

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM SERVICES LLC
and AMAZON WEB SERVICES, INC.,

Petitioners,

v.

HEADWATER RESEARCH LLC,

Patent Owner

IPR2026-00106
Patent No. 10,321,320

PATENT OWNER'S RESPONSE TO THE PETITION

TABLE OF CONTENTS

I.	Introduction	1
II.	Overview of the '320 Patent	2
III.	Level of Ordinary Skill in the Art	3
IV.	Claim Construction	4
	A. “periodic timer”	4
	B. “interprocess communication service”	7
V.	The Petition’s Grounds 1A–1G all fail under the Petition’s theory.....	9
	A. Overview of TS-23.140.	9
	B. The Petition fails to show that TS-23.140 discloses the claim limitations, even under the Petition’s claim interpretation of “message delivery trigger.”	13
VI.	The Petition’s Ground 1 additionally fails regarding claims 3 and 4.	19
	A. The Petition’s Ground 1A (TS-23.140 alone) fails with respect to claim 3.....	19
	1. The Petition’s “earliest desired time of delivery” theory fails.	19
	2. The Petition’s “periodic polling” theory also fails.	21
	B. The Petition’s Grounds 1B and 1C (TS-23.140 in combination with Adamczyk and/or Herzog) also fail with respect to claims 3 and 4.	27
VII.	The Petition’s Ground 1 additionally fails to establish the obviousness of claim 17.....	33
	A. The Petition fails to establish the obviousness of its proposed combination of TS-23.140 with Fok.....	34
	B. The Petition fails to show that its proposed combination, even if obvious, would satisfy the claim language.	37
VIII.	The Petition’s Ground 2 fails with respect to all challenged claims.....	40
	A. The Petition sets forth no plausible theory as to how “for at least some of the received network element messages, the receipt of such a message... is not a message delivery trigger.”	40

B.	The Petition fails to establish the obviousness of its proposed Houghton/Munson combination.	43
C.	The Petition’s Ground 2 additionally fails to establish the obviousness of claims 3 and 4.	49
1.	The Petition’s Grounds 2A and 2B fail with respect to claims 3 and 4.	49
2.	The Petition’s Ground 2C fails with respect to claims 3 and 4.	55
D.	The Petition’s Ground 2 additionally fails to establish the obviousness of claim 17.	59
IX.	Conclusion	62

TABLE OF AUTHORITIES

Cases

<i>ActiveVideo Networks, Inc. v. Verizon Commc'ns, Inc.</i> , 694 F.3d 1312 (Fed. Cir. 2012)	53
<i>KSR Int'l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007)	35, 48, 51, 52

Statutes

37 C.F.R. § 42.65(a)	39
----------------------------	----

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	DocketNavigator statistics for contested motions to stay pending IPR before Judge Alan Albright
2007	Exhibit C-7 to Amazon's December 12, 2025 invalidity contentions in <i>Headwater Research LLC v. Amazon.com Services LLC et al.</i> , No. 7:25-cv-00286-ADA (W.D. Tex.)
2008	Declaration of Michael C. Brogioli, Ph.D. in Support of Patent Owner's Response
2009	<i>Curriculum vitae</i> of Michael, C. Brogioli, Ph.D.
2010	July 9, 2026 deposition transcript of Dr. Patrick Traynor
2011	August 28, 2024 deposition transcript of Dr. Patrick Traynor
2012	Supplemental Declaration of Dr. Patrick Traynor, Submitted in IPR2024-00010

2013	The Authoritative Dictionary of IEEE Standards Terms (7th Ed. 2000) (excerpted)
2014	U.S. Patent No. 12,210,419, assigned to Amazon Technologies, Inc.
2015	Susmit Bagchi, <i>Distributed IPC Using Virtual Device Driver in Monolithic Kernel</i> , 2012 IEEE Int'l Conf. on Embedded & Real-Time Computing Systems & Applications (2012)
2016	Christophe Lizzi & Eric Gressier-Soudan, <i>A Real-Time IPC Service over ATM Networks for the Chorus Distributed System</i> , 1998 IEEE (1998)

I. Introduction

The Petition fails to establish that any challenged claim is unpatentable. As explained below, the Petition's grounds propose an interpretation of "message delivery trigger" so broad that any condition bearing on whether a message is ultimately delivered qualifies as a "trigger." Under that interpretation, however, the receipt of every received network element message in the Petition's mapped systems is itself a "message delivery trigger," and the Petition therefore cannot satisfy all challenged claims' requirement that "for at least some of the received network element messages, the receipt of such a message by the message buffer system is not a message delivery trigger." *See* Sections V, VIII.A below. The Petition's Houghton/Munson grounds additionally fail because the Petition never articulates a reason a POSITA would have combined Houghton with Munson's store-and-forward delivery model. *See* Section VIII.B below.

The Petition's theories for claims 3 and 4 further fail because none of the Petition's mapped systems delivers messages upon "the expiration of a periodic timer." Neither TS-23.140's "earliest desired time of delivery" nor the "periodic polling" the Petition attributes to the MMS User Agent is a periodic timer whose expiration causes delivery, and the Petition identifies no logic that would even detect such polling. Houghton's "periodic message" is a keep-alive that the Petition never shows causes delivery of anything, and the single paragraph of

Munson on which the Petition relies discloses no regularly-repeating timer. The Petition's fallback combinations with Adamczyk fail for the further reason that the Petition establishes neither a motivation to make them nor a reasonable expectation of success. *See* Sections VI and VIII.C below.

Finally, the Petition's theories for claim 17 additionally fail for two independent reasons. The Petition never identifies the "secure interprocess communication service" that claim 17 requires, much less shows that any such service is itself "secure"; it instead relabels Fok's application-to-application handshake as the claimed service. And the combinations the Petition proposes would not have been obvious, because each departs from what Fok actually teaches (direct communication between applications) for reasons the Petition never justifies. *See* Sections VII and VIII.D below.

II. Overview of the '320 Patent

At a high level, the '320 patent is directed to a networked system comprising both a network server system and the device link agents residing on a plurality of wireless end-user devices. The network server system receives messages from various network elements for delivery to software components identified in those messages, buffers the message content, and evaluates a plurality of message delivery triggers before delivering the buffered content to a given device over a secure Internet data message link. Every other challenged claim—claims 2-18—

depends, directly or indirectly, from claim 1. Ex. 2008, ¶¶32-33.

Relevant to the disputes raised herein, the claims require both that the claimed “network server system” includes “logic to determine when one of a plurality of message delivery triggers for a given one of the wireless end-user devices has occurred,” and *additionally* require that “for at least some of the received network element messages, the receipt of such a message by the message buffer system is *not* a message delivery trigger.” As explained below, applying a broad view of “message delivery trigger” in order to arrive at the first of these requirements, the Petition cannot show that the second of these requirements is satisfied under the Petition’s theory of invalidity. Ex. 2008, ¶¶33, 35.

III. Level of Ordinary Skill in the Art

The Petition proposes that a POSITA in the field of the ’320 Patent “would have had (1) at least a bachelor’s degree in computer science, electrical engineering, or a related field, and (2) 3-5 years of experience in services and application implementation in communication networks. Additional graduate education could substitute for professional experience, and vice versa.” Pet. 5 (citations omitted).

For purposes of this response, Patent Owner agrees that a person having a bachelor’s degree in computer science, electrical engineering, or a related field, plus at least three years of experience in services and application implementation

in communication networks qualifies as a POSITA. The three years of additional experience can be in either a professional setting or a research setting. Ex. 2008, ¶¶30-31.

IV. Claim Construction

A. “periodic timer”

As explained in this Section and in Sections VI.A and VIII.C below, certain of the Petition’s theories regarding the claim term “periodic timer” are inconsistent with that term’s plain and ordinary meaning. *See* Ex. 2008, ¶¶37-42.

The term “periodic timer” arises in the context of claim 3, which depends from claim 1 and further specifies that “one of the message delivery triggers is the expiration of a periodic timer.”

While the ’320 patent does not explicitly define “periodic timer,” a POSITA would understand it to refer to a timer that recurs in accordance with a given time period. Ex. 2008, ¶37. For example, The Authoritative Dictionary of IEEE Standards Terms (7th Ed. 2000) defines “periodic” as “[o]f or pertaining to a series of specified waveforms or features which *repeat or recur regularly... in time*,” as opposed to “aperiodic” in which the “waveforms or features repeat or recur... *irregularly in time*.” Ex. 2013, 13.

In the context of messaging systems, an excellent example of something that occurs *periodically* is an example the Petition itself raises: “periodic polling.” *See*

*Pet. 27 (relying on periodic polling as satisfying the claim 3 requirements, and acknowledging that “periodic polling is a periodic-timer trigger **because polling happens at regularly-repeating** intervals”); see also Section VI.A.2, *infra* (explaining why the Petition’s “periodic polling” theory fails for reasons *other than* the polling not satisfying the requirement of a “periodic timer”). A POSITA would understand that periodic polling is a scheme in which a system will “poll” (or inquire as to the status of some value) in a recurring manner at some defined frequency—the time between these recurring polls is the “period” of the periodic polling. Ex. 2008, ¶38.*

This understanding of “periodic” is consistent with Petitioners’ affiliate Amazon Technologies, Inc.’s usage of “periodic” in its own patents. *See, e.g., Ex. 2014 (U.S. Patent No. 12,210,419, assigned to Amazon Technologies, Inc.), claims 3, 10, 17 (noting that performing a task according to a set schedule is not necessarily “periodic”); Ex. 2008, ¶39.*

As explained in further detail in Sections VI.A and VIII.C below, the Petition and Dr. Traynor apply a far broader understanding of what constitutes a “periodic timer.” For instance, the Petition alleges that a timer ending “after a particular period” constitutes a periodic timer. *Pet. 26-27.* Dr. Traynor confirmed at his deposition that he applied the meaning of “periodic timer” wherein the timer is “related to a time period” such that a timer that runs “exactly once for a single

message” could constitute a “periodic timer.” Ex. 2010 at 28:20–29:14. And in a previous deposition, Dr. Traynor stated that “expiration of a periodic timer” might even be satisfied by a “user looking at the clock and waiting” until a particular time had passed before they sent a message. *See* Ex. 2011, 17:14-18:7; Ex. 2008, ¶40.

The Petition’s and Dr. Traynor’s definition of a “periodic timer” is inconsistent with that term’s ordinary meaning discussed above, and would have the effect of rewriting claim 3 to remove the term “periodic” as follows: “The networked system of claim 1, wherein one of the message delivery triggers is the expiration of a ~~periodic~~ timer.” The phrase “expiration of a... timer” alone implies that some time *period* is being measured. *See* Ex. 2013, 12 (“arm a timer: To start a timer measuring the passage of time, enabling the notification of a process when the specified time or time interval has passed.”). Thus, if all that were required for a “periodic timer” were a timer being “triggered and continu[ing] for a particular period” (Ex. 1003, ¶641), then *any* timer would be “periodic,” because all “timers” that can “expire” must necessarily expire after *some* period of time. Ex. 2008, ¶41. Indeed, at his deposition, Dr. Traynor was unable to identify *any* timer that would not be “periodic” under the Petition’s usage of that term. Ex. 2010 at 31:1–16 (“And sitting here right now, can you give me an example of a timer that is *not* relating to a time period? THE WITNESS: ... I don’t have any examples at the moment off the top of my head.”). But claim 3 specifically recites a *periodic* timer,

and the Petition's attempts to read "periodic" out of the claim language should be rejected. *See* Ex. 2008, ¶¶37–42.

B. "interprocess communication service"

Claim 17 requires that at least a given one of the devices further comprise "a secure interprocess communication service separately secured from the secure Internet data message link," and that the device link agent for that device cause "messages to be securely delivered to a software process by initiating delivery of each such message on the secure interprocess communication service." Ex. 1001, claim 17. The claim thus requires that (1) there be an identifiable "service" on which the messages are delivered, and (2) that the service itself is "secure." Ex. 2008, ¶¶48, 51.

A POSITA would understand an "interprocess communication service" to be narrower than "interprocess communication." Ex. 2008, ¶48. An "interprocess communication service" is a service that is provided to the applications by some component; whereas "interprocess communication" describes any communication between processes, however accomplished. *Id.* An example of this distinction is the difference between shared memory and D-bus. The operating system establishes a shared memory region, and one process may write a message that another process then reads. But once that region is established, the processes using it manage the communication themselves, including its timing, access control, and

the format of the exchanged data. Ex. 2008, ¶49. Shared memory is therefore a mechanism by which interprocess communication occurs, not an interprocess communication service. *Id.*

D-Bus, which the '320 patent describes as “a message bus system for inter-process software communication that, for example, helps applications/agents to talk to one another,” falls on the other side of that line. Ex. 1001, 43:24-31. Such a service defines how applications interact with one another and is thus a service for interprocess communications. Ex. 2008, ¶50.

As described by Dr. Brogioli, a real-world analogy is helpful in further illustrating this distinction. Telephone service and mail service are *communication services*; two neighbors who speak with one another across a fence *communicate* without using any *communication service* at all. *Id.* What separates an “interprocess communication service” from “interprocess communication” is whether a specific service is being provided that defines and manages the communication protocols between processes. *Id.*

Contemporaneous technical literature reflects the same understanding, describing interprocess communication “as a service from the underlying operating systems,” a “system-wide virtual layer of service to user applications,” and a “real-time interprocess communication service” implemented “as servers running in the micro-kernel space.” Ex. 2008, ¶52 (citing Exs. 2015–2016). That literature also

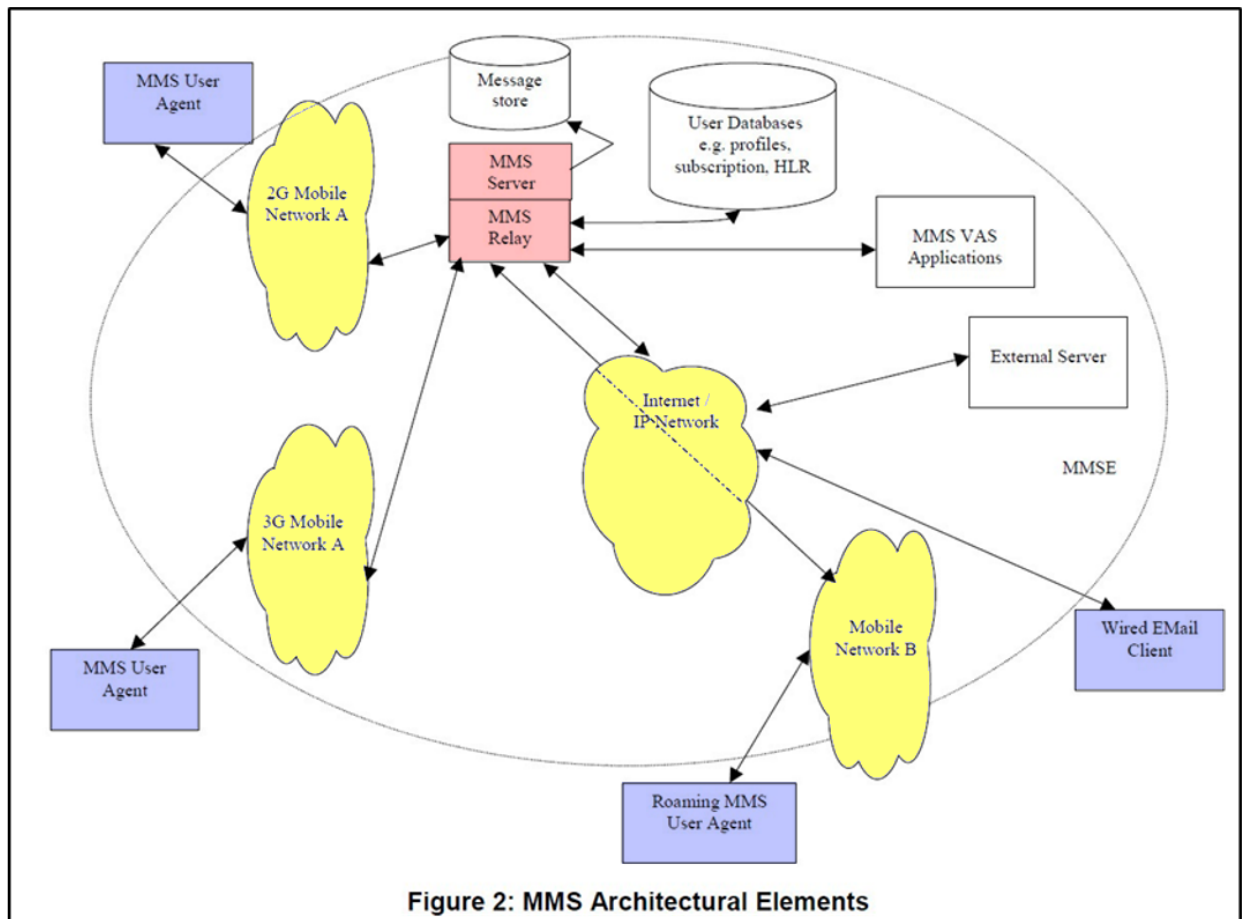
confirms that the entity providing an interprocess communication service is distinct from the entities that use it. *Id.*

Accordingly, the claimed “secure interprocess communication service” is not satisfied by the secure delivery of a message between two processes. It requires a specific service on which the message is delivered, and that service must itself be secure. Ex. 2008, ¶51. As explained in Sections VII and VIII.D below, the Petition identifies neither.

V. The Petition’s Grounds 1A–1G all fail under the Petition’s theory.

A. Overview of TS-23.140.

By way of background, the Petition specifically relies on the MMS Relay/Server as constituting the claimed “network server system.” Pet. 6-7. The Petition’s annotation of Fig. 2 (highlighting the MMS Relay/Server) of TS-23.140 is below.



Pet. 8 (Petition’s annotation of Ex. 1004 at 17, FIG. 2).

As shown in the above diagram, the MMS Relay/Server can receive messages from various entities, including MMS VAS Applications and an External Server. The MMS Relay/Server also sends and receives messages from various MMS User Agents, which as the Petition acknowledges, are the claimed “device link agents” “executing on user equipment (“UE”)/mobile devices.” Pet. 7.

The TS-23.140 specification defines certain roles and responsibilities for certain of these components. For example, the reference states that the “MMS

Relay/Server shall provide” specified functionalities including “receiving and sending MM [multimedia messages].”

The MMS Relay/Server shall provide the following functionalities:

- receiving and sending MM;
- conversion of messages arriving at the recipient MMS Relay/Server from legacy messaging systems to MM format (e.g. facsimile to MM) if interworking with legacy messaging systems (MM3) is supported;
- conversion of MMs leaving the originator MMS Relay/Server to legacy messaging systems to the appropriate message format (e.g. MM to internet email) if interworking with legacy messaging systems (MM3) is supported;
- message content retrieval;
- MM notification to the MMS User Agent;

Ex. 1004, 21; Ex. 2008, ¶¶53–56.

Although as shown above the MMS Relay/Server is the entity responsible for physically “sending [the] MM” to the MMS User Agent, TS-23.140 elsewhere specifies that it is actually the MMS User Agent that is tasked with *initiating* “MM delivery to the MMS User Agent.” *Id.*, 19 (“The MMS User Agent *shall*¹ provide the following application layer functionalities: the retrieval of MMs (*initiate MM delivery to the MMS User Agent*):”

5.1.1 MMS User Agent operations

The MMS User Agent shall provide the following application layer functionalities:-

- the retrieval of MMs (initiate MM delivery to the MMS User Agent);
- terminal capability negotiation.

¹ A POSITA would understand the use of the word “shall” to disclose a mandatory requirement of an implementation of the specification, as opposed to “may,” which refers to something that may be optionally implemented. Ex. 2008, ¶55.

Id., 19; Ex. 2008, ¶56.

The TS-23.140 specification provides additional context regarding how the MMS User Agent initiates MM delivery to the MMS User Agent. Specifically, it states that whenever an MMS Relay/Server receives a MM, it “shall generate a notification to the recipient MMS User Agent.” Ex. 1004, 28 (“Upon reception of an MM the recipient MMS Relay/Server... shall generate a notification to the recipient MMS User Agent.”). Then, “[i]n a response to the notification the MMS User Agent” has two options—either reject the MM, or retrieve the MM:

In a response to the notification the MMS User Agent shall be able to

- reject the MM or
- retrieve the MM, either immediately or at a later time (i.e. defer the MM retrieval), either manually or automatically, as possibly determined by the operator configuration and user profile.

Ex. 1004, 28; Ex. 2008, ¶57.

TS-23.140 also makes clear that the retrieval occurs in specific response to a retrieval request from the MMS User Agent. *See* Ex. 1004, 29 (“The recipient MMS User Agent shall be able to request retrieval of an MM from the recipient MMS Relay/Server based on the Message Reference received in a notification.”).

When the MMS User Agent provides a valid retrieval request (i.e., a retrieval request conforming to the TS-23.140’s requirements, including being within the message expiry period and having an appropriate size restriction), the MMS Relay/Server must initiate delivery of the MM to the MMS User Agent:

7.1.3 Retrieval of a Multimedia Message in the recipient MMSE

The recipient MMS User Agent shall be able to request retrieval of an MM from the recipient MMS Relay/Server based on the Message Reference received in a notification. If MMBoxes are supported, the MMS User Agent shall be able to request retrieval of an MM from the user's MMBox, based on a Message Reference received from a previous MMBox operation.

Within a retrieval request the recipient MMS User Agent may indicate a size restriction of the returned MM (i.e., maximum size) that the MMS Relay/Server is to use in processing the retrieval request.

Upon retrieval request the recipient MMS Relay/Server

- shall deliver the MM to the recipient MMS User Agent

Ex. 1004, 29 (annotated); Ex. 2008, ¶58.

The *only* exception to this requirement—that the “MMS Relay/Server shall deliver the MM” “[u]pon [a received] retrieval request”—is that the “MMS Relay/Server... should *not* deliver the MM... to the recipient MMS User Agent unless the size restriction set by the MMS User Agent is met.” Ex. 1004, 29. In other words, so long as a retrieval request does not specify a size restriction inconsistent with the requested non-expired message, a retrieval request *shall* initiate delivery of the message to the MMS User Agent. And this is the *only* condition that TS-23.140 discloses that “shall” initiate delivery. Ex. 2008, ¶¶59-60.

B. The Petition fails to show that TS-23.140 discloses the claim limitations, even under the Petition's claim interpretation of “message delivery trigger.”

All challenged claims include, *inter alia*, two requirements concerning message delivery triggers. First, the claims require that the network server system have “logic to determine when one of a plurality of message delivery triggers... has occurred.” Ex. 1001, claim 1[d3]. Second, the claims require that “for at least

some of the received network element messages, the receipt of such a message by the message buffer system is *not* a message delivery trigger.” Ex. 1001, claim 1[d3].

As noted above, TS-23.140 discloses only one condition under which an MMS Relay/Server “shall” initiate delivery of a message. Thus, in order to arrive at the first of these requirements (which requires not just a single “message delivery trigger,” but a “plurality of message delivery triggers”), the Petition and Dr. Traynor apply an extraordinarily broad interpretation of “message delivery trigger.”²

Specifically, the Petition makes clear that it believes that a trigger constitutes *any* “condition controlling whether an attempted message delivery is made.” Pet. 21; Ex. 1003, ¶601 (noting that “a ‘trigger’ can include conditions that” act as “any traffic light on the way to a destination,” as well as “conditions imposed on the messages controlling whether a message will be delivered at all”). And Petitioner’s

² Patent Owner does not agree that the Petition’s interpretation is the correct interpretation. *See* IPR2024-00010, Paper No. 19 at 5. However, for purposes of this response, Patent Owner exclusively responds to the invalidity theory presented by Petitioners and Dr. Traynor under their interpretation of “message delivery trigger,” because the Petition fails even under that interpretation.

expert Dr. Traynor, in a prior proceeding, made clear that if “a message cannot be [delivered] without” a first condition occurring, then that condition is a message delivery trigger. *See* Ex. 2012 at ¶¶51–58 (noting that any event which “initiates a message delivery sequence culminating with the delivery of the message” will “constitute[] a message delivery trigger”). Dr. Traynor opined that because “a message cannot be [delivered] without first receiving the MM notification and without the corresponding inclusion of the message reference,” the MM notification constitutes a “message delivery trigger.” *Id.* at ¶¶53, 58 (explaining that a “message notification[]” is a message delivery trigger even though it occurs “before the message retrieval requests”); *id.* at 34 (heading F, disagreeing with any distinction between a “message delivery trigger” and “a condition that must occur before message is delivered”) (cleaned up).

However, for *every* received network element message in TS-23.140, receipt of the message by the memory store/MMBox is necessary for that message to be ultimately delivered to an MMS User Agent. Ex. 2008, ¶¶61, 64. Thus, receipt of the message is a “condition controlling whether an attempted message delivery is made” (Pet. 21), because without such receipt, no attempted message delivery will be made in the context of TS-23.140. Consistent with the understanding of Petitioners and their expert, receipt of a message by the MMS Relay/Server is a “condition controlling whether an attempted message delivery is made” (Pet. 21), it

is a “traffic light on the way to a destination” of eventual message delivery (*see* Ex. 1003, ¶601), and it is “a condition that must occur before [the] message is delivered.” Ex. 2012 at 34; *id.* ¶53 (noting that receipt of a MM notification is a message delivery trigger because “a message cannot be retrieved without first receiving the MM notification”).

Thus, because for *every* received network element message in TS-23.140, receipt of the message by the memory store/MMBox constitutes a “message delivery trigger” under Petitioner’s own proffered interpretation of that term, the Petition fails to set forth any plausible theory under which TS-23.140 discloses this limitation. Indeed, Petitioners do not even allege that “for at least some received network element messages, message receipt by memory store/MMBox is not a message delivery trigger.”³ Instead, the Petition contends that “message receipt by

³ The Petition does contend that “*if* a trigger is a condition after which the next step is attempted delivery,” then the receipt of the MM “is not a trigger.” Pet. 21. But the Petition does not argue for this interpretation of “message delivery trigger,” and Dr. Traynor expressly rejected this interpretation at his deposition. *See* Ex. 2010 at 7:20–9:14 (disagreeing that “the claimed trigger must be a condition after which the next step is an attempt to deliver the message,” because “the plain and ordinary meaning of ‘trigger’ encompasses” everything discussed in paragraph 200 of Dr.

memory store/MMBox *alone* is not a trigger.” Pet. 20 (emphasis added). But the claim does not require that “message receipt *alone* is not a message delivery trigger” for some messages—it requires that “message receipt” with no qualification “is not a message delivery trigger.” The claim cannot be satisfied if there are not some received messages for which “message receipt is not a message delivery trigger,” even if there are some messages for which message receipt is not the *only* trigger. Ex. 2008, ¶65.

Under the plain claim language as applied to Petitioners’ proffered interpretation of “message delivery trigger,” TS-23.140 plainly does not disclose this requirement, because for every received message, message receipt is “a condition controlling whether an attempted message delivery is made,” is one of the first “traffic light[s] on the way to [the] destination” of message delivery (Ex. 1003, ¶601), and it is a “condition that must occur before [the] message is delivered.” (Ex. 2012 at 34). *See* Ex. 2008, ¶¶64-65.

Indeed, the only form of message receipt that the Petition contends would not constitute a “message delivery trigger” under its own proposed, broad interpretation of that term, is the “receipt of an MM with a specified delivery-

Traynor’s declaration, Exhibit 1003, which is substantively identical to paragraph 601); Ex. 2008, ¶¶46, 65.

time.” Pet. 21. As an initial matter, the Petition misunderstands TS-23.140; no received messages specify a particular delivery time, but instead specify an *earliest* delivery time after which the message may be forwarded to another Relay/Server such that a notification may be provided to the User Agent, in response to which the User Agent may potentially request delivery of the message. Ex. 2008, ¶66; *see also* Ex. 2012 at ¶52 (“[W]hen the MMS Relay/Server receives a message, it generates and sends a notification [when the message is ready for delivery] to a MMS User Agent that includes a message reference corresponding to the message.... [A] message cannot be retrieved without... the message reference.”) (citing Ex. 1004, 28–29). And even *if* a message could be delivered immediately upon the specified earliest desired delivery time being reached, the initial reception of the message is still a “condition controlling whether an attempted message delivery is made” (Pet. 21), it is a “traffic light on the way to a destination” of eventual message delivery (*see* Ex. 1003, ¶601), and it is “a condition that must occur before [the] message is delivered.” Ex. 2012 at 34; *id.* ¶53 (noting that receipt of a MM notification is a message delivery trigger because “a message cannot be retrieved without first receiving the MM notification”). All of these are ways in which Petitioners and their expert have expressly alleged the term “message delivery trigger” can be satisfied, such that receipt of a message—even if the message specifies a delivery time—would constitute a “message delivery

trigger” under Petitioners’ chosen interpretation of that term.

VI. The Petition’s Ground 1 additionally fails regarding claims 3 and 4.

The Petition challenges claim 3 under both Ground 1A (involving TS-23.140 alone) and Ground 1B (TS-23.140 in combination with Adamczyk). As discussed below, both of these grounds fail to establish the obviousness of claim 3. And because claim 4 depends from claim 3, the Petition likewise fails to show the obviousness of claim 4 under Ground 1C.

A. The Petition’s Ground 1A (TS-23.140 alone) fails with respect to claim 3.

The Petition alleges that TS-23.140 discloses or renders obvious claim 3 under two theories: an “earliest desired delivery-time” and “periodic polling.” Pet. 26-27. As discussed in the following subsections, both of these theories fail.

1. The Petition’s “earliest desired time of delivery” theory fails.

The Petition notes that an “earliest desired time of delivery” can be specified by an originating MMS User Agent, and then contends that “POSITAs would have understood/found obvious specifying delivery time is a periodic-timer trigger because the timer is triggered upon message receipt and message delivery is triggered at expiration after a particular period.” Pet. 26-27; Ex. 2008, ¶84.

Even accepting this characterization of the Petition, however, the reaching of an “earliest desired time of delivery” is not an “expiration of a *periodic* timer.” The Petition’s sole explanation as to why an earliest delivery time satisfies a “periodic timer” is because “each time a message is received, the timer is triggered and continues for a particular period.” Ex. 1003, ¶641. But as discussed above in Section IV, a “periodic timer” is a timer that recurs at regular intervals, rather than just a timer “that continues for a particular period” as the Petition asserts. See Ex. 2008, ¶85.

The Petition sets forth no explanation why TS-23.140’s “earliest desired time of delivery” would be in accordance with “expiration of a *periodic* timer,” as opposed to just “a timer.” For instance, the Petition presents no evidence that any “earliest desired time of delivery” timer in TS-23.140 would recur at regular intervals. Ex. 2008, ¶85. Notably, this is in sharp contrast to the Petition’s analysis for TS-23.140’s “periodic polling” theory discussed below in Section VI.A.2, where the Petition specifically alleged that “periodic polling is a periodic-timer trigger *because polling happens at regularly-repeating intervals.*” *Pet. 27.*

In sum, the Petition fails to establish that the “earliest desired time of delivery” is related to the expiration of a periodic timer (in addition to failing to establish that the earliest desired time of delivery is a “message delivery trigger” for delivery of a message to an MMS User Agent). Additionally, although the

Petition alleges that “POSITAs would have understood/found obvious specifying delivery time is a periodic-timer trigger,” the Petition sets forth no plausible obviousness theory. See Pet. 26-27. For example, the Petition sets forth no motivation to modify the disclosures of TS-23.140 to satisfy the claim requirements, nor does it even explain what any modifications would have entailed. Ex. 2008, ¶¶85-86.

2. The Petition’s “periodic polling” theory also fails.

The Petition contends that the requirement of claim 3 is satisfied because “‘periodic polling’ involves User Agent retrieving messages from an external server via Relay/Server.” Pet. 27. Petitioners’ expert Dr. Traynor acknowledged that he did not provide any opinions in his declaration about how this would be achieved, other than via a “message retrieval request.” Ex. 2010 at 18:17–19:2. However, there is no teaching in TS-23.140 that the MMS User Agent engages in the alleged periodic polling, and that understanding would be inconsistent with TS-23.140’s teachings. Ex. 2008, ¶¶92–100.

Specifically, the Petition relies upon Section 8.3.3 of TS-23.140, which states that “[f]or discovery of incoming messages from external messaging systems different mechanisms may be utilized, e.g.,... periodic polling for messages on External Server, followed by retrieval by the MMS User Agent via the MMS Relay/Server.” Ex. 1004, 90-91.

Notably, the above teaching of TS-23.140 does **not** say (as the Petition asserts) that the ***MMS User Agent*** engages in periodic polling. Rather, it says that the periodic polling is used for “discovery of incoming messages from external messaging systems.” Ex. 1004, 90-91. A POSITA would understand that it is the MMS Relay/Server’s role (not the MMS User Agent’s role) to interact with external messaging systems (i.e., “External Servers”). See Ex. 1004, 22 (“*Convergence functionality between External Servers and MMS User Agents is provided by the MMS Relay/Server* which enables the integration of different server types across different networks.”); Ex. 2008, ¶¶93–95.

The fact that it is the MMS Relay/Server, **not** the MMS User Agent, that coordinates with the external messaging system is further shown by FIGS. 2 and 4 of TS-23.140, both of which show the MMS User Agent separated from the external servers by MMS Relay/Server, and the MMS User Agent having no direct communication path to an External Server:

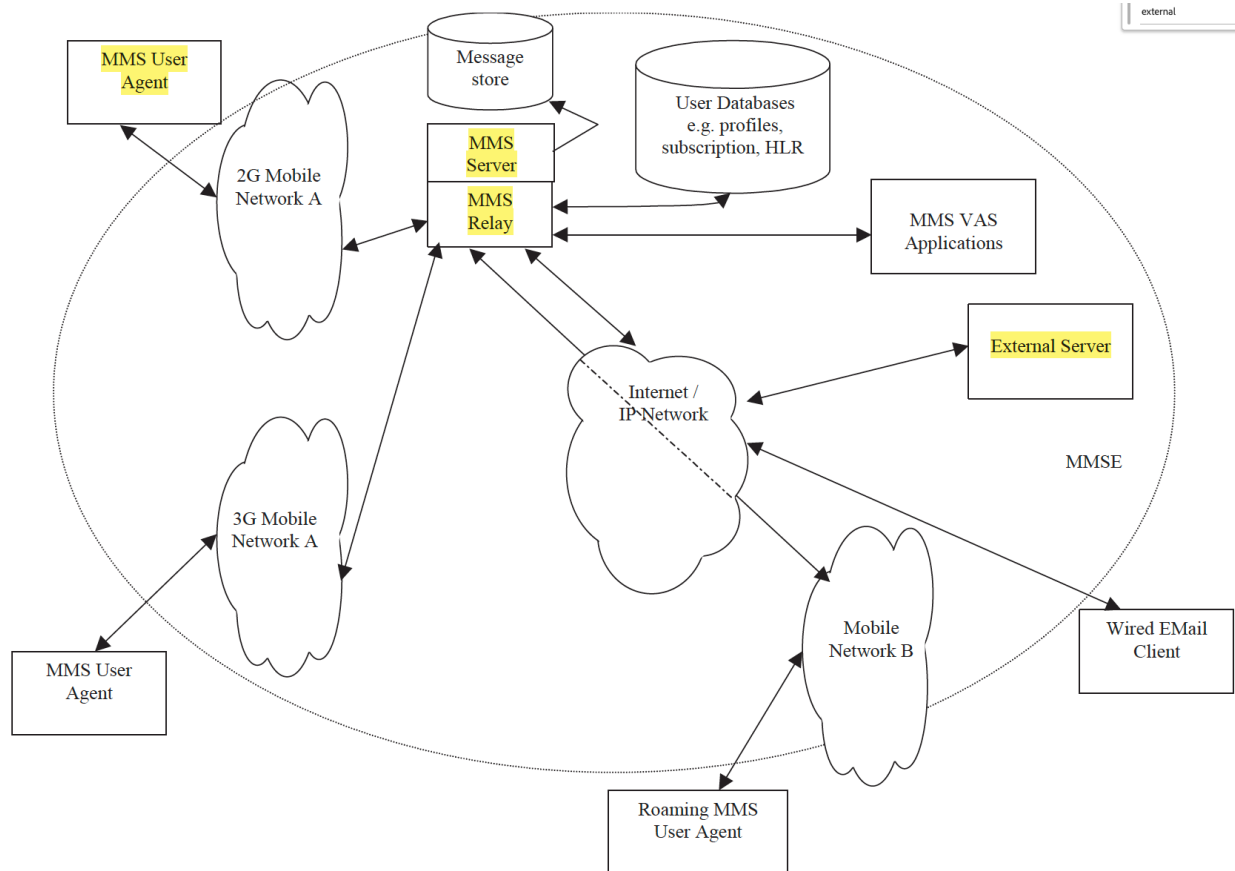


Figure 2: MMS Architectural Elements

Ex. 1004, 17 (annotated); Ex. 2008, ¶95.

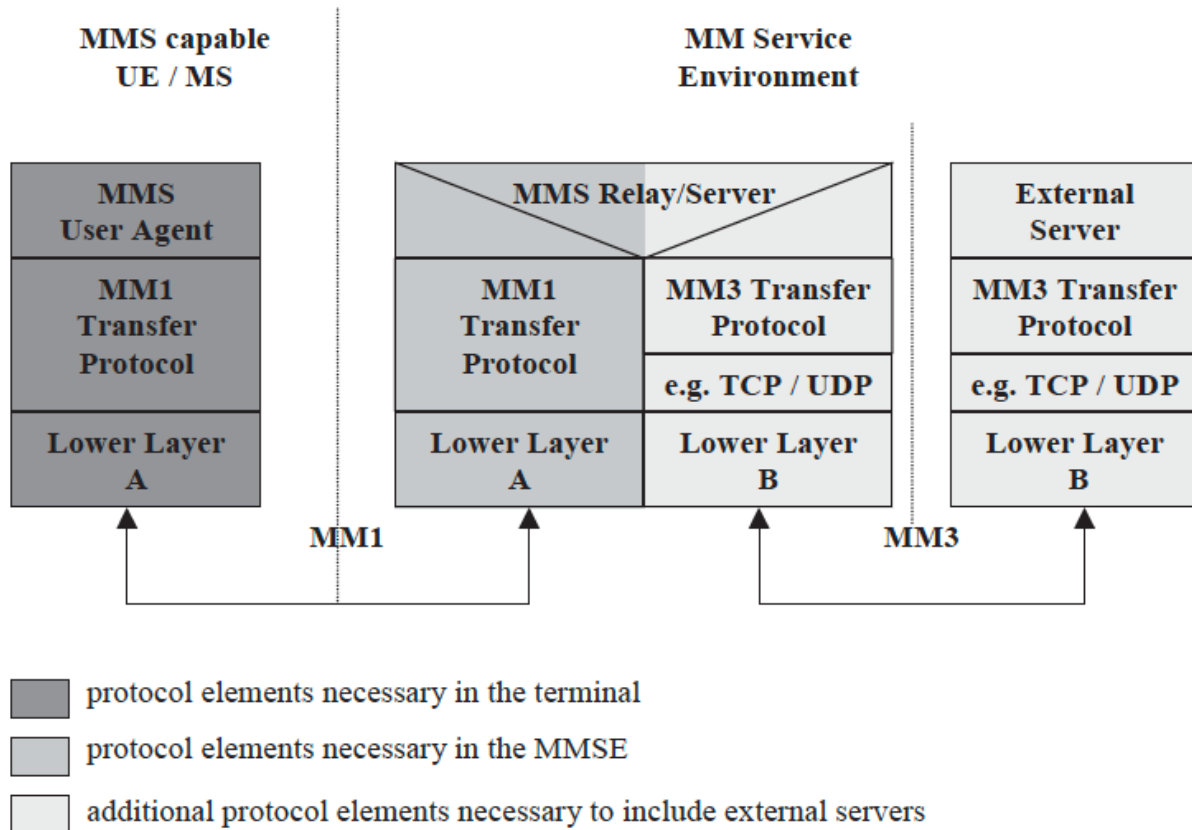


Figure 4: Protocol Framework to provide MMS

Ex. 1004, 24.

In fact, TS-23.140 gives a specific example of how periodic polling would work with an external messaging system (such as a “voice mailbox”). Specifically, where polling is used, “the *MMS Relay/Server* may poll the voice mailbox via POP3 or IMAP4 for new messages received.” Ex. 1004, 157-158. Nowhere does this example (or any other teaching of TS-23.140) indicate that the *MMS User Agent* would periodically poll an external server, as the Petition asserts. Ex. 2008, ¶96.

Indeed, at his deposition, Dr. Traynor, when asked whether “TS 23.140 actually discloses that the MMS User Agent performs the periodic polling,” he could not answer in the affirmative, instead only stating that this would be “the scalable way to” implement periodic polling. Ex. 2010 at 33:17–34:10. In other words, Dr. Traynor’s opinion at deposition was only that it would be *obvious* for the MMS User Agent to implement periodic polling because that would be a scalable implementation. But Dr. Traynor conceded at his deposition that he did not “have any text [in his declaration] on” why it “would make the most sense for the User Agent to do the polling” in his declaration (Ex. 2010 at 35:9–21), such that this is not a theory that is supported by Dr. Traynor’s declaration.

And as discussed above, Dr. Traynor’s improper obviousness theory at his deposition is directly inconsistent with TS-23.140’s express disclosure that *the MMS Relay/Server* performs the periodic polling. Dr. Traynor’s assertion that the periodic polling (which he previously testified “would be retrieval requests”—Ex. 2011, 14:12–15:1) would be performed by the MMS User Agent makes no sense in the context of TS-23.140. *See* Ex. 2008, ¶¶98-99. Specifically, TS-23.140 makes clear that the purpose of “periodic polling” would be for “*discovery* of incoming messages.” Ex. 1004, 90-91. But retrieval requests can only occur *after* the MMS User Agent is *aware of* an incoming message. Ex. 2008, ¶98. Specifically, a retrieval request message *must* include a “Message Reference” value. *See* Ex.

1004, 71 (stating that “Message Reference” is a “Mandatory” component of “MM1_retrieve.REQ”). TS-23.140 further teaches that “[t]his reference *was previously delivered to the MMS User Agent*” through any number of messages that, as Dr. Brogioli explains, would be sent by the MMS Relay/Server. Ex. 1004, 69.

In other words, without previously being *aware of* an incoming message, it would be *impossible* for an MMS User Agent to perform the polling that Dr. Traynor alleges would have occurred. Accordingly, it would be impossible for Dr. Traynor’s alleged “periodic polling” to be used for “discovery of incoming messages from external messaging systems” as the Petition alleges. *See* Ex. 1004, 90-91; Ex. 2008, ¶¶97–98.

Furthermore, the above discussion is entirely consistent with the stated purpose of Section 8.3 of TS 23.140. Specifically, TS 23.140 states that Section 8.3 “*defines the interworking between MMS Relay/Servers and External Messaging Servers.*” Ex. 1004, 90. This further confirms that Section 8.3’s disclosing of “periodic polling” would be performed by the MMS Relay/Server, *not* the MMS User Agent as Petitioner and Dr. Traynor assert without any evidentiary support.⁴ Ex. 2008, ¶99.

⁴ The Petition’s reliance on TS-23.140’s teaching of “polling for messages on [an]

Finally, even if the MMS User Agent *did* engage in periodic polling, the Petition still fails to show it meets the claim requirements. Specifically, if “periodic polling” performed by the MMS User Agent is “one of a plurality of message delivery triggers,” then the Petition would need to show that the network server system has logic to determine whether periodic polling has occurred. But the Petition identifies no such logic (and does not allege that such logic would be obvious—*see Pet. 27*), and Dr. Traynor at his deposition was unable to identify any such logic. *See Ex. 2010 at 36:16–23* (“Q. Does TS 23.140 disclose any logic to determine whether or not a given message retrieval request was the result of periodic polling? THE WITNESS: ... I’m not aware off the top of my head.”). *See also Ex. 2008, ¶100.*

B. The Petition’s Grounds 1B and 1C (TS-23.140 in combination with Adamczyk and/or Herzog) also fail with respect to claims 3 and 4.

The Petition presents an alternative theory with respect to claim 3, based on an allegedly obvious combination of TS-23.140 and Adamczyk. However, this

External Server” fails for an additional reason: “An External Server is connected to [an] MMS Service Provider via non-MMS-specific protocols,” (Ex. 1004, 14), and the Petition points to no evidence that a message received from an external server would meet the requirements of claim 1. *Ex. 2008, ¶99.*

alternative theory also fails to establish the obviousness of claim 3. *See* Ex. 2008, ¶¶101-102, 104-109.

By way of background, the Petition's theory of how TS-23.140 would allegedly be modified in view of Adamczyk is that, consistent with what is disclosed in Adamczyk, TS-23.140's MMS Relay/Server would be "configured to send [] notification messages to the recipient on demand, at a specific future time, and/or on a periodic schedule." Pet. 33-34; *see also* Ex. 1003, ¶692 (alleging: "[A] POSITA would have been motivated to implement a periodic timer like Adamczyk's to have the MMS Relay/Server issue **notifications** at pre-determined intervals [with]⁵ all messages received during that interval."). Dr. Traynor confirmed at his deposition that his theory was **not** that the messages would be automatically delivered according to the periodic timer, but simply that in the proposed combination, the User Agent would be sent notification messages notifying the User Agent that messages are available to be retrieved, according to a periodic interval:

Q. Okay. So is it your opinion that on expiration of the periodic timer, the MMS Relay/Server would issue a notification to the User Agent that a message is available and then, in response to that

⁵ Dr. Traynor confirmed that "will" should be read as "with" in this context. Ex. 2010 at 37:24–38:10.

notification, the User Agent could request delivery of that message?

THE WITNESS: Again, on expiration of the timer at the MMS server, it -- a notification that messages were available would be sent. If the device on which the User Agent was running was available, you know, met a number of criteria and, you know, including things like had sufficient storage for the messages, it then might request to receive them.

...

Q. Right. And so my question is -- is -- I believe TS 23.140 discloses there are a number of messages exchanged before the messages actually deliver. There's a notification there's a message retrieval request and then there's the attempted delivery. *My question is, is your proposed combination with TS 23.140 changing the operation of TS 23.140 such that upon expiration of the periodic timer, the MMS Relay/Server would just transfer the application message to the User Agent without first waiting for a retrieval request?*

A. *I -- I don't believe that I'm suggesting that.* I'm saying that the expiration of a timer triggers a series of additional events, each of which could potentially stop the delivery. *So if a notification was sent to the User Agent and it didn't respond, that -- those messages would not be delivered.* So I'm just saying that there's a - a trigger that's happening at the MMS server.

Ex. 2010, 41:19–42:24.

The Petition fails to show that its proposed combination of TS-23.140 and Adamczyk would be obvious. The Petition is clear that its proposed combination would issue notifications regarding “*all* messages received during the time interval at once” (Pet. 34), which conflicts with TS-23.140’s teaching that an earliest desired delivery time can be specified. By using a periodic timer that provides notifications for all messages received during the interval period to send the notification, the resulting system would prevent the ability to set an earliest desired delivery time. Ex. 2008, ¶104.

For example, say that the MMS Relay/Server were configured to transmit notifications regarding all messages received in the preceding 60 minutes at the top of every hour. But for a given message transmitted at 1:30 PM with an earliest specified delivery time of 10:30 PM, the Petition’s proposed combination would issue a notification message at 2 PM according to the periodic schedule, thus violating TS-23.140’s teachings regarding earliest desired delivery time. Ex. 2008, ¶104.

And for messages *without* a specified earliest desired time of delivery, the Petition’s proposed combination would have the clear drawback of unnecessarily delaying the notifications regarding those messages. Nor is this drawback outweighed by any benefits. The only benefits the Petition even alleges are “(1) providing user control over when messages are received; and (2) increasing

UE/mobile device battery life by avoiding repeated pull requests.” Pet. 34. Regarding the first of these theories, the Petition fails to explain *why* a user would have allegedly found it beneficial to specify notification messages to be delivered periodically, thus either (1) delaying their arrival if no earliest delivery time is specified, or (2) violating the earliest desired delivery time. Dr. Traynor provides no additional detail in this regard, merely alleging that this would result in a “more configurable system” but failing to allege why this additional configurability would be beneficial in any way. *See* Ex. 1003, ¶692. And the resulting system would be even *less configurable* overall, because it would prevent the ability to effectively specify an earliest desired time of delivery as discussed above. *See* Ex. 2008, ¶105.

Regarding the Petition’s second alleged benefit, “increasing UE/mobile device battery life by avoiding repeated pull requests,” the Petition fails to set forth any competent evidence that a POSITA would have identified any such benefit. Instead, as is clear from Dr. Traynor’s declaration, this alleged benefit is based on the User Agent issuing “repeated pull requests” in TS-23.140 as it exists, such that it would “repeatedly ping the MMS Relay/Server to see if messages are pending.” Ex. 1003, ¶693. But as discussed above in Section VI.A.2, TS-23.140 *does not disclose* “repeatedly ping[ing] the MMS Relay/Server to see if messages are pending”; this is Petitioners’ and Dr. Traynor’s own misunderstanding of how TS-

23.140 would operate. As discussed in Section VI.A.2, the MMS Relay/Server (not the MMS User Agent) may periodically poll other servers to check for messages. Ex. 2008, ¶106.

Indeed, Dr. Traynor’s testimony does not support the Petition’s allegation that the proposed combination would increase battery life. Dr. Traynor’s declaration asserts only that this combination would “*arguably* improve the battery life,” not that a POSITA would believe it would *actually* improve battery life. And as noted above, the Petition’s theory sets forth no plausible theory for battery life improvement at all because the Petition’s theory is based on a misunderstanding or mischaracterization of TS-23.140’s operation. Ex. 2008, ¶¶107–108.

Furthermore, even if the proposal *might* “arguably” save some unspecified amount of battery life (a proposition for which the Petition and Dr. Traynor present no evidence), there is no reason to believe that a POSITA would have thought the *certain* drawback of delayed message delivery would justify what even Petitioners’ expert at most contends is an “arguabl[e]” benefit of some unspecified battery savings. Ex. 2008, ¶108.

The Petition also fails to establish a reasonable expectation of success. As explained above, periodic delivery of message notifications is inconsistent with TS-23.140’s description of how it will handle the “earliest desired time of delivery” operation (*see* Ex. 1004, 27), and the Petition articulates no way to

reconcile its proposal with the requirements of the MMS standard where the earliest-desired-delivery-time feature is supported. *See* Ex. 1004, 27 (“Upon reception of an MM... the originator MMS Relay/Server... ***shall*** retain the MM until the earliest desired time of delivery.... *If this feature is not supported then ***the MM is immediately routed forward.****”); Ex. 2008, ¶109.

The Petition’s Ground 1C similarly fails with respect to claim 4, because that Ground relies on the Ground 1A and Ground 1B theories to arrive at the requirements of claim 3, from which claim 4 depends. *See* Pet. 35.

VII. The Petition’s Ground 1 additionally fails to establish the obviousness of claim 17.

The Petition challenges claim 17 under Ground 1G, contending that the claim would have been obvious over TS-23.140 in combination with Fok. Pet. 47-48. That ground fails for two independent reasons. First, the combination the Petition proposes is not what Fok teaches. Fok discloses applications that communicate directly with one another through a single handshake, and the Petition instead routes every such communication through the MMS User Agent, a departure the Petition never justifies. Second, even if the proposed combination were obvious, the Petition does not show that it satisfies claim 17, which requires an “interprocess communication service” that is itself “secure.” Either failure is independently fatal to Ground 1G. Ex. 2008, ¶¶110-124.

A. The Petition fails to establish the obviousness of its proposed combination of TS-23.140 with Fok.

The Petition proposes to modify TS-23.140 “based on Fok’s teachings to provide ‘secure inter-application communication’ for two or more applications residing on UE/wireless device to transmit application-specific MMs to one another through User Agent.” Pet. 47-48. In other words, the Petition’s combination requires two communications and handshakes (between an originating application and the MMS User Agent, and the MMS User Agent and a recipient application). That is not what Fok teaches.

In Fok, a “primary” application communicates *directly* with one or more “recipient” applications through a *single* handshake. Ex. 1038, [0047]-[0053]. Neither Fok nor TS-23.140 discloses the two-handshake arrangement the Petition proposes, in which a transmission between two applications on the same device is routed through the MMS User Agent as an intermediary. Ex. 2008, ¶110. Dr. Traynor acknowledges the departure, explaining that “although Fok discloses a direct handshake between the two applications,” his combination “would instead pass through the User Agent (so one application sends the User Agent a message, which the User Agent then forwards to the receiving application).” Ex. 1003, ¶758. The Petition therefore does not propose to *implement* Fok’s teachings in TS-23.140; it proposes to *depart* from them, and the reasons the Petition offers do not

establish the obviousness of this departure. Ex. 2008, ¶111. An obviousness case resting on a modification that the Petition never explains cannot be sustained by “mere conclusory statements”; it requires “some articulated reasoning with some rational underpinning.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007).

Notably, the question is **not** whether a POSITA would have been motivated to allow secure communication between applications on a single device, because Fok’s disclosure as applied to TS-23.140 *without modification* would already allow for that result by providing a technique whereby two applications that wish to communicate with each other may do so directly through a handshake process. Ex. 2008, ¶¶110, 115.

The Petition’s stated motivation is that “a device may have more than one application installed capable of consuming MM data, and secure inter-application communication would have allowed, e.g., a first application receiving an MM to ensure a second application was trusted before allowing the second application to receive its data.” Pet. 48. Fok, however, allows applications to ensure other applications are trusted through a direct handshake between the applications (Ex. 1038, [0047]-[0053]), *without* the need for an inefficient intermediary as the Petition proposes. *See* Ex. 2008, ¶¶112–118.

The Petition’s own proposal confirms that no intermediary is needed. Under that proposal a sending application must already identify the intended recipient

application, because the message must carry an application identifier consistent with TS-23.140's disclosure of transmission of abstract messages containing application data. The sending application could therefore address the recipient directly using Fok's handshake as Fok describes it, without the MMS User Agent as an intermediary. Ex. 2008, ¶115. And the Petition offers no evidence that a POSITA would have found Fok's direct-communication technique deficient in the first place. *Id.*

Finally, the Petition's proposed deviation from Fok's teachings would have obvious disadvantages. For instance, if application A wanted to communicate with application B, instead of only a single handshake between applications A and B, application A and the User Agent would need to conduct a handshake, and the User Agent and application B would need to conduct a handshake. This would double the number of handshakes and communications required for applications A and B to communicate with one other, a disadvantage that a POSITA would readily recognize because it would delay transmissions and increase processing power and battery drain. *See* Ex. 2008, ¶117.

Furthermore, the Petition's deviation from Fok's teachings would restrict secure interprocess communication to "MMS-compliant applications." Ex. 1003, ¶757. A POSITA who saw value in secure on-device communication in the first place would have had no reason to design a solution that excludes applications that

are not MMS-compliant, including the “photo-storage or editing application” that Dr. Traynor offers as his own example. *Id.*; Ex. 2008, ¶118. Implementing Fok’s teachings as Fok describes them (i.e., allowing for direct communication between sending and recipient applications rather than using a third component as an intermediary) would permit that communication, and would do so with less signaling overhead. Ex. 2008, ¶118. Because the Petition never explains why a POSITA would have departed from Fok’s teaching of direct communication between applications, it fails to establish that its proposed combination would have been obvious. Ex. 2008, ¶¶110-118.

B. The Petition fails to show that its proposed combination, even if obvious, would satisfy the claim language.

Claim 17 requires that at least a given one of the devices comprise “a secure interprocess communication service separately secured from the secure Internet data message link,” and that the device link agent for that device cause “messages to be securely delivered to a software process by initiating delivery of each such message on the secure interprocess communication service.” Ex. 1001, claim 17. As explained in Section IV.B above, that language imposes two requirements: the messages must be delivered on an identifiable “service,” and that service must itself be “secure.” Ex. 2008, ¶119. The Petition establishes neither.

The Petition equates the claimed service with Fok's "secure inter-application communication." Pet. 47-48. But Fok's secure inter-application communication is a handshake protocol that the communicating applications implement and coordinate between themselves, not a service provided to those applications by some other component. Ex. 1038, [0047]-[0053]; Ex. 2008, ¶122. Beyond that label, neither the Petition nor Dr. Traynor identifies anything in TS-23.140 or in Fok that constitutes the claimed service. Pet. 47-48; Ex. 1003, ¶¶754-760 (using the word "service" only in restating the claim language); Ex. 2008, ¶120. As best can be discerned, the Petition's position is that the secure delivery of a message to a software process inherently supplies a secure interprocess communication service. Ex. 1003, ¶754. But interprocess communication need not involve an interprocess communication service, any more than two individuals speaking with one another must be using a communication service such as telephone or mail service. Ex. 2008, ¶120; *see also id.*, ¶¶48-51.

Even accepting the Petition's label, claim 17 requires that the service itself be "secure," and the Petition does not show that. Ex. 2008, ¶121. Fok secures communication through a handshake between the communicating applications, which is a security mechanism that each application implements and that the applications coordinate between themselves. Ex. 1038, [0047]-[0053]; Ex. 2008, ¶122. Fok in fact proceeds from the premise that whatever mechanism carries the

communication is not itself secure, which is why the applications must establish trust through the handshake. Ex. 2008, ¶121. As explained by Dr. Brogioli, an unsecured public Wi-Fi hotspot provides a helpful analogy. The unsecured hotspot provides an unsecured communication service between a laptop and the Internet. Messages sent over that service may nevertheless be encrypted and thereby delivered securely, but that encryption does not convert the hotspot into a secure communication service. Ex. 2008, ¶122. Security of that kind comes not from the communication service but from mechanisms that the endpoints apply to the messages they exchange. *Id.*

The same is true of the Petition’s proposed combination. Whatever security that combination provides comes from the handshaking mechanisms that the applications implement, not from any service on which the messages are delivered. Ex. 2008, ¶123. That does not satisfy claim 17, which requires that the interprocess communication service itself be secure rather than that the processes communicating over it supply their own security. Ex. 2008, ¶¶119, 123. The Petition thus neither identifies an “interprocess communication service” nor shows that any such service is “secure.” Ground 1G fails for that reason as well. Ex. 2008, ¶124.

VIII. The Petition’s Ground 2 fails with respect to all challenged claims.

A. The Petition sets forth no plausible theory as to how “for at least some of the received network element messages, the receipt of such a message... is not a message delivery trigger.”

As noted above, all challenged claims require that “for at least some of the received network element messages, the receipt of such a message by the message buffer system is *not* a message delivery trigger.” Ex. 1001, claim 1[d3]. However, the Petition does not provide any plausible allegation (or indeed any coherent analysis) regarding why, in the Petition’s proposed combination of Houghton with Munson, receipt of any claimed network element messages would *not* constitute a “message delivery trigger.” See Ex. 2008, ¶¶125–127. The entirety of the Petition’s discussion of this aspect of the claims is reproduced in its entirety below:

Houghton-Munson teaches 1[d3]. Traynor, ¶¶841-53.

Houghton’s “push message from the push server 401 may be triggered by any trigger event, local or remote, defined at the server,” and triggers can “include an alarm, notification, or measurement result received to the push server 401 from another device or system” (*message-delivery triggers [in which] the receipt of a message is not a message delivery trigger*). Houghton, 21, 14; Traynor, ¶¶842-44.

...

Houghton-Munson teaches a *buffer* with *logic* to determine when a *trigger* has occurred—including triggers that are not message receipt—including asynchronous content push. Traynor, ¶¶841-53.

Pet. 68–70 (italics and alterations in original). As shown above, the Petition does not dispute that triggers *do* include message receipt for every received message, alleging only that the Petition’s proposed combination *also* contemplates *additional* “triggers that are not message receipt.” Pet. 70. This does not satisfy the claims, which require “for at least some of the received network element messages, *the receipt of such a message* by the message buffer system *is not a message delivery trigger*.” This claim language plainly cannot be satisfied if for all received messages, “the receipt of such a message” *is* “a message delivery trigger,” even though there might be *additional* message delivery triggers associated with that received message. But that is precisely how the Petition’s theory rewrites the claims. Ex. 2008, ¶¶125-126.

The Petition’s rewriting of the claim requirement is, indeed, *explicit* within the Petition. For instance, in the excerpt reproduced above, the Petition alleges the claim requirement to be “message-delivery triggers [in which] the receipt of a message is not a message delivery trigger,” and contends only that *that* made-up requirement is satisfied. See Pet. 68.

The Petition nowhere alleges that its Houghton/Munson combination would

disclose “for at least some of the received network element messages, the receipt of such a message... is not a message delivery trigger.”

Nor could the Petition have done so, consistently with the Petition’s own invalidity theory (and the understanding of Petitioners’ expert) under which a “message delivery trigger” can be satisfied by any of a “condition controlling whether an attempted message delivery is made” (Pet. 21), a “traffic light on the way to a destination” of eventual message delivery (*see* Ex. 1003, ¶601), or “a condition that must occur before [the] message is delivered.” Ex. 2012 at 34. Ex. 2008, ¶127.

The Petition’s only allegation in this regard is that the proposed combination of Houghton and Munson disclose message delivery triggers such as “an alarm, notification, or measurement result,” which triggers are not themselves “receipt of a message.” See Pet. 68. Indeed, the Petition’s analysis explicitly *misreads* the claim language as requiring “message-delivery triggers [in which] the receipt of a message is not a message delivery trigger.” Ex. 2008, ¶127.

Because the Petition does not set forth any plausible theory how for any received network element messages, reception of such a message is *not* a message delivery trigger as claimed, the Petition’s Ground 2 fails.

B. The Petition fails to establish the obviousness of its proposed Houghton/Munson combination.

The Petition’s Ground 2 theory additionally fails because the Petition fails to motivate its combination of Houghton/Munson in the first place. *See* Ex. 2008, ¶¶128–135. The Petition alleges that a POSITA would have found it obvious “specifically to incorporate Munson’s *store-and-forward functionality* into Houghton’s push-messaging system.” Pet. 51. The Petition alleges that “Munson provides implementation details... for ‘store and forward messaging systems’ that Houghton already contemplates,” and alleges that the resulting combination would “yield predictable results (push-messaging systems that store and forward messages).” Pet. 51-52. As explained below, each of the Petition’s proffered reasons to combine rests on a mischaracterization of Houghton or on generic benefits untethered to Houghton’s stated design goals, and its “predictable results” formulation merely restates the proposed combination (a push-messaging system that stores and forwards messages) as if it were the *reason* for *creating* that combination in the first place. That is precisely the sort of conclusory reasoning that cannot sustain an obviousness challenge. *See KSR*, 550 U.S. at 418.

What the Petition neglects to mention is that, while Houghton *does* “contemplate[]” “store and forward messaging systems,” Houghton is critical of such systems. The Petition’s citations to Houghton “contemplat[ing]” “store-and-

forward messaging systems” (Ex. 1007, 3, 7—see Pet. 52) are telling. For example, Houghton’s discussion of “store and forward messaging systems” at page 3 explains:

SMS and MMS (Multimedia Messaging Service) are the most common forms of "push" messaging in the mobile industry. *These store and forward messaging systems* run by mobile telecommunications operators *push all or part of the data to the mobile terminal over closed telecommunications signaling networks* such as SS7 (Signaling System Seven). These signaling networks are closed; they are not part of the Internet. *Each operator controls and encrypts this traffic, charging high prices for their use and limiting their use and usefulness.*

Ex. 1007, 3.

Houghton goes on to explicitly contrast its own invention with the problematic structure of store-and-forward messaging systems:

IP push messages utilizing the principles of the invention are delivered rapidly, bypassing the public and often crowded SMS, MMS, WAP Push and email store-and-forward mechanism within an operator and between operators. This guarantee of very rapid event push provides usability benefits to users which may be then designed into the mobile applications. For example the ability to expect prompt delivery allows modification to interactive mobile application system design. The mobile process surrounding the application can be changed such that a user on one mobile terminal

may take actions which result in the initiation of a push event to a user on a second terminal and then wait for a short time for the second user to respond after the launching or commanding to return to be the focus of user interface attention their copy of the same, similar or complimentary mobile application function involving Internet communication directly or mediated by a server between the two mobile applications.

Ex. 1007, 13; Ex. 2008, ¶¶129-130.

As shown above, Houghton describes store-and-forward messaging systems as having limited “use and usefulness,” describes their delivery mechanisms as “often crowded,” and touts as the defining advantage of its own invention that its IP push messages are “delivered rapidly, bypassing” those very mechanisms—a “guarantee of very rapid event push” that provides “usability benefits to users which may be then designed into the mobile applications.” Ex. 1007, 3, 13. Houghton thus does not merely express a preference for a different transport network; it criticizes and discourages the store-and-forward delivery model itself, because that model defeats the prompt, deterministic delivery that Houghton identifies as its central benefit. Ex. 2008, ¶131.

To be clear, Patent Owner does not contend that a base reference must be defective before a POSITA would consult other art, nor that Houghton forbids any storage of messages—Houghton itself retains “lazy or low priority messages” for

later, cost-efficient transmission when a higher-bandwidth connection becomes available. Ex. 1007, 33-34 (claim 24). The point is that the Petition proposes to interpose Munson’s store-and-forward delivery model—by default holding received messages at the push server and delivering them only upon later conditions—into the delivery path of the time-critical push messaging that is Houghton’s reason for being. In Houghton, a “trigger event” is what causes the push server to generate and send a push message in the first place (or in the case of “low priority messages,” can be a more efficient network becoming available). Ex. 1007, 21; Ex. 2008, ¶132. Munson’s model inverts that sequence: messages that have already been received, by default, sit in storage awaiting some later delivery condition. Grafting that model onto Houghton would delay exactly the messages Houghton promises will be “delivered rapidly,” and the Petition’s only stated justification is that the result would be a “push-messaging system[] that store[s] and forward[s] messages.” Pet. 52. That is not a *reason* to make the change; instead, it is merely a restatement of the change itself, and it supplies no “articulated reasoning” why a POSITA designing Houghton’s rapid-push system would have made a modification that runs counter to Houghton’s guarantee of rapid delivery. *KSR*, 550 U.S. at 418; Ex. 2008, ¶132.

Nor do the Petition’s two alleged benefits supply the articulated reasoning that its “predictable results” rationale lacks. The Petition contends that there are

two alleged benefits to transforming Houghton into the store-and-forward systems that it criticizes: “(1) ensur[ing] messages are not lost if a recipient’s mobile terminal is unreachable (e.g. if not connected to the network at transmission time)” and “(2) provid[ing] users greater options to control delivery times/conditions.” Pet. 51-52; Ex. 2008, ¶¶133–135.

Regarding the first alleged advantage, the Petition’s premise—that in Houghton, messages would otherwise be lost “if a recipient’s mobile terminal is unreachable”—mischaracterizes Houghton. Houghton’s system maintains a “permanent IP connection” to each mobile terminal and uses standard networking techniques to “automatically re-establish the connection when the connection gets broken.” Ex. 1007, 11. And where a message should not be sent immediately, Houghton already retains it: Houghton’s push server “uses the connection availability information as a trigger event for sending lazy or low priority messages including application updates and application data updates routed through the higher bandwidth capacity device to the wireless terminal”—that is, it holds such messages until a suitable connection is available, rather than losing them. Ex. 1007, 33-34 (claim 24); Ex. 2008, ¶¶133-134. The Petition and Dr. Traynor never engage with these disclosures, never identify any scenario in which Houghton’s own architecture actually loses a message, and never offer evidence that a POSITA would have recognized message loss as a problem in Houghton’s architecture. See

Pet. 51-52; Ex. 1003, ¶763; Ex. 2008, ¶134. A motivation premised on solving a problem that the Petition has not shown to exist is not a motivation supported by “articulated reasoning with some rational underpinning” as required. *KSR*, 550 U.S. at 418.

Regarding the second alleged benefit—“provid[ing] users greater options to control delivery times/conditions”—neither the Petition nor Dr. Traynor identifies a single “option to control delivery times/conditions” in Munson that a POSITA would have viewed as beneficial in the context of Houghton’s system, much less explains why this might even allegedly be so. The supporting testimony asserts only that “by providing computer memory to store messages mid-transmission, both the originator and recipient would be able to better control the conditions of message delivery.” Ex. 1003, ¶764. Dr. Traynor’s accompanying discussion does not say which delivery “conditions” a POSITA would have wanted to “control,” for which messages, or why any such control would have been viewed as an improvement in a system whose declared value is that push messages arrive as rapidly as possible—Houghton touts the ability of its users to “expect prompt delivery,” not to defer it. Ex. 1007, 13; Ex. 2008, ¶135. Expert testimony of this kind, which discloses no underlying facts or reasoning, is entitled to little or no weight. 37 C.F.R. § 42.65(a); *see ActiveVideo Networks, Inc. v. Verizon Commc’ns, Inc.*, 694 F.3d 1312, 1327-28 (Fed. Cir. 2012) (rejecting as “generic”

and conclusory expert motivation testimony that did not explain why a POSITA would have made the specific combination). Unexplained “flexibility” or “greater options” cannot substitute for a reason to modify this system in this way.

C. The Petition’s Ground 2 additionally fails to establish the obviousness of claims 3 and 4.

1. The Petition’s Grounds 2A and 2B fail with respect to claims 3 and 4.

Claim 3 further requires that “one of the [plurality of] message delivery triggers is the expiration of a periodic timer.”

In support of claim 3, the Petition does not contend that Houghton’s “periodic message” alone causes delivery of any buffered message content; it observes only that the periodic message is “sent upon expiration of a timer” so that devices in the communication path “do not time expire the connection.” Pet. 76-77 (citing Ex. 1007, 19-20, 26). Elsewhere the Petition does label that same periodic message a “message-delivery trigger.” Pet. 79, 86. But the Petition fails to explain how such a keep-alive message would satisfy the Petition’s interpretation of “message-delivery trigger”; for example, the Petition does not explain how such a message would act as a condition or a traffic light on the way to message delivery. Indeed, even in the absence of such periodic messages, message delivery would be possible because Houghton’s system is configured to “automatically re-establish the connection when the connection gets broken” (Ex. 1007, 11), such that

Houghton's periodic messages are not conditions that "must occur" before message delivery as Petitioner's and Dr. Traynor's understanding of "message delivery trigger" entails. *See* Ex. 2012, ¶55 (discussing message delivery triggers as "a sequence of triggers that must occur before message delivery is initiated"), 34 (disagreeing with any attempt "to distinguish a message delivery trigger from a condition that must occur before [the] message is delivered") (cleaned up). *See also* Ex. 2008, ¶136.

Instead, the Petition alleges that **Munson** discloses or renders obvious that a "group push" would be "triggered by the expiration of a regularly-repeating timer" (Pet. 77), and further that it would be obvious to modify Houghton to use "a regularly-repeating timer to cause attempted message delivery in Houghton." Pet. 77; Ex. 2008, ¶137.

The Petition's reliance on Munson in this regard is limited to Munson at paragraph [0044], the entirety of which is reproduced below.

In one embodiment, a content can be pushed according to a schedule of time or event. For example, when used with a group push, an application service provider can use pushes (i.e., pushing contents) for a system maintenance purpose during off-peak hours to take advantage of lower rates.

Ex. 1017, [0044].

There are several problems with the Petition’s theory in this regard. *See* Ex. 2008, ¶¶138–142. First, as noted above, the baseline combination of Houghton and Munson would not have been obvious. This applies with even more force here, where the only example of “a schedule of time or event” being used to push a message in Munson is “when used with a group push” (Ex. 1017, [0044]), a functionality that Houghton does not disclose (and the Petition does not allege would be obvious to modify Houghton to implement). Ex. 2008, ¶138.

Second, Munson does not disclose the use of a periodic timer to trigger message delivery. Rather, as shown in the Munson excerpt above, Munson merely discloses that an application server (e.g., “application service provider”) can decide to push content “according to a schedule of time or event.” Ex. 1017, [0044]. This certainly does not *disclose* the use of a periodic timer—as Amazon’s own patents acknowledge, an event can be accomplished “according to a published, but not necessarily periodic, schedule.” Ex. 2014 (U.S. Patent No. 12,210,419), claim 3. This is precisely what Munson discloses—pushing content “according to a schedule of time or event.” Ex. 1017, [0044]. While the Petition alleges in conclusory fashion that a POSITA “would have understood/found obvious would be triggered by the expiration of a regularly-repeating timer,” the Petition identifies no evidence supporting this assertion other than the similarly conclusory testimony of its expert Dr. Traynor. *See* Ex. 1003, ¶¶884-885 (merely repeating the same

conclusory allegation as the Petition). And critically, despite seemingly recognizing that Munson does not *disclose* the use of a periodic timer, the Petition identifies no motivation to *modify* Munson’s teachings to make use of a periodic timer. *See also* Ex. 2008, ¶¶139–140.⁶

⁶ Indeed, a “schedule” or “event” does not even imply a *timer*, because humans often will initiate push messages at an application server according to a work schedule, without using a strict “timer” to initiate the push. Pushing content according to “a schedule of time or event” would encompass, e.g., an application service provider deciding that with Halloween approaching, they may want to push Halloween-related content. This would not relate to the expiration of any timer, but rather simply coincide with the schedule / event of Halloween approaching. Likewise, it would encompass sending system maintenance updates (to the extent they are needed) at some point after 9pm the day the update is ready to go. This would not need to be a “timer” at all—it would instead commonly involve an employee of the application service provider triggering the update. Ex. 2008, ¶139. Notably, the Petition sets forth *no* obviousness theory as to *why* pushing according to a “schedule or event” would involve *any* sort of “timer.” Pet. 77.

This is fatal to the Petition’s theory, because Munson’s alleged “use of periodic timers for message delivery” is what motivates the Petition’s alleged combination to arrive at claim 3. *See* Pet. 77 (alleging that a POSITA would have been “motivated... in implementing *Munson’s use* of a regularly-repeating timer to cause attempted message delivery in Houghton”); Ex. 1003, ¶885 (“Given Houghton’s disclosures of delivery triggers and *Munson’s use of periodic timers for message delivery*, in my opinion, a POSITA would have been motivated and would have had a reasonable expectation of success in implementing Munson’s teaching of using periodic timers”). The also asserts that a POSITA would have been motivated to use Munson’s “periodic timer as a message delivery trigger to enable more efficient use of network resources, avoid repeated message requests, and attempt delivery according to a desired schedule.” Pet. 77. However, the Petition makes no effort to explain why that rationale would apply to Houghton, which discloses no repeated message requests from the push client that a periodic delivery timer would eliminate: Houghton maintains a permanent IP connection over which the push server sends messages when trigger events occur, and the only periodic messages Houghton discloses on that connection are keep-alive messages, which a periodic delivery timer would not displace. Ex. 1007, 11, 21, 26; Ex. 2008, ¶140.

Furthermore, even if Munson *did* disclose the use of periodic timers for message delivery, and a POSITA *would* have found it obvious to combine that (non-existent) disclosure with Houghton, it still would not satisfy the claim requirements. Claim 1 requires that the claimed network server system include “logic to determine when one of a plurality of message delivery triggers for a given one of the wireless end-user devices has occurred,” with claim 3 further specifying that “one of the [plurality of] message delivery triggers is the expiration of a periodic timer.” Accordingly, the server must have logic to determine when a periodic timer has expired. But the Petition identifies nothing in the push server of the proposed Houghton/Munson combination that would have “logic to determine when” a message transmission was requested using a periodic timer. For instance, *even if* an application server in the proposed combination initiated content pushes according to a periodic timer, that does not mean that the push server would have logic to determine *when* such a periodically-triggered event took place. Ex. 2008, ¶¶141-142.

In summary, the Petition’s theory for claim 3 based on Houghton and Munson fails because it would not have been obvious to combine the references in the first place; neither reference discloses the use of a periodic timer as a message delivery trigger (even under the Petition’s broad interpretation of “message delivery trigger”), such that even if the references were combined they would not

satisfy the use of a periodic timer as a message delivery trigger. And finally, even if expiration of a periodic timer could trigger message delivery, the Petition identifies no logic in the combined push server to “determine when” the expiration of a periodic timer has occurred as required.

This deficiency applies equally to the Petition’s Ground 2B, which relies on a further combination with TS-23.140 as applied to claims 1-7 (including claim 3), but which combination is not alleged to cure any of the above deficiencies with respect to claim 3. See generally Pet. 83–85 (analysis of Ground 2B addressing only application registration and application identifiers, and not the requirements of claim 3). Furthermore, because claim 4 depends from claim 3, the deficiencies discussed above apply equally with respect to claim 4. See also Ex. 2008, ¶143.

2. The Petition’s Ground 2C fails with respect to claims 3 and 4.

The Petition alleges that a POSITA “would have found it obvious to supplement Houghton-Munson’s triggers... with Adamczyk’s additional trigger types,” specifically, by “provid[ing] users the option to schedule delivery of messages—including on a ‘periodic schedule,’ as Adamczyk teaches, sufficient to control when messages are pushed to their devices.” Pet. 85. The Petition also alleges that “POSITAs would have been motivated to do so with a reasonable expectation of success for the same reasons described regarding TS-23.140-

Adamczyk.” Pet. 85-86. However, as noted above in Section VI.B, a POSITA would not have been motivated or reasonably expected to succeed in making the TS-23.140 and Adamczyk combination, and the Petition’s incorporation by reference of arguments with respect to TS-23.140 and Adamczyk thus fails. Ex. 2008, ¶144.

Furthermore, a POSITA would not have recognized any benefits in combining Houghton with Adamczyk in the manner the Petition proposes. First, Houghton discloses that, with the exception of data that should be transmitted “in the most cost efficient manner” (e.g., “lazy application code update[s] and lazy application data update[s]”) (*see, e.g.*, Ex. 1007, 24), messages within its system should be transmitted as rapidly as possible due to the need for “real time information, interactivity and rich media content” messaging. *See* Ex. 1007, 10–11 (“Given the shortcomings in the prior art, there is a need for a method whereby a computer on the Internet can rapidly and at reasonable cost be permitted to at any time initiate control of one or more applications on an individual's mobile terminal to deliver through that mobile application real time information, interactivity and rich media content....”). A POSITA would understand that transmitting messages according to expiration of a periodic timer would not be suitable with Houghton’s goal of transmitting “real time information, interactivity and rich media content” “rapidly,” because doing so would unnecessarily delay Houghton’s transmission of

such content which Houghton repeatedly describes as time-sensitive. *See, e.g.*, Ex. 1007, 10 (noting that “low speed” solutions are not suitable for the type of time-critical messaging contemplated by Houghton); Ex. 2008, ¶145.

The sole disclosed exception to the time-critical needs of Houghton’s messaging system is specified in the context of data that should be “managed and updated automatically... in the most cost efficient manner for non-time-critical capabilities such as lazy application code update[s] and lazy application data update[s].” *E.g.*, Ex. 1007, 24. A POSITA, however, would not be motivated to use a periodic timer to initiate message delivery in this context either, because Houghton teaches a better approach for handling transmission of such data. Specifically, Houghton discloses that a push server will receive “connection availability information,” either from a mobile terminal or from a “device with a higher bandwidth capacity” (e.g., a “broadband access point”—Ex. 1007, 14) so that “the push server uses the connection availability information as a trigger event for sending lazy or low priority messages... routed through the higher bandwidth capacity device to the wireless terminal.” Ex. 1007, 33–34 (claims 24–28, each discussing transmitting non-time-critical data using high bandwidth networking options). Ex. 2008, ¶146.

In other words, in the sole exception that Houghton teaches to time-critical messaging, Houghton teaches a better and more efficient solution for its needs:

transfer of data according to the availability of high-efficiency transfer options, rather than transfer of data according to expiration of a periodic timer. And nothing in Houghton, Munson, or Adamczyk counsels against this solution and in favor of the Petition's hindsight-motivated approach. For instance, as noted above, Houghton and Munson do not disclose triggering transmission of data based on expiration of a periodic timer at all, much less disclose any benefits to doing so. And while Adamczyk *does* disclose sending notification messages on a "periodic schedule," it does so in passing, as one of several options, with no disclosure of any actual benefits of doing so. *See, e.g.*, Ex. 1021 at [0010] ("In still other embodiments, the notification server is configured to send the notification messages to the recipients on demand, at a specific future time, and/or on a periodic schedule."). This non-descript discussion of the use of a periodic schedule would certainly not motivate a POSITA to deviate from Houghton's detailed discussion of the benefits of either (1) rapid transmission for time critical messaging, or (2) cost-efficient transmission on high-bandwidth networks for non-time critical messaging. Ex. 2008, ¶¶146-147.

Furthermore, the Petition's theory is that under its proposal, it would "require configuring push server to... push[ing] messages received during the time interval at once." Pet. 86. A POSITA would understand that this could result in a large number of messages being transferred simultaneously, which could overload

the network demands of the device, particularly for non-time-critical messaging of the type contemplated by Houghton which would involve larger data transfers. Nowhere does the Petition even attempt to explain why a user would prefer to receive numerous messages at once in the context of Houghton's system, much less provide evidence that a POSITA would have found it obvious to do so. And because claim 4 depends from claim 3, the Petition's Ground 2C showing for claim 4 (Pet. 86) fails for the same reasons. Ex. 2008, ¶¶148-149.

D. The Petition's Ground 2 additionally fails to establish the obviousness of claim 17.

The Petition contends that claim 17 would have been obvious over Houghton, Munson, and Fok. Pet. 93-94. Ground 2G is effectively Ground 1G with Houghton's push client substituted for the MMS User Agent, and the Petition offers no motivation particular to it: its only stated reason for making the combination is "the same reasons discussed regarding TS-23.140-Fok." Pet. 94. Ground 2G accordingly fails for the reasons set forth in Section VII above, which Patent Owner incorporates here, and the Petition's reliance on its Ground 1G reasoning carries that reasoning's defects into Ground 2G. Ex. 2008, ¶¶150-165.

Like the Petition's TS-23.140-Fok combination, the Petition's Houghton-Munson-Fok combination interposes an intermediary in what Fok teaches as direct communication between applications, and so doubles the handshakes and

communications required for a single transmission. *See* Pet. 93 (alleging that its combination would “provide ‘secure inter-application communication’ (secure interprocess communication service) for two or more applications residing on a single terminal to transmit application-specific messages to one another ***through [the] push client on the terminal***”); Ex. 2008, ¶¶151, 158. Dr. Traynor concedes the departure, acknowledging that “although Fok discloses a direct handshake between the two applications,” his combination “would instead pass through the push client (so one application sends the push client a message, which the push client then forwards to the receiving application).” Ex. 1003, ¶981; Ex. 2008, ¶¶151–152. The Petition offers no rationale for obviousness independent of the Petition’s Ground 1G combination (*see* Pet. 93–94), and the Petition’s theory fails for the same reasons. *See* Ex. 2008, ¶¶151–159.

And while the Petition never addresses the differences between its two combinations, the differences do not help the Petition. First, the intermediary is Houghton’s push client rather than the MMS User Agent, but the Petition identifies nothing about the push client that would be better suited to the combination than TS-23.140’s User Agent. Dr. Traynor acknowledges that Houghton-Munson “does not expressly describe inter-application transmissions on a given device,” which means it prescribes no protocol for such transmissions and no handshake for the applications to reuse. Ex. 1003, ¶978; Ex. 2008, ¶153.

Second, whereas TS-23.140 describes no handshake at all, Houghton describes one: the “3-way handshake during connection establishment” that Houghton identifies as overhead best incurred before transmission precisely so as to avoid “delaying the transmission.” Ex. 1007, 20. In other words, Houghton confirms that a POSITA would have understood that doubling the handshakes required for each transmission would be undesirable because it would cause delays. Ex. 2008, ¶158. (Munson describes no handshake at all. *See generally* Ex. 1017.)

Third, whereas the TS-23.140-Fok proposal would confine secure interprocess communication to “MMS-compliant applications,” the Houghton-Munson-Fok proposal would confine it to those applications that already interface with the push client. Ex. 1003, ¶980; Ex. 2008, ¶159. That limitation is the same in substance, and a POSITA who saw value in secure on-device communication would have had no reason to modify Fok’s teachings only to arrive at a more limiting (and more inefficient) result.

Ground 2G separately fails to satisfy the claim language, for the reasons given in Section VII.B. The Petition again equates the claimed “interprocess communication service” with Fok’s “secure inter-application communication” (Pet. 93), which is a handshake protocol that the communicating applications implement between themselves rather than a service provided to them. And the

Petition again identifies nothing in Houghton-Munson or in Fok that constitutes the claimed service, or that shows any such service to be itself “secure.” Ex. 1003, ¶¶977-983 (using the word “service” only in restating the claim language); Ex. 2008, ¶¶160-165. The Petition therefore fails to establish that claim 17 is unpatentable.

IX. Conclusion

For at least the foregoing reasons, Patent Owner respectfully requests that all challenged claims be found not unpatentable.

Date: August 4, 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 12,868 words in the paper excluding the portions exempted under 37 C.F.R. §42.24(a)(1).

Date: August 4, 2026

/James A. Milkey/
James A. Milkey, Reg. No. 79,503
RUSS, AUGUST & KABAT
12424 Wilshire Blvd., 12th Floor
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 August 4, 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:

Jessica Kaiser (Reg. No. 58,937)
PERKINS COIE LLP
1900 Sixteenth Street, Suite 1400
Denver, CO 80202-5255
kaiser-ptab@perkinscoie.com

Christopher Marando (Reg. No. 67,898)
PERKINS COIE LLP
700 Thirteenth Street N.W.
Suite 800
Washington, DC 20005-3960
Marando-ptab@perkinscoie.com

Matthew A. Lembo (Reg. No. 75,633)
PERKINS COIE LLP
1155 Avenue of the Americas 22nd Floor
New York, NY 10036-2711
Lembo-ptab@perkinscoie.com

Amazon-Headwater-IPR@perkinscoie.com

Date: August 4, 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