Tex.)

I. Introduction

The '359 patent claims a wireless end-user device that conserves its own scarce network and battery resources through differential, per-application traffic control: the device can indicate that a single “Internet data service” is unavailable to one application even while that *same* service remains available to a different application (as recited by limitation [1e]), and it can determine that an application is “active” based on a user interaction with that application (as recited by claim 26). The Petition challenges 26 (which depends from claim 1) based on Shell and Cole, alone (Ground 1) or combined with Flack (Grounds 2 and 3). Each ground fails.

First, the Shell-Cole combination maps the claimed “Internet data service” to a “connection” in Shell, but each such connection is established through a socket for, and bound to, the particular requesting application. A denied connection request therefore never concerns the same connection (i.e., the alleged “Internet data service”) that remains “currently available” to a different application, as claim limitation [1e] requires.

Second, Shell-Cole does not disclose or render obvious claim 26’s requirement that the device determine that an application is active “based on a user interaction” with that application. The Petition’s theory in this regard is that Shell checks whether or not a user initiated a given connection request, but Shell classifies and prioritizes connection requests by type, treating a user-initiated email connection

request no differently than an automated one performed on a fixed schedule.

Third, adding Flack (Grounds 2 and 3) cures none of these deficiencies. Flack classifies connection requests by port number, assuming, for example, that every message retrieved over a designated port reflects user activity even in the absence of user interaction. And neither the Petition nor Petitioner's expert articulates how Flack's alleged determination of application status would actually be incorporated into Shell-Cole's decision logic. Pressed at deposition, Petitioner's expert could offer only an improvised theory that appears to *disregard* a user's interaction with an application if the application were unauthorized.

Accordingly, the sole challenged claim (dependent claim 26) should be found not unpatentable.

II. Overview of the '359 Patent

Mobile devices have finite network capacity and battery life, both of which can be consumed by applications, even absent user interaction. The '359 patent identifies email synchronization, RSS feed updates, social-networking applications, and similar "persistent signaling" applications as common sources of this problem, explaining that "*even when the user is not directly interacting with* or benefiting from this type of application, the application can be running in the background and continuing to consume potentially significant network resources." Ex. 1001, 8:8–22 (emphasis added). Rather than relying on "purely central/core network based

approaches,” the patent addresses this problem by “controlling network service usage activities at the source of the demand—the device.” *Id.*, 9:23–46.

The claimed wireless end-user device does so by applying a “differential traffic control policy” that selectively grants or restricts network access on a per-application basis. The device includes both a wireless wide area network (“WWAN”) modem and a wireless local area network (“WLAN”) modem, and one or more processors that apply the differential traffic control policy to Internet data service provided via the WWAN modem and communicate the results of that policy to individual applications through an application program interface, or “API.” *See* Ex. 1001, 92:7–11 (describing a “network access API” that “facilitates communication of... network access conditions”).

Independent claim 1 recites (limitation labels in brackets follow the numbering used in the Petition and throughout this Response):

[1pre] A wireless end-user device, comprising:

[1a] a wireless wide area network (WWAN) modem to receive and transmit Internet data between the device and at least one WWAN when configured for and connected to the at least one WWAN;

[1b] a wireless local area network (WLAN) modem to receive and transmit Internet data between the device and at least one WLAN when configured for and connected to the at least one WLAN; and

[1c] one or more processors configured to apply a stored WWAN differential traffic control policy to Internet data

service provided using the WWAN modem and the at least one WWAN, and

[1d] indicate to a particular application, via an application program interface (API), one or more network access conditions based on the applied WWAN differential traffic control policy,

[1e] wherein the one or more network access conditions includes a network access condition that *indicates the unavailability to the particular application of an Internet data service that is currently available via the WWAN modem to a different application*, but that is indicated as unavailable to the particular application.

Ex. 1001, 105:50–106:4 (emphasis added).

Limitation [1e] illustrates an important concept of the invention’s differential, per-application control: it requires that a *single* Internet data service be simultaneously indicated as unavailable to one (“particular”) application while remaining available to a “different” application. Thus, the device does not need to disconnect from the network altogether to conserve resources, but selectively restricts network access to specific applications, so that (for example) an application the user is not interacting with can be denied access to the Internet data service without disrupting applications with which the user *is* interacting. Ex. 2006, ¶¶23–27.

Claim 26 depends from claim 1 and adds a further requirement:

[26] The wireless end-user device of claim 1, wherein the one or more processors are further configured to determine that the different application is an active application *based on a user interaction with the different application*.

Ex. 1001, 108:11–14 (emphasis added).

This limitation explicitly ties the device’s determination that an application is “active” to user interaction with that particular application. The specification similarly teaches this solution: “a network service usage activity is determined to be an active application or process (e.g., based on a user interaction with the device and/or network service usage activity, such as a pop-up and/or other criteria/measures).” Ex. 1001, 90:14–18. This solution ensures that the differential traffic control policy targets applications according to a user’s interaction with them. Ex. 2006, ¶¶28–29.

III. Level of Ordinary Skill in the Art

The Petition proposes that a POSITA in the field of the ’359 patent “would have had at least a bachelor’s degree in computer science, electrical engineering, or a similar field, and approximately two years of industry or academic experience in a field related to computer software development and/or computer networking,” and that “[m]ore related education would compensate for fewer years of work experience (and vice versa).” Pet. 6. For purposes of this Response, Patent Owner and its expert apply the Petition’s proposed level of ordinary skill in the art. Ex. 2006, ¶30.

IV. Claim Construction

The Petition states that “no terms need construction to resolve the controversy in this forum.” Pet. 7. Patent Owner agrees that for evaluation of the theories articulated in the Petition, no terms need be construed.

V. The Petition’s Ground 1 fails.

The Petition’s Ground 1 alleges that the combination of Shell and Cole renders obvious independent claim 1 and its dependent claim 26. Petitioner bears the burden of proving unpatentability by a preponderance of the evidence. 35 U.S.C. § 316(e). As shown below, Ground 1 fails because the Petition does not show that Shell-Cole discloses or renders obvious limitation [1e] or the additional limitation of claim 26.

A. The Petition’s Shell-Cole theory fails with respect to limitation [1e].

Limitation [1e] requires a *single* Internet data service that is simultaneously indicated as unavailable to a “particular application” at the same time that service is available to a “different application.” See Ex. 1001, 105:65–106:4 (requiring one or more processors configured to indicate “*the unavailability to the particular application of an Internet data service that is currently available via the WWAN modem to a different application*,” but that is indicated as unavailable to the particular application”); Ex. 2006, ¶31.

Furthermore, if the condition indicated the unavailability of a *different* Internet data service than one which was currently available, the claim would not be

satisfied. Thus, the “Internet data service” that is “indicated as unavailable to the particular application” must be the *same* Internet data service as the one that is “currently available . . . to a different application.” Ex. 2006, ¶31.

However, the Petition’s theory in this regard fails. The Petition’s theory maps the “Internet data service” to a “connection” in Shell using a “resource” (such as a WWAN modem). *See* Pet. 47 (noting that a resource in Shell constitutes a “WWAN modem” and a connection in Shell constitutes “Internet data service”); Pet. 53 (mapping “Internet data service via the WWAN modem” to “a connection using resource 502”).

Dr. Butler confirmed this theory at his deposition, stating that the claimed “Internet data service” is “the connection by which” a given application’s activity “would be facilitated,” implemented “through a socket.” Ex. 2008 at 59:16–60:1. But a specific *connection* to a WWAN modem via a socket is established for, and bound to, the requesting application. Accordingly, each application’s connection is a separate, application-specific Internet data service. Ex. 2006, ¶¶32–33.

Thus, when a “different application” is using a connection to access the Internet and a connection request from the “particular application” is denied, two entirely different connections are at issue. The first is the connection already in use by the “different application.” The second is the connection the “particular application” requested—a request that was denied, so the second connection is never

established. These are *different* connections, and thus the particular application is not requesting an “Internet data service” that is “currently available . . . to a different application.” And it likewise follows that an indication that the requested connection has been denied is not an indication of unavailability of an “Internet data service” which is “currently available” to a different application. Ex. 2006, ¶34.

In short, the service “indicated as unavailable to the particular application” is not the *same* service “currently available . . . to a different application,” because the connection the particular application requested (and was denied) is a different connection than the one already in use by the different application. And limitation [1e] requires that whatever the “Internet data service” is mapped to, the Internet data service that is “currently available to a different application” must be the *same* Internet data service that is indicated as unavailable to the particular application. Under the Petition’s mapping, the two cannot be the same service, and the Petition’s theory thus fails. Ex. 2006, ¶¶33–34.

Notably, although the Petition does not explicitly allege that the claimed “Internet data service” was WWAN connectivity generally (rather than a specific *connection* to a WWAN modem), that theory also fails. For instance, the Petition at various points *implies* (but never actually argues) that Internet connectivity generally could be the “Internet data service,” because the Petition contends, e.g., that “a notification that a connection to a resource is blocked is an indication of

unavailability *because the application is prevented from accessing the Internet.*”

Pet. 53.

But that argument, even if it had been properly preserved by the Petition, would have failed. A notification that a connection has been blocked is *not* an indication that “the application is prevented from accessing the Internet” (Pet. 53). As Petitioner’s expert acknowledged, “there would be nothing preventing a program from requesting multiple sockets for different types of activity.” Ex. 2008 at 60:8–15. And critically, a POSITA would understand that, in the context of Shell, a single application may access the Internet to reach one remote resource while simultaneously being denied a connection to a different remote resource. *See* Ex. 1005, [0033], [0051], [0070]; Ex. 2008 at 60:8–15; Ex. 2006, ¶35.

At his deposition, Dr. Butler acknowledged he had *not* analyzed whether a rejected connection request would inform an application whether that application is prevented from accessing the Internet, because he acknowledged that he had not considered “whether in the Shell-Cole combination an application could be actively using the network using one connection and then separately at the same time request to use the network using a second connection and have that second connection request be denied.” *See* Ex. 2008 at 60:20–62:14 (“I would have to give it some thought.... I haven’t specifically formed an opinion in my declaration regarding [that] particular scenario.”). The fact that an application could be *actively accessing*

the Internet when a separate connection request is denied confirms that denial of a connection request does *not* indicate that the application is prevented from accessing the Internet. Ex. 2006, ¶36.

Because a rejected connection request in the Shell-Cole combination does not indicate that the application cannot reach the Internet through *other* connections, it cannot indicate to the particular application that it is barred from accessing the Internet while a “different application” is accessing the Internet. Accordingly, any theory that maps the “Internet data service” to Internet connectivity fails. Ex. 2006, ¶¶36–37.

B. The Petition’s Shell-Cole theory additionally fails with respect to claim 26.

The Petition’s Shell-Cole theory additionally fails because the Petition fails to show that Shell discloses that its system is “configured to determine that the different application is an active application *based on a user interaction* with the different application,” as claim 26 additionally requires.

The stated logic of the Petition is that because “[i]n Shell-Cole, a user’s interaction with an application may instigate connection requests,” it therefore follows that Shell-Cole’s “processor... determine[s] that the different application is active.” Pet. 59 (“In Shell-Cole, a user’s interaction with an application may instigate connection requests, allowing its processor to determine that the different application

is active.”). This logic, however, falls apart because the determination of whether an application is classified as “active” (or is instead classified as “background”) does not in any way evaluate whether a user instigated a connection request in the context of Shell. Ex. 2006, ¶38.

For example, there is no disclosure that Shell would base its active-versus-background application status on *user activity*. The Petition alleges that “the connection manager registers that the user interacted with the email application, which initiated the request.” Pet. 61. But nowhere does Shell disclose that whether “the user interacted with the email application” to initiate a request is in any way relevant to the active application determination, and Shell certainly contains no disclosure that Shell’s “connection manager registers [whether] the user interacted with the email application” to initiate a given request as the Petition contends. *See* Pet. 61; Ex. 2006, ¶39.

For example, Shell is explicit that connection requests can be a result of a user request, or alternatively, can be automated. Ex. 1005, [0051] (“[S]ync engine 202 may request that a remote electronic mail box be synchronized.... Activating a control in a user interface... may initiate such requests.”); *id.*, [0071] (noting that alternatively, “an electronic mail application may register to synchronize an electronic mailbox every hour between 8:00 AM and 5:00 PM”). Google’s expert agreed with this understanding in both his declaration and deposition testimony. *See*

Ex. 1003, ¶126 (“Shell contemplates that certain applications (and their requests for network connection), such as email, may be initiated by the user or done automatically.”); Ex. 2008 at 16:13–17:12 (reiterating this opinion at deposition, in making clear that it is “possible for an email synchronization to occur without a user having specified the synchronization of the email account through a user interface”).

And although Shell teaches that the *type* of request (e.g., email synchronization requests as opposed to “messaging” requests—*see* Ex. 1005, [0057]) can be used in prioritization, nowhere does Shell disclose (or render obvious) that determination of application status would change in any way depending on whether a connection request was *automatically initiated* as opposed to *initiated by a user*. For instance, Shell does not disclose or imply that a user-initiated email synchronization performed at 8:30 AM would be treated any differently than an automated synchronization performed *without* user interaction at 8:00 AM pursuant to the hourly update schedule. *See* Ex. 1005, [0071] (discussing an automated update schedule). Instead, Shell discloses that email synchronization requests can be prioritized over other types of requests, with no indication that *whether or not the user initiated the request through interaction* would change the priority of the synchronization request. *See, e.g.,* Ex. 1005, [0017] (“[A]ssume that synchronizing corporate electronic mail has a higher priority than browsing stock quotes on the

Internet.”); *id.*, [0057] (“For example, requests from sync engine 202 may be assigned a higher priority than requests from messaging application 206.”); Ex. 2006, ¶40.

Patent Owner acknowledges that the Petition additionally cites to a disclosure in Shell that requests from “active applications” may be given higher priority than requests from applications that are “running in the background.” Pet. 60 (quoting Ex. 1005, [0057]). But that disclosure does not relate to “determin[ing] that the different application is an active application *based on a user interaction with the different application*,” as claim 26 requires. Instead, it simply discloses determining whether an application is *active*, with no disclosure that this determination is “based” in any way “on a user interaction with the different application.” Ex. 2006, ¶41.

Nor can Google contend that Shell’s disclosure of determining an “active” application as opposed to a “background” application is the same as a determination of user interaction. For instance, in a prior litigation, Google’s corporate representative testified that an accused system “classif[ies] applications as either foreground or background,” yet that same system does *not* “use interacting as the key difference between the two states.” Ex. 2009 at 499–500; *see also id.* at 533 (testifying “there’s a big difference” between “a determination as to whether an application is running in the foreground or background versus a determination as to whether an application is interacting in the device display foreground with the user”). And *another* Google corporate representative in another litigation in which Google

had a financial interest testified that an application providing “foreground services” might be non-“interactive.” Ex. 2010 at 277:15–278:1. And as explained by Headwater’s expert, it is likewise possible for a user to interact with a background application while the application is still in the background. Ex. 2006, ¶42. In other words, depending on the system, user interaction may not be determinative in evaluating background-versus-non-background classification, because an application can be classified as background in the presence of user interaction, or classified as active despite no user interaction taking place. Ex. 2006, ¶¶42–43.

Indeed, the ’359 patent’s own specification confirms what actual disclosure of this limitation looks like, and highlights that Shell contains nothing comparable. The specification expressly states that “[i]n some embodiments, a network service usage activity is determined to be an active application or process (e.g., based on a user interaction with the device and/or network service usage activity, such as a pop-up and/or other criteria/measures).” Ex. 1001, 90:14–18. Shell contains no equivalent disclosure tying its active/background classification to any user-interaction trigger. Tellingly, when Petitioner’s own expert was asked at his deposition whether Shell (or the ’359 patent) discusses “active” application status in this sense, he testified that “there is no specific discussion of active -- with regards to application in the -- in this or in the ’359 itself.” Ex. 2008 at 22:19–23. Ex. 2006, ¶¶44–45.

Dr. Butler’s testimony in this regard is correct as to Shell, such that there is no dispute that Shell does not disclose how an “active application” classification would be based on user interaction. However, his testimony is *incorrect* as to the ’359 patent, which (as shown above) expressly teaches an active-application determination based on a user interaction with that application. Ex. 1001, 90:14–18 (teaching that “a network service usage activity is determined to be an active application... based on a user interaction with the... network service usage activity”).

Thus, the Petition cannot show that the limitation is disclosed or rendered obvious merely by pointing to a classification between “background” and “active” applications; the Petition must show that the “active application” determination is “based on a user interaction with the different application” as the claim requires. The Petition sets forth no plausible theory under which Shell discloses this limitation.

Nor does the Petition’s Ground 1 set forth any motivation to *modify* Shell’s disclosure to base its determination “on a user interaction” as claimed. Instead, the Petition’s Ground 1 alleges only that this concept is *disclosed* in Shell, and sets forth no obviousness theory based on a modification to Shell in this regard. *See generally* Pet. 47–58.

VI. The Petition’s Grounds 2–3 (Shell-Cole in combination with Flack) fail.

As explained below, the Petition’s Grounds 2 and 3 (which involve a combination of Shell, Cole, and Flack) do not cure the deficiencies of the Shell-Cole combination discussed above.

A. The Petition’s Ground 2 fails for the same reason as the Petition’s Ground 1 fails with respect to limitation [1e].

The Petition’s Ground 2 alleges the obviousness of claim 1 based on Shell-Cole in combination with Flack. However, this combination does not cure the deficiencies discussed above relating to the Shell-Cole combination, standing alone, as to limitation [1e].

As explained regarding limitation [1e] above, in Shell-Cole informing an application that *the requested* connection has been denied does not inform the application that an “Internet data service” is unavailable, while the same “Internet data service” is “currently available to a different application,” because the *requested connection* (i.e., the alleged “Internet data service”) is not available to any other application.

The Petition’s Ground 2 analysis does nothing to cure this deficiency. The Petition’s Ground 2 does not allege (much less show) that the Shell-Cole and Flack combination would deny any application’s request for connection, while that *same* connection was “currently available” to a different application. Nor could the Petition do so because the requested connections are application-specific. Ex. 2006, ¶¶47–48.

And to the extent the Petition contends that for its Ground 2 theory, “network connectivity” more generally is the “Internet data service” that is indicated as unavailable, that does not work because just as in the Shell-Cole combination, nothing in the proposed Shell-Cole-Flack combination indicates to an application that network connectivity is unavailable to the requesting application. Instead, a denied connection request merely informs the application that the *specific requested* connection is unavailable, even though network connectivity may be available to the application for other connections.

Indeed, Flack itself is clear on this point. The Petition acknowledges that in Flack, a connection may be allowed in part based on Flack’s “client determin[ing] that communications activity represents user actions.” Ex. 1007, [0019] (quoted by Pet. 71). Flack provides an explicit example of this with respect to an email application, wherein mail being “Received over Port 1017” is assumed to represent user activity, but updating “Mailbox settings over Port 1015” is assumed to constitute “NONUSER ACTIVITY,” as shown in Flack’s FIG. 2, excerpted below:

COMMUNICATIONS CONFIGURATION

Instant messenger Settings (110)

Sent over Port 1020

Received over Port 1021

Mail Settings (120)

Mailbox settings over Port 1015 (NONUSER ACTIVITY) (HOST AGENT CAPABLE TO KEEP ALIVE CONNECTION)

Sent over Port 1016

Received over Port 1017

Ex. 1007, FIG. 2; *id.* at [0027]–[0028] (“Mail settings 210 indicate that three ports are being used... for three different activities.... ‘Mailbox’ communications do not represent user activities while ‘send message’ and ‘receive message’ are determined to represent user activities.”); Ex. 2006, ¶49.

Thus, if Flack’s system denies a network connection request from the email application because a request pertaining to port 1015 (mailbox settings) is assumed to constitute nonuser activity, that does not indicate to the email application that network connectivity more broadly is unavailable. Rather, network connectivity may very well be available to the email application for mail sent over port 1016 or mail received over port 1017, such that a denied connection request does not indicate unavailability to the application of the same network connectivity that is currently available to a different application, as limitation [1e] requires. Ex. 2006, ¶50.

B. The Petition’s Ground 3 fails because, like Shell-Cole, Flack does not disclose or render obvious basing an active-application determination on user interaction.

With respect to claim 26, the Petition’s theory fails because just like Shell, Flack *also* does not disclose any determination of application status based on user activity. Instead, Flack discloses classifying types of *connection requests* according to the category of connection being established, such that the same connection would be given the same priority *regardless* of whether or not the user interacted with the application. Ex. 2006, ¶51.

For example, Flack teaches that updating “Mailbox settings over Port 1015” is assumed to constitute “NONUSER ACTIVITY,” whereas mail being “Received over Port 1017” *does* represent user activities. Ex. 1007, FIG. 2 & [0027]–[0028]. This determination is made on the basis of *which port* is used to request the connection, not whether that connection request was in fact initiated by user interaction. For instance, Flack teaches that its classification logic assumes that “packets using a first port number represent a user request to transmit a message, (2) packets using a second port number represent a user request to retrieve a message, and (3) packets using a third port number represent a ‘keep-alive’ message generated by a background process.” *See* Ex. 1007, [0018]. But a POSITA would also understand that receiving mail can be reflective of user interaction with an app *or not*, depending on whether the user requested that the email app check for mail; as any user of email applications will be familiar, email applications can both send and receive emails automatically, without any user interaction with the application. For instance, a mail application may check for new messages every five minutes, even with no user interaction, thereby triggering a message retrieval connection request. In such a scenario, Flack will *assume* the message retrieval request “represent[s] a user request to retrieve a message” (because port 1017 for message retrieval would be used), even though there was no user interaction with the application. Ex. 2006, ¶¶52–53.

Because Flack’s purported evaluation of user interaction is strictly based on the port number the application is requesting, which may or may not correspond to any user interaction at all, Flack does not disclose that its system is “configured to determine that the different application is an active application *based on a user interaction with the different application*” as claim 26 requires. Instead, Flack merely looks at the port number requested and uses that port number to evaluate connectivity rules, without regard to any actual user interaction with the application. Ex. 2006, ¶54.

Notably, Flack’s approach (looking at the type of connection requested to determine whether a connection request should be allowed, as opposed to making an active application determination based on user interaction) is also consistent with the teachings of the primary reference Shell, which discloses that synchronization of an electronic mailbox can be *either* user-initiated or automatic (and thus not involving user interaction), and that such synchronization should be prioritized similarly regardless of whether or not that synchronization was user-initiated. *See Section V.B, supra*. A POSITA would understand that, like Shell, Flack would treat email synchronization the same way *regardless* of how it was initiated, because Flack merely looks at the type of communication rather than any details regarding how that communication was requested. For example, assuming that email synchronization used port 1017 to request messages, Flack is clear that such a request would be treated

the same way *regardless* of whether the request was user-initiated or not. Ex. 2006, ¶54.

C. The Petition’s Ground 3 theory likewise fails because it is not adequately articulated.

The Petition alleges that its Shell-Cole combination with Flack renders obvious claim 26 because allegedly, “[a] POSITA would have found it obvious to implement Flack’s determination of application status based on user activity as part of Shell-Cole’s traffic policy.” Pet. 73. However, the Petition does not articulate *what* it contends in Flack constitutes “Flack’s determination of application status based on user activity.” *Id.* As explained in the preceding subsection, Flack does *not* make any determination of application status based on user activity. Ex. 2006, ¶55.

And Google’s expert, when asked at his deposition what in Flack discloses this alleged determination, acknowledged that he had not analyzed this in his declaration:

How does Flack determine application status based on user activity?

A. Well, to -- to the previous discussion, the paragraph 50 in Flack and then the subsequent paragraphs discusses the identification of applications by Flack, including the -- the identification of the application identifying how it's being used and information

indicative of communications representing user actions.

Q. Okay. So I understand that paragraph 50 in Flack and subsequent paragraphs talk about that there is information indicative of user actions.

My question is, specifically, how does Flack make any specific determination of application status based on user activity?

MR. MILKEY: Objection. Form.

THE WITNESS: I didn't specifically analyze this in my declaration, but, again, I think there -- there -- there are paragraphs within Flack that discuss for different types of applications the communications that represent user actions.

Ex. 2008 at 38:19–39:13 (highlighting added). When pressed on the issue, Dr. Butler stated that an identification of which “protocols” are used for communication constitutes a “determination of user activity” (*id.* at 41:12–42:9), but neither the Petition, Dr. Butler, nor Flack establishes that looking at what protocol is used for communication constitutes making a determination of “active application” status that is “based on a user interaction with the different application” as claim 26 requires. To the contrary, a POSITA would readily understand that the *protocol* used for a communication is independent from whether a *user interacted* with the

application to trigger transmission of a communication over that protocol. Ex. 2006, ¶¶55–56.

Furthermore, *even assuming* Flack discloses a “determination of application status based on user activity” (Pet. 73)—it does not, as discussed above—the Petition’s Ground 3 would still fail because the Petition fails to explain how integrating any such “determination of application status based on user activity” into “Shell-Cole’s decision logic” (Pet. 73) would actually operate to produce the claimed result. *See generally* Pet. 73–76 (not describing *how* Flack’s supposed “determination of application status based on user activity” would be implemented in the Shell-Cole decision logic to modify that logic).

An assertion that one reference’s teaching would become “part of” another reference’s decision logic does not describe what the resulting logic would actually look like. And without articulating how a proposed combination of references would operate, a party challenging validity cannot meet its burden of establishing that its proposed combination would (1) in fact operate in a manner that satisfies the challenged claim or (2) be obvious for a POSITA to implement.

Dr. Butler could not supply that critical missing detail at his deposition. When asked directly how any teaching of Flack would modify the decision logic in the proposed combination, Dr. Butler could not answer other than to say that Flack “provides implementation detail and provides additional context.” Ex. 2008 at 46:12–

47:6. Eventually, after continued questioning, Dr. Butler was able to offer a single theory as to how Shell-Cole’s decision logic might arguably change in view of Flack: Flack’s classification of a program as “unauthorized” could “short circuit” the Shell-Cole decision logic, such that “no further determination needs to be made about whether it is an active application.” *Id.* at 52:14–53:3. He also testified that he “didn’t specifically opine about unauthorized applications in [his] declaration,” but implied that under his new proposal, the unauthorized application would “not [be] considered active” even if a user was interacting with it. *Id.* at 54:11–55:5. Ex. 2006, ¶¶57–58.

In other words, Dr. Butler could not articulate how Flack could be used to modify Shell’s decision logic other than to create a new theory not presented with the Petition. And even then he was unable to explain how such a system would base its determination on ***user interaction*** as opposed to whether or not an application was authorized or not (irrespective of whether a user interacted with the unauthorized application).

The Petition’s conclusory statement that a POSITA would have “implement[ed] Flack’s determination . . . as part of Shell-Cole’s decision logic” (Pet. 73) is thus unsupported by any articulation of ***how*** this implementation would be made, much less a showing that such an implementation either (1) would have arrived at the challenged claim or (2) would have been obvious for a POSITA to implement.

This is precisely the defect the Federal Circuit identified in *Personal Web Technologies, LLC v. Apple, Inc.*, where the court held that a finding that two references could be combined “is not enough,” because it “does not imply a motivation to pick out those two references and combine them to arrive at the claimed invention.” 848 F.3d 987, 993–94 (Fed. Cir. 2017). The court faulted the Board’s finding of invalidity because “the Board nowhere clearly explained, or cited evidence showing, how the combination of the two references was supposed to work,” and held that “such a clear, evidence-supported account of the contemplated workings of the combination is a *prerequisite* to adequately explaining and supporting a conclusion that a relevant skilled artisan would have been motivated to make the combination and reasonably expect success in doing so.” *Id.* (emphasis added). In other words, without a clear explanation of how the combination of references would operate, subsidiary questions of obviousness cannot be addressed. The Petition’s bare assertion that a POSITA would have “implement[ed] Flack’s determination . . . as part of Shell-Cole’s decision logic” (Pet. 73), unaccompanied by any evidence-supported account of how that combination would actually work, thus is insufficient to establish obviousness.

VII. CONCLUSION

For the reasons addressed above, the Petition fails to establish that the sole challenged claim—claim 26—is unpatentable.

Date: July 8, 2026

Respectfully submitted,

/James A. Milkey/

James A. Milkey, Reg. No. 79,503
RUSS, AUGUST & KABAT
12424 Wilshire Blvd., 12th Fl.
Los Angeles, CA 90025

Counsel for Patent Owner

CERTIFICATION REGARDING WORD COUNT

Pursuant to 37 C.F.R. §42.24(d), Patent Owner certifies that there are 5,448 words in the paper excluding the portions exempted under 37 C.F.R. §42.24(a)(1).

Date: July 8, 2026

Respectfully submitted,

/James A. Milkey/
James A. Milkey, Reg. No. 79,503
RUSS, AUGUST & KABAT
12424 Wilshire Blvd., 12th Fl.
Los Angeles, CA 90025

Counsel for Patent Owner

CERTIFICATE OF SERVICE (37 C.F.R. § 42.6(e))

The undersigned hereby certifies that the above document was served on July 8, 2026 by filing this document through the Patent Trial and Appeal Case Tracking System (PTACTS) as well as delivering a copy via electronic mail upon the following attorneys of record for Petitioner:

Erika H. Arner, Reg. No. 57,540
Daniel C. Tucker, Reg. No. 62,781
Elliot Cook, Reg. No. 61,769
Michael V. Young, Reg. No. 61,180
Alexander M. Boyer, Reg. No. 66,599
Bradford C. Schulz, Reg. No. 75,006
Christopher B. Anderson, Reg. No. 77,898
Umbra Aggarwal, Reg. No. 76,098
Abhay A. Watwe, Reg. No. 59,461
Kathleen C. Galleher, Reg. No. 80,837
Finnegan, Henderson, Farabow, Garrett, & Dunner LLP
1875 Explorer Street, Suite 800
Reston, VA 20190-6023
erika.arner@finnegan.com
daniel.tucker@finnegan.com
elliott.cook@finnegan.com
michael.young@finnegan.com
alexander.boyer@finnegan.com
bradford.schulz@finnegan.com
christopher.anderson@finnegan.com
umbra.aggarwal@finnegan.com
abhay.watwe@finnegan.com
kathleen.galleher@finnegan.com

Joshua L. Goldberg, Reg. No. 59,369
Daniel F. Klodowski, Reg. No. 72,693
Pier D. DeRoo, Reg. No. 69,340
Christopher C. Johns, Reg. No. 68,664
Kassandra M. Officer, Reg. No. 74,083

William C. Neer, Reg. No. 78,874
Kai Rajan, Reg. No. 70,110
Connor M. McGregor, Reg. No. 79,634
Heather J. Ren, Reg. No. 82,089
Finnegan, Henderson, Farabow, Garrett & Dunner, LLP
901 New York Avenue NW
Washington, DC 20001-4413
joshua.goldberg@finnegan.com
daniel.klodowski@finnegan.com
pier.deroo@finnegan.com
Christopher.Johns@finnegan.com
Kassandra.Officer@finnegan.com
william.neer@finnegan.com
kai.rajan@finnegan.com
connor.mcgregor@finnegan.com
heather.ren@finnegan.com

Benjamin A. Saidman, Reg. No. 69,325
Finnegan, Henderson, Farabow, Garrett & Dunner, LLP
271 17th Street NW, Suite 1400
Atlanta, GA 30363-6209
Phone: 404-653-6510
Fax: 404-653-6444
benjamin.saidman@finnegan.com

Yi Yang, pro hac vice to be filed
Finnegan, Henderson, Farabow, Garrett & Dunner, LLP
2 Seaport Lane, 6th Floor
Boston, MA 02210
yi.yang@finnegan.com

HeadwaterIPRTeam@finnegan.com

Date: July 8, 2026

Respectfully submitted,

/James A. Milkey/

James A. Milkey, Reg. No. 79,503
RUSS, AUGUST & KABAT
12424 Wilshire Blvd., 12th Fl.
Los Angeles, CA 90025

Counsel for Patent Owner