

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

LENOVO (UNITED STATES) INC. and
MOTOROLA MOBILITY LLC,

Petitioner

v.

HEADWATER RESEARCH LLC,

Patent Owner

Patent No. 9,198,076
Case No. IPR2024-01180

PETITION FOR *INTER PARTES* REVIEW

TABLE OF CONTENTS

I.	Introduction.....	1
II.	The '076 Patent.....	1
	A. Overview	1
	B. Prosecution History	3
III.	Level of Ordinary Skill.....	4
IV.	Claim Construction.....	4
V.	Statement of Precise Relief Requested.....	5
	A. Ground 1: Claims 1-2, 5, and 7-16 Are Obvious over <i>Rao</i> in View of <i>Araujo</i>	6
	1. Overview of <i>Rao</i>	6
	2. Overview of <i>Araujo</i>	9
	3. Rationale to Combine	13
	4. Claim 1	18
	a. 1[pre]: “A wireless end-user device, comprising:”	18
	b. 1[a]: “a wireless wide area network (WWAN) modem to communicate data for Internet service activities between the device and at least one WWAN, when configured for and connected to the at least one WWAN; and” 19	
	c. 1[b]: “one or more processors”.....	20
	d. 1[c]: “configured to, for a time when data communication for Internet service activities is provided by the WWAN modem and the at least one WWAN, classify whether a first end-user application associated with an	

Internet access request, and capable of both interacting with a user in a user interface foreground of the device, and at least some Internet service activities when not interacting with a user in the device user interface foreground, is interacting with the user in the device user interface foreground, and,”	22
e. 1[d(1)]: “allow or disallow the Internet access request based on the classification as to whether the first end-user application is interacting with the user and based on a power control state,”	25
f. 1[d(2)]: “such that, when the power control state is a first power control state and the first end-user application is classified as not interacting with the user, the Internet access request is disallowed, and”	30
g. 1[d(3)]: “when the power control state is at least one other power control state, the Internet access request is allowed.”	31
5. Claim 2: “The wireless end-user device of claim 1, wherein the one or more processors are configured to classify that the first end-user application is interacting with the user in the device user interface foreground when the user of the device is directly interacting with that application or perceiving any benefit from that application.”	33
6. Claim 5: “The wireless end-user device of claim 1, further comprising, when the Internet access request is disallowed, queuing the Internet access request until a power control state change occurs.”	34
7. Claim 7: “The wireless end-user device of claim 1, wherein the one or more processors are further configured to allow the Internet access request when the power control state is the first power control state and the first end-user application is classified as interacting with the user.”	36

8. Claim 8: “The wireless end-user device of claim 1, further comprising a user interface, wherein the user interface is to inform the user of the device when there are options to set, control, override, or modify service usage controls that affect the processor configuration that determines whether to allow or disallow Internet access requests in the first power control state.” 37
9. Claim 9 39
 - a. 9[a]: “The wireless end-user device of claim 1, further comprising: a wireless local area network (WLAN) modem to communicate data for Internet service activities between the device and at least one WLAN, when configured for and connected to the at least one WLAN,” 39
 - b. 9[b]: “wherein the one or more processors are further configured to allow the Internet access request when the WLAN modem provides data communication for the Internet access request and the power control state is the first power control state.” 40
10. Claim 10: “The wireless end-user device of claim 1, wherein the power control state is a power state of the device.” 42
11. Claim 11: “The wireless end-user device of claim 1, wherein the first power control state is a power save state of the device.” 43
12. Claim 12: “The wireless end-user device of claim 1, wherein the power control state is a power state of the WWAN modem.” 44
13. Claim 13: “The wireless end-user device of claim 1, wherein the one or more processors are further configured to, based on a device usage state, dynamically change a determination of whether to allow or disallow the Internet access request in the first power control state when the first

	end-user application is classified as not interacting with the user.”	45
14.	Claim 14: “The wireless end-user device of claim 1, wherein the one or more processors are further configured to, based on power control state changes for the WWAN modem, dynamically change a determination of whether to allow or disallow the Internet access request in the first power control state when the first end-user application is classified as not interacting with the user.”	47
15.	Claim 15: “The wireless end-user device of claim 1, wherein, when the Internet access request is disallowed, the one or more processors are configured to prevent the Internet access request from causing a change to a power state of the modem.”	49
16.	Claim 16: “The wireless end-user device of claim 1, wherein the one or more processors are configured to, when the Internet access request is disallowed, instruct the first end-user application to transition to a different state.”	50
B.	Ground 2: Claim 3 Is Obvious over <i>Rao</i> in View of <i>Araujo</i> and <i>Singh</i>	52
1.	Overview of <i>Singh</i>	52
2.	Rationale to Combine	53
3.	Claim 3: “The wireless end-user device of claim 1, wherein the one or more processors are configured to classify that the first end-user application is interacting with the user in the device user interface foreground based on a state of user interface priority for the application.”	55
C.	Ground 3: Claim 4 Is Obvious over <i>Rao</i> in View of <i>Araujo</i> and <i>Freund</i>	56
1.	Overview of <i>Freund</i>	56

2.	Rationale to Combine	57
3.	Claim 4: “The wireless end-user device of claim 1, wherein the one or more processors are configured to classify that the first end-user application is not interacting with the user in the device user interface foreground when the application is providing or utilizing a background data service.”	59
D.	Ground 4: Claim 6 Is Obvious over <i>Rao</i> in View of <i>Araujo</i> and <i>Aleksic</i>	61
1.	Overview of <i>Aleksic</i>	61
2.	Rationale to Combine	62
3.	Claim 6: “The wireless end-user device of claim 1, wherein the one or more processors are further configured to, for at least a second application or service, allow Internet access requests in the first power control state without regard to whether the second application or service is classified as interacting with the user.”	63
E.	Ground 5: Claim 16 Is Obvious over <i>Rao</i> in View of <i>Araujo</i> and <i>Montemurro</i>	64
1.	Overview of <i>Montemurro</i>	64
2.	Rationale to Combine	65
3.	Claim 16: “The wireless end-user device of claim 1, wherein the one or more processors are configured to, when the Internet access request is disallowed, instruct the first end-user application to transition to a different state.”	67
VI.	The Petition Is Proper Under 35 U.S.C. § 325(d)	69
VII.	The Petition Is Proper Under <i>Fintiv</i>	70
VIII.	Mandatory Notices.....	71
A.	Real Party-in-Interest	71

B.	Related Matters.....	71
C.	Lead and Back-Up Counsel.....	72
D.	Service Information.....	72
IX.	Grounds for Standing.....	73
X.	Conclusion	73

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>Advanced Bionics, LLC v. MED-EL Elektromedizinische Geräte GmbH,</i> IPR2019-01469, Paper 6 (PTAB Feb. 13, 2020).....	69, 70
<i>Apple Inc. v. Fintiv Inc.,</i> IPR2020-00019, Paper 11 (PTAB Mar. 20, 2020).....	70, 71
<i>Ariosa Diagnostics v. Verinata Health, Inc.,</i> 805 F.3d 1359 (Fed. Cir. 2015)	43
<i>Becton, Dickinson, & Co. v. B. Braun Melsungen AG,</i> IPR2017-01586, Paper 8 (PTAB Dec. 15, 2017)	69
<i>Brown v. 3M,</i> 265 F.3d 1349 (Fed. Cir. 2001)	33
<i>Keynetik, Inc. v. Samsung Elecs. Co.,</i> No. 2022-1127, 2023 WL 2003932 (Fed. Cir. Feb. 15, 2023).....	17
<i>Navistar, Inc. v. Fatigue Fracture Tech., LLC,</i> IPR2018-00853, Paper 13 (PTAB Sept. 12, 2018).....	69
<i>Phillips v. AWH Corp.,</i> 415 F.3d 1303 (Fed. Cir. 2005) (en banc)	4
Statutes	
35 U.S.C. § 102(a)	5, 6
35 U.S.C. § 102(b)	5, 6
35 U.S.C. § 102(e)	5, 6
35 U.S.C. § 103	5
35 U.S.C. § 112, ¶2.....	4

35 U.S.C. § 311	5
35 U.S.C. § 314(a)	70, 71
35 U.S.C. § 325(d)	70

Regulations

37 C.F.R. § 42.100(b)	4
-----------------------------	---

Other Authorities

USPTO Memorandum, Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation (June 21, 2022).....	70-71
--	-------

LIST OF EXHIBITS

<u>Exhibit</u>	<u>Description</u>
Ex-1001	U.S. Patent No. 9,198,076 to Raleigh et al. (“the ’076 patent”).
Ex-1002	Patent Center Prosecution History of U.S. Patent No. 9,198,076.
Ex-1003	Declaration of Harry Bims, Ph.D.
Ex-1004	Curriculum Vitae of Harry Bims, Ph.D.
Ex-1005	U.S. Patent Appl. Publ. No. 2006/0039354 to Rao et al. (“ <i>Rao</i> ”).
Ex-1006	U.S. Patent Appl. Publ. No. 2009/0217065 to Araujo (“ <i>Araujo</i> ”).
Ex-1007	U.S. Patent Appl. Publ. No. 2009/0207817 to Montemurro et al. (“ <i>Montemurro</i> ”).
Ex-1008	International Appl. Publ. No. WO 2008/113986 to Scahill (“ <i>Scahill</i> ”).
Ex-1009	U.S. Patent Appl. Publ. No. 2007/0038763 to Oestvall (“ <i>Oestvall</i> ”).
Ex-1010	U.S. Patent No. 5,987,611 to Freund (“ <i>Freund</i> ”).
Ex-1011	U.S. Patent No. 8,028,060 to Wyld et al. (“ <i>Wyld</i> ”).
Ex-1012	U.S. Patent Appl. Publ. No. 2009/0093247 to Srinivasan (“ <i>Srinivasan</i> ”).
Ex-1013	U.S. Patent Appl. Publ. No. 2008/0080458 to Cole (“ <i>Cole</i> ”).
Ex-1014	U.S. Patent Appl. Publ. No. 2008/0057894 to Aleksic et al. (“ <i>Aleksic</i> ”).
Ex-1015	U.S. Patent Appl. Publ. No. 2009/0119773 to D’Amore et al. (“ <i>D’Amore</i> ”).
Ex-1016	U.S. Patent No. 8,909,211 to Huq et al. (“ <i>Huq</i> ”).
Ex-1017	U.S. Patent No. 8,381,127 to Singh et al. (“ <i>Singh</i> ”).

<u>Exhibit</u>	<u>Description</u>
Ex-1018	U.S. Patent No. 6,898,654 to Senior et al. (“ <i>Senior</i> ”).
Ex-1019	U.S. Patent Appl. Publ. No. 2009/0318124 to Haughn (“ <i>Haughn</i> ”).
Ex-1020	U.S. Patent No. 8,886,261 to Aerrabotu (“ <i>Aerrabotu</i> ”).
Ex-1021	European Patent Appl. No. 1,484,871 to Kelz et al. (“ <i>Kelz</i> ”).
Ex-1022	Scheduling Order, <i>Headwater Research LLC v. Motorola Mobility LLC, et al.</i> , No. 4:23-cv-04496-JST (N.D. Cal. Jan. 30, 2024) (ECF No. 46).
Ex-1023	Joint Claim Construction and Prehearing Statement, <i>Headwater Research LLC v. Motorola Mobility LLC, et al.</i> , No. 4:23-cv-04496-JST (N.D. Cal. June 11, 2024) (ECF No. 64).

I. Introduction

Lenovo (United States) Inc. and Motorola Mobility LLC (“Petitioners”) request *inter partes* review of claims 1-16 of U.S. Patent No. 9,198,076 (Ex-1001). The ’076 patent relates to a wireless end-user device that manages application requests to access network resources on an application-by-application basis. As the patent explains, one or more processors in the wireless device classify whether an application associated with an Internet access request is interacting with a user. (’076 patent, Abstract.) Then, depending on a power control state (of the device or of a wireless modem), the Internet access request is either allowed or disallowed. (*Id.*, Abstract, claim 1.)

There is nothing novel about any of the claimed elements or their combination. Instead, all claimed elements were known individually and in various combinations before the ’076 patent’s earliest priority date, including as taught by *Rao* (Ex-1005), *Araujo* (Ex-1006), *Singh* (Ex-1017), *Freund* (Ex-1010), *Aleksic* (Ex-1014), and *Montemurro* (Ex-1007). Considering these references, Petitioners request that the Board institute review and cancel the challenged claims.

II. The ’076 Patent

A. Overview

According to the ’076 patent, wireless network resources are stretched thin. “The increasing popularity of various smart phone devices, net book devices, [and]

tablet computing devices” is leading to a “network capacity crunch.” (’076 patent, 3:28-37.) Network resources, e.g., base stations, are inundated with requests for network access (i.e., requests for Internet access), leading to “overloading” of base stations. (*Id.*, 6:19-67.) When this happens, base stations become “overwhelmed” by the requests and may begin “fall[ing] behind” in servicing the requests. (*Id.*, 6:56-67.)

The ’076 patent explains that a device’s access requests may vary in several ways. First, they may vary in their priority. Typically, each request is associated with an application. (*Id.*, 4:6-21.) Because the services provided by these applications vary, the priority of requests may vary. For example, a service that is “interactive” in the foreground of a user’s device may be higher priority because it requires “reasonably low response time” for a request, while a service occurring in the background may be “lowest priority.” (*Id.*, 16:35-44.) For some services, it may make sense to hold off sending requests until a “scheduled transmission time.” (*Id.*, 86:44-55.)

Second, requests may vary in their likelihood to lead to repeated requests. As one example, a device may permit an application to access the network and then later interrupt the access due to the device’s “power save algorithm.” (*Id.*, 7:57-62.) This algorithm causes the device to “terminate[] the connection to the base station” each time the device enters a “power save state” and to reinitiate the

connection when exiting the “power save state.” (*Id.*, 7:50-57.) As a result, “each time the device enters [the] power save state and then exits [the] power save state[,] network resources are consumed.” (*Id.*, 7:42-57.) If the device employs “an aggressive power save algorithm that enters [the] power save state after a short idle period,” repeated requests for network access occur, causing the device to “consume a proportionally large amount of resources.” (*Id.*, 7:50-62; *see also id.*, 93:31-42.)

To avoid “overloading” the base stations, the ’076 patent explains, the device intercepts application requests to access the network. (*Id.*, 6:19-67, 11:15-37, 65:12-28.) The device then handles requests differently based on conventional classifications, such as priority or likelihood to lead to repeated requests, as set forth in a “control policy.” (*Id.*, 11:46-49, 12:38-46.) For example, one or more device processors classify whether an application associated with an Internet access request is interacting with a user. (*Id.*, Abstract.) Then, depending on a power control state (of the device or of a wireless modem), the Internet access request is either allowed or disallowed. (*Id.*, Abstract, claim 1.)

B. Prosecution History

The ’076 patent was allowed without a single Office Action during a Track 1 prioritized examination. (*See generally* Ex-1002.) During prosecution, the Examiner did not apply any of this Petition’s references. (*See id.*)

III. Level of Ordinary Skill

A person of ordinary skill in the art (“POSITA”) at the effective filing date of the ’076 patent would have (1) held at least a bachelor’s degree in computer science, electrical engineering, or a related field, and (2) had three to five years of experience with networking, power consumption of networked computing devices, and/or wireless digital communications systems. (Ex-1003 (“Bims”), ¶¶24-26.) More related education would compensate for fewer years of work experience (and vice versa). (*Id.*)

IV. Claim Construction

Claims in an IPR are construed under the claim construction principles set forth in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc). 37 C.F.R. § 42.100(b). Petitioners believe no terms need to be expressly construed, so at this time Petitioners identify no terms for construction.

In the parallel district court litigation, the parties filed a Joint Claim Construction and Prehearing Statement (Ex-1023). The analysis in this Petition is consistent with the agreed construction (*id.*, 2) and satisfies both parties constructions for all disputed terms (*id.*, 2-3). Petitioners additionally argued the following term is indefinite under pre-AIA 35 U.S.C. § 112, ¶2: “perceiving any benefit from” (claim 2). But the Board need not reach that issue because the

“perceiving any benefit from” aspect of the claim is optional in the context of prior-art analysis, as discussed below.

V. Statement of Precise Relief Requested

Petitioners respectfully request review under 35 U.S.C. § 311 of claims 1-16 of the '076 patent and cancellation of those claims as obvious under pre-AIA 35 U.S.C. § 103 based on the following unpatentability grounds.

Ground	References	Challenged Claims
1	<i>Rao</i> in view of <i>Araujo</i>	Claims 1-2, 5, and 7-16
2	<i>Rao</i> in view of <i>Araujo</i> and <i>Singh</i>	Claim 3
3	<i>Rao</i> in view of <i>Araujo</i> and <i>Freund</i>	Claim 4
4	<i>Rao</i> in view of <i>Araujo</i> and <i>Aleksic</i>	Claim 6
5	<i>Rao</i> in view of <i>Araujo</i> and <i>Montemurro</i>	Claim 16

The '076 patent claims priority to multiple provisional applications, the earliest of which was filed January 28, 2009. ('076 patent, Cover.) While Petitioners do not concede that January 28, 2009, is the priority date that should be afforded to the '076 patent, all references asserted in this Petition qualify as prior art under pre-AIA 35 U.S.C. § 102(a), (b), and/or (e) even if the January 28, 2009, date is used as the priority date of the '076 patent:

Ex-1005	U.S. Patent Publication 2006/0039354 (“ <i>Rao</i> ”), published February 23, 2006; prior art under § 102(a), (b), and (e).
Ex-1006	U.S. Patent Publication 2009/0217065 (“ <i>Araujo</i> ”), filed February 26, 2008; prior art under § 102(e).
Ex-1007	U.S. Patent Publication 2009/0207817 (“ <i>Montemurro</i> ”), filed February 15, 2008; prior art under § 102(e).
Ex-1017	U.S. Patent 8,381,127 (“ <i>Singh</i> ”), filed February 2, 2006; prior art under § 102(e).
Ex-1010	U.S. Patent 5,987,611 (“ <i>Freund</i> ”), issued November 16, 1999; prior art under § 102(a), (b), and (e).
Ex-1014	U.S. Patent Publication 2008/0057894 (“ <i>Aleksic</i> ”), filed August 31, 2006, and published March 6, 2008; prior art under § 102(a) and (e).

A. Ground 1: Claims 1-2, 5, and 7-16 Are Obvious over *Rao* in View of *Araujo*

1. Overview of *Rao*

Like the '076 patent, *Rao* “relates to optimizing peer-to-peer network communications, and in particular, to techniques for application-aware prioritizations of network communications on a client.” (*Rao*, [0002].) *Rao*

discloses a “remote access architecture” for providing “peer-to-peer communications and remote access connectivity.” (*Rao*, Abstract.) *Rao* is analogous art to the ’076 patent. (Bims, ¶¶50-54.)

Rao addresses the inefficient use of network resources among competing network communications. Typically, these communications are “processed in the order generated by the activity of a user and applications of the client,” which is not “always desirable” since “a network packet generated or received for an application running in the background may be processed ahead of a network packet generated or received for the application running in the foreground.” (*Rao*, [0003].) This results in increased latency and reduced quality of voice communication. (*Id.*)

To address these inefficient prior-art architectures, *Rao* describes a prioritization scheme that “provide[s] application-aware, client-specific prioritization of packet traffic.” (*Id.*) *Rao*’s system includes one or more computing devices 102 (alternatively, clients 105) connected to a network 104 via one or more network connections 341. (*Rao*, [0086].) At client 105, the prioritization scheme includes a remote access client 120 that interacts with applications 338 and a network stack 310. (*Rao*, [0098], [0100].) Remote access client 120 “manages and controls network communication prioritizations on the client.” (*Rao*, [0080].) Figure 5A (below) shows a system having the remote access client for routing network packets from a client 105 to a network 104:

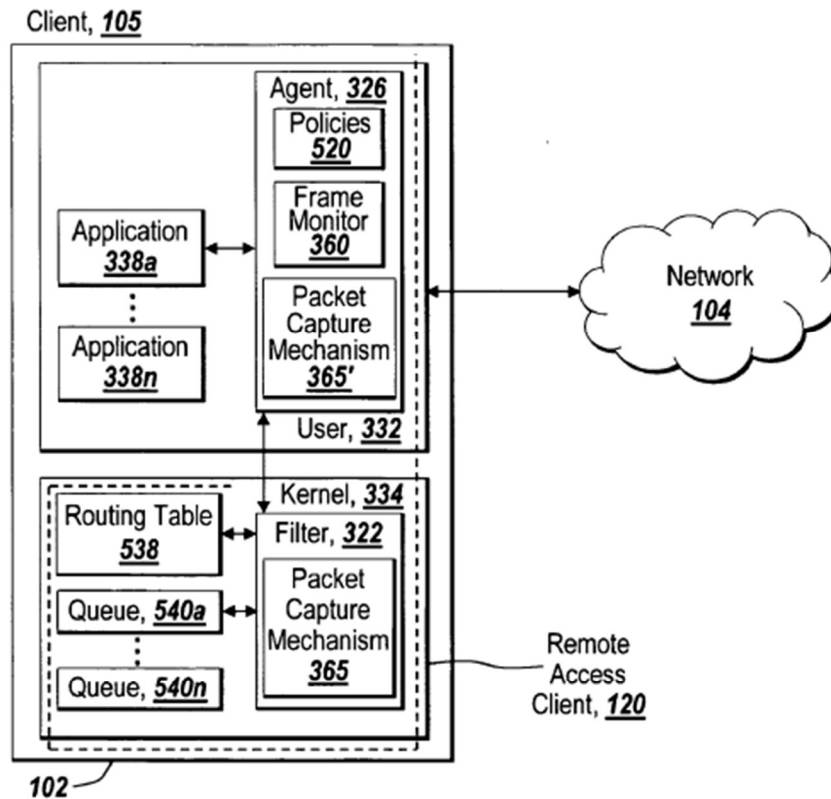


Fig. 5A

(*Rao*, Fig. 5A.) Client 105 runs an agent 326 and a filter 322, which in one embodiment may together form remote access client 120 “for routing packets via a network, or for otherwise providing remote access connectivity.” (*Rao*, [0099].) *Rao* discloses implementing filter 322 as a “network driver, such as a network driver operating at any layer, or portion thereof, of a network stack” at the client. (*Rao*, [0101].) *Rao*’s network stack includes, for example, “one or more network

drivers” associated with “any network interface cards or network access components of the computing device.” (*Rao*, [0100].)

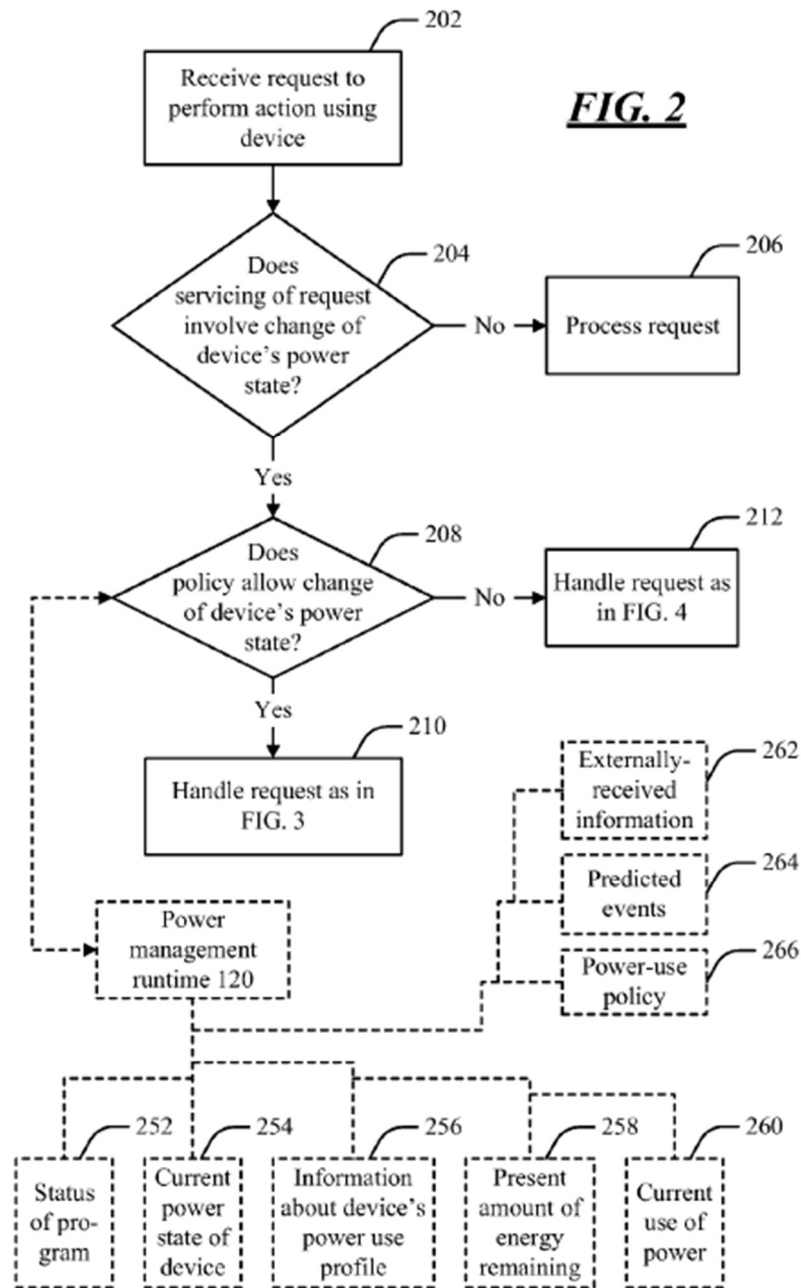
Rao’s prioritization can be based on policies, which “apply a priority to network packets of applications 338a-338n in accordance with [specified] prioritization rules.” (*Rao*, [0189].) *Rao*’s policies are specified by an application’s name, type, protocol, or any other suitable means. (*Rao*, [0182].) These policies offer several approaches for prioritization, including whether an application is in the foreground or background. (*Id.*)

2. Overview of *Araujo*

Araujo teaches application-aware optimization of power management. (Bims, ¶55.) Like the ’076 patent and *Rao*, *Araujo* implements policies to control activities of “wireless telephones” on an application-by-application basis. (*Araujo*, [0014].) *Araujo* is analogous art to the ’076 patent. (Bims, ¶¶55-61.)

Just as *Rao* uses policies to prioritize network access by certain applications over others, *Araujo* discloses using “power-management polic[ies]” to prioritize power consumption by certain applications (“programs” in *Araujo*) over others. (*Araujo*, Abstract.) As *Araujo* explains, “[s]ome programs merit the use of power to service any request[,]” while “[o]ther programs do not.” (*Araujo*, [0016].) Yet typical power management techniques “generally do not take [this] ... into account when evaluating requests” from programs “to power a device” at a “machine,”

such as a “wireless telephone[.]” (*Araujo*, [0014], [0016].) One example of a “device” is “machine 102’s wireless network card” that allows the machine “to send data over a wireless network.” (*Araujo*, [0020].)



(*Araujo*, Fig. 2.)

Figure 2 illustrates *Araujo*'s power management technique. (*Araujo*, [0023]-[0035].) At 202, a "request" is received from an application, such as "a request to send data over a wireless network" using a "network card." (*Araujo*, [0020], [0024].) The request is received by a "[r]equest deflector component," "interposed between" the application and a driver for the network card, that "intercepts requests and responds to the requests based on power management considerations before the request reach[es] a ... [network card] driver." (*Araujo*, [0020], [0024], [0050].) "At 204, it is determined whether servicing of the request involves a change in the [network card's] power state, such as a change from off to on, a change from one intermediate power state to another, or a change between an extreme power state (on or off) and an intermediate power state." (*Araujo*, [0025].) If so, "it may be determined (at 208) whether a policy (e.g., a power management policy) allows the [network card's] power state to be changed to service the request." (*Id.*) To make this determination, *Araujo* discloses considering the requesting application's "status" and a "current power state of the device." (*Araujo*, [0027]-[0028].)

The status "reflects its worthiness to consume power." (*Araujo*, [0027].) "For example," some applications "have a 'VIP' status that allows them to consume power in situations where other [applications] would not be permitted to consume power." (*Id.*) This status could be a "binary VIP/non-VIP status," or

“could be indicated on a scale.” (*Id.*) “Whether [an application] is permitted to consume power could be based on a finding as to whether the [application’s] status justifies the consumption of power under the circumstances that are present.” (*Id.*)

The “current power state of a device” is also considered in determining whether the network card’s power state can be changed to service the application’s request. (*Araujo*, [0028].) “[A] device [e.g., a network card] may use a considerable amount of power to turn from off to on,” but “changing a device from a low power state to a high power state may use less power than turning the device from off to on.” (*Id.*) “Thus, the current power state of the device may be a factor in deciding whether to change [the] power state, and the way in which the current power state is considered may be written into a policy.” (*Id.*)

Figure 2 shows a “power use policy” that “specifies how the various items of information,” including the “status” of the application and the “current power state of a device” (e.g., the network card), “are used to determine whether [the network card’s] power state will be changed.” (*Araujo*, [0027], [0028], [0034].) “If the policy allows the [network card’s] power state to be changed, then the request is allowed.” (*Araujo*, [0035].) But “[i]f the policy does not allow the [network card’s] power state to be changed ..., then the request may be processed in the manner shown in FIG. 4 (at [step] 212).” (*Id.*) For example, Figure 4 shows the application’s request may be “block[ed].” (*Araujo*, [0040].)

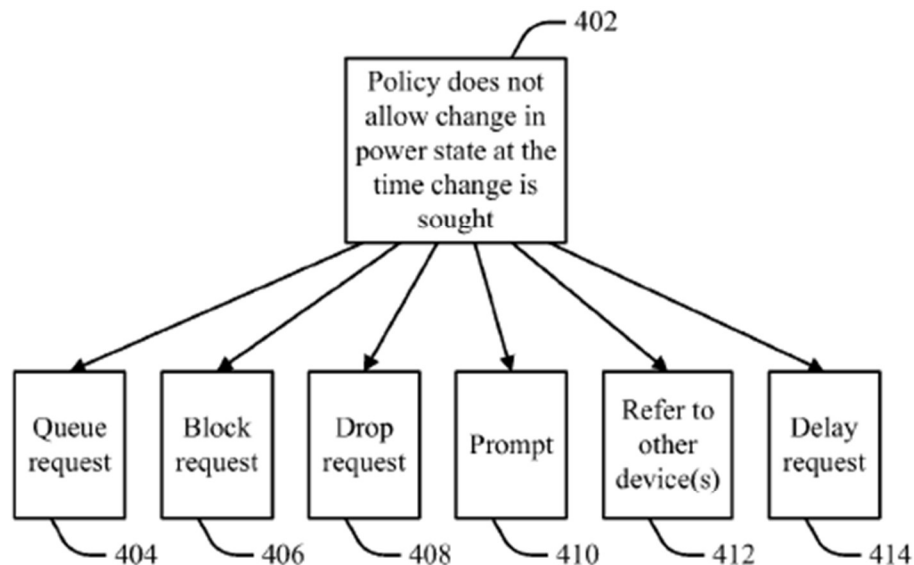


FIG. 4

(*Araujo*, Fig. 4.)

Thus, *Araujo* discloses a power management technique in which requests from applications to perform an action using a network card (one of *Araujo*'s "devices") may be allowed or blocked based on an application-specific policy. (Bims, ¶¶57-61.) By implementing such policies, a device "use[s] the energy in a way that strikes a balance between providing functionality and maintaining longevity of the charge" at the wireless telephone. (*Araujo*, [0001].)

3. Rationale to Combine

It would have been obvious to a POSITA to modify *Rao* to incorporate *Araujo*'s power management policy to provide a combination that better "use[s]

the energy in a way that strikes a balance between providing functionality and maintaining longevity of the charge.” (*Araujo*, [0001]; Bims, ¶¶62-69.) *Rao* discloses prioritizing network packets from applications based on application-by-application “prioritization rules specified or indicated by ... policies.” (*Rao*, [0189].) *Rao* envisions a “multitude of ways to define client-side priorities” in its policies, including “whether an application is running in the foreground or the background.” (*Rao*, [0182].) But *Rao* does not expressly disclose using a device’s power state in its policies to prioritize application network packets.

Araujo would have motivated a POSITA to augment *Rao*’s policies. (Bims, ¶63.) In particular, a POSITA would have been motivated to augment *Rao*’s policies to specify that network packets from certain applications should be allowed or blocked based not only on whether the application is currently in the foreground or the background but also on a device’s power state. (*Id.*)

As explained above, *Araujo* discloses a power management technique in which a request from an application to perform an action using a network card (one of *Araujo*’s “devices”) may be allowed or blocked based on an application’s power management policy. (Bims, ¶64.) This policy specifies whether to allow or block the request based both on an application status and the device’s power state. *Rao*’s policies, when augmented to consider the device’s power state, would specify whether an application request is to be allowed or blocked based both on whether

the application is currently in the device's foreground or background (taught by *Rao*, and similar to *Araujo*'s VIP/non-VIP status) and the device's power state. (*Id.*) These policies are based on characteristics that may change over time, requiring dynamic association of the policies as the application's status or device's power state changes. (*Id.*)

Araujo's request deflector functionality would be implemented in connection with or as part of the filter 322 of *Rao*'s remote access client, which—like *Araujo*'s request deflector—applies application-specific policies to determine how to handle applications' network access attempts. (Bims, ¶65; *Araujo*, [0024], [0050]; *Rao*, [0191]-[0193].) In this example, *Rao/Araujo*'s remote access client would employ application-specific policies that specify whether to allow or block a request from an application based on a check of (i) whether the application is currently in the device's foreground or the background (taught by *Rao*) and (ii) the device's power state (taught by *Araujo*). (Bims, ¶65.)

A POSITA would have been motivated to augment *Rao*'s policies to employ *Araujo*'s power management technique at least because *Araujo*'s technique “strike[s] the right balance between power conservation and performance”—a goal that was known in the prior art. (*Araujo*, [0015].) Battery-powered devices “have a finite amount of energy stored between charges.” (*Araujo*, [0001].) These devices therefore “attempt to use the energy in a way that strikes a balance between

providing functionality and maintaining longevity of the charge,” and “employ power management schemes to govern the use of energy.” (*Id.*) The prior art, however, recognized “a need ... for an improved power management operation in portable devices.” (*Aleksic*, [0004].) This was especially true for devices running multiple applications and having access to multiple network connections. (Bims, ¶66.) Thus, prior art like *Araujo* recognized that one method of improving a device’s power management was an application-aware prioritization of network communications that optimized the power consumption of individual applications. (*Id.*; *Araujo*, [0002]-[0005].)

A POSITA would have understood that improved, application-specific, power management schemes reduce competition among applications for other resources, including processing power at the device and network resources. (Bims, ¶67.) By selectively allowing network access, power management schemes reduce competition among applications for network resources. (*Id.*) The result is that certain applications may access the network at the expense of others. (*Id.*) This is particularly helpful, for example, when accessing a low bandwidth network or running a high-priority application requiring more processing power. (*Id.*) Using a scheme to manage an individual application’s power consumption optimizes a device’s resource consumption, improving an application’s operation and user experience. (*Id.*)

Indeed, *Rao* recognized that policy-based prioritization techniques “may improve or increase the performance, operational characteristics, and user experience on the client based on the applications running on the client.” (*Rao*, [0080].) A POSITA would have understood that the “performance, operational characteristics, and user experience” described in *Rao* would implicate, for instance, battery power, device longevity, and availability of network resources, but *Rao* does not expressly disclose its prioritization techniques serve to improve all of these aspects of the device. (Bims, ¶68.) A POSITA would have thus been motivated to modify *Rao* to include well-known, application-aware, power monitoring capabilities like those described in *Araujo* that reduce power consumption, decrease energy expenses, optimize network resource allocation, and increase device longevity. (*Id.*)

A POSITA would have had a reasonable expectation of success in modifying *Rao* to incorporate *Araujo*’s power management functionality. (Bims, ¶69.) Such a modification would have been well within the abilities of a POSITA. (*Id.*) See *Keynetik, Inc. v. Samsung Elecs. Co.*, No. 2022-1127, 2023 WL 2003932, at *2 (Fed. Cir. Feb. 15, 2023) (“Normally, once the function to be performed by software has been identified, writing code to achieve that function is within the skill of the art.” (citation omitted)). For example, *Araujo*’s policy “can be implemented as software that is stored in one or more of the data remembrance

component(s) 704 and that executes on one or more of the processor(s) 702.”

(*Araujo*, [0055].) This scheme could thus be implemented using *Rao*’s processor and memory units. (*See Rao*, [0119], [0120]; *Bims*, ¶69.) The resulting modification would have involved combining prior-art elements according to known methods to yield predictable results and use of a known technique to improve similar devices in the same way.

4. Claim 1

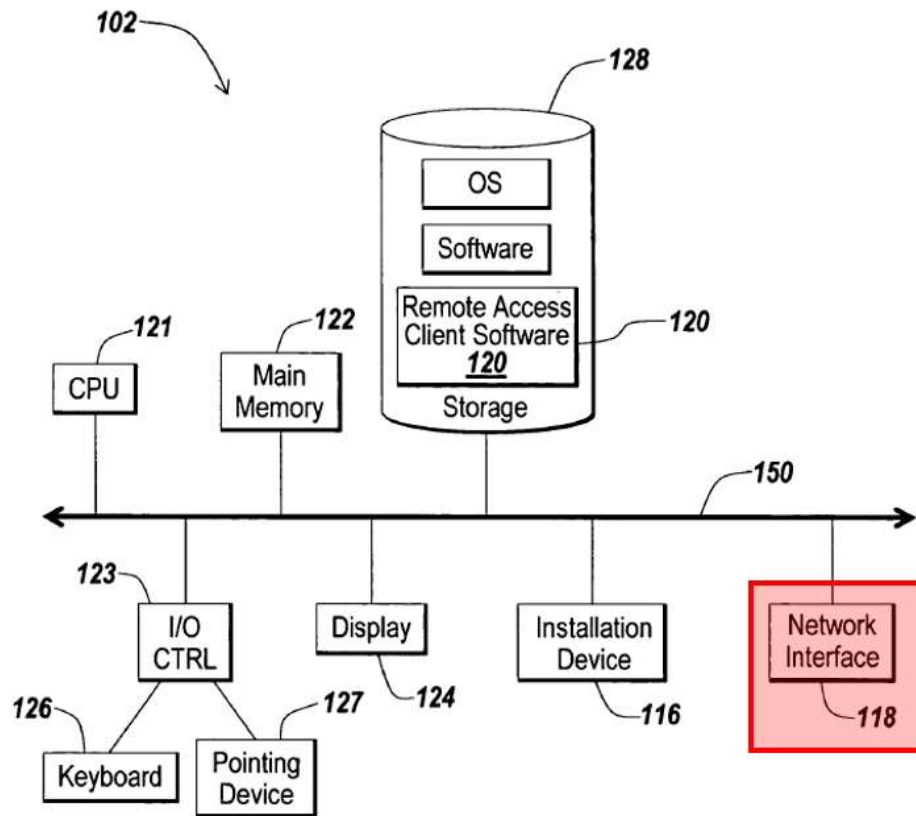
a. 1[pre]: “A wireless end-user device, comprising:”

The *Rao/Araujo* combination (“*Rao/Araujo*”) renders obvious the preamble, to the extent it is limiting. (*Bims*, ¶70.) *Rao* discloses “computing devices 102a-102c” (alternatively “clients 105a-105c”). (*Rao*, [0086].) *Rao*’s device “can be any type and/or form of computing device 102 that can run one or more applications 338.” (*Rao*, [0088].) For instance, *Rao*’s device may be a “mobile client,” such as “a smart phone” or “telecommunication device.” (*Rao*, [0198]; *see also Rao*, [0130].) *Rao*’s device “may include a network interface 118,” such as a “network interface card,” to “interface to a Local Area Network (LAN), Wide Area Network (WAN)[,] or the Internet through a variety of connections[,], including ... wireless connections.” (*Rao*, [0125].) *Rao*’s device is an “end-user device” because *Rao* discloses a device “user.” (*Rao*, [0183], [0199]; *Bims*, ¶70.)

b. 1[a]: “a wireless wide area network (WWAN) modem to communicate data for Internet service activities between the device and at least one WWAN, when configured for and connected to the at least one WWAN; and”

Rao/Araujo renders obvious element 1[a]. (Bims, ¶71.) *Rao*’s device includes network interface 118, which enables connectivity to a “Local Area Network (LAN), Wide Area Network (WAN)[,] or the Internet through a variety of connections,” such as “standard telephone lines, LAN or WAN links,” “broadband connections,” “*wireless* connections,” or “some combination of ... the above.” (*Rao*, [0095], [0125].)¹ Network interface 118 includes “a built-in network adapter, network interface card, PCMCIA network card, card bus network adapter, wireless network adapter, USB network adapter, [or] *modem*.” (*Rao*, [0125].) This is shown in Figure 1D below:

¹ All emphases added unless otherwise noted.

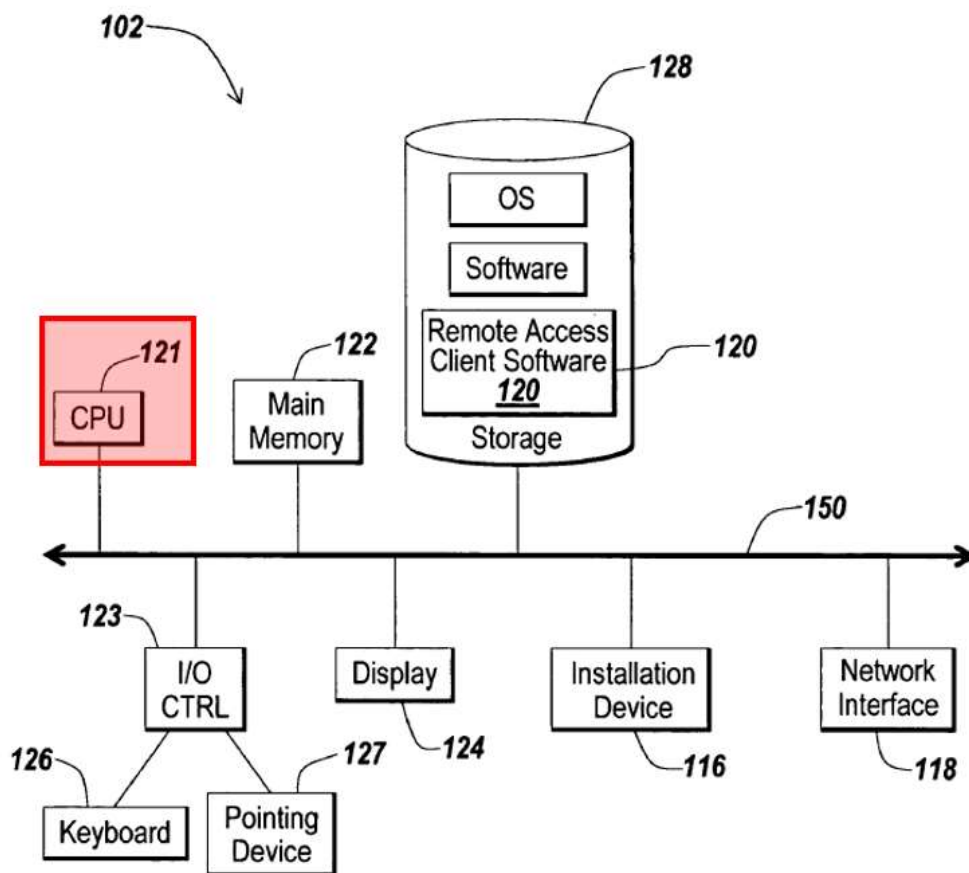


(*Rao*, Fig. 1D (annotated).) *Rao*'s network interface, including the modem, enables connectivity and data communication for Internet service activities between the device and a wireless wide area network (WWAN), such that *Rao* discloses a “[WWAN] modem,” as claimed. (Bims, ¶71.)

c. 1[b]: “one or more processors”

Rao/Araujo renders obvious element 1[b]. (Bims, ¶72.) *Rao*'s device includes “central processing unit 102” (“CPU”). (*Rao*, [0119], Fig. 1D.) According to *Rao*, “[t]he [CPU] is any logic circuitry that responds to and processes instructions fetched from the main memory unit 104.” (*Id.*) In some embodiments,

“the [CPU] is provided by a microprocessor unit.” (*Id.* (listing examples).) This CPU is shown below:



(*Rao*, Fig. 1D (annotated) (incorrectly labeled as “121” and not “102”).)

d. 1[c]: “configured to, for a time when data communication for Internet service activities is provided by the WWAN modem and the at least one WWAN, classify whether a first end-user application associated with an Internet access request, and capable of both interacting with a user in a user interface foreground of the device, and at least some Internet service activities when not interacting with a user in the device user interface foreground, is interacting with the user in the device user interface foreground, and,”

Rao/Araujo renders obvious element 1[c]. (Bims, ¶¶73-78.) *Rao*’s device includes “one or more applications 338a-338n.” (*Rao*, [0087].) These applications are shown below:

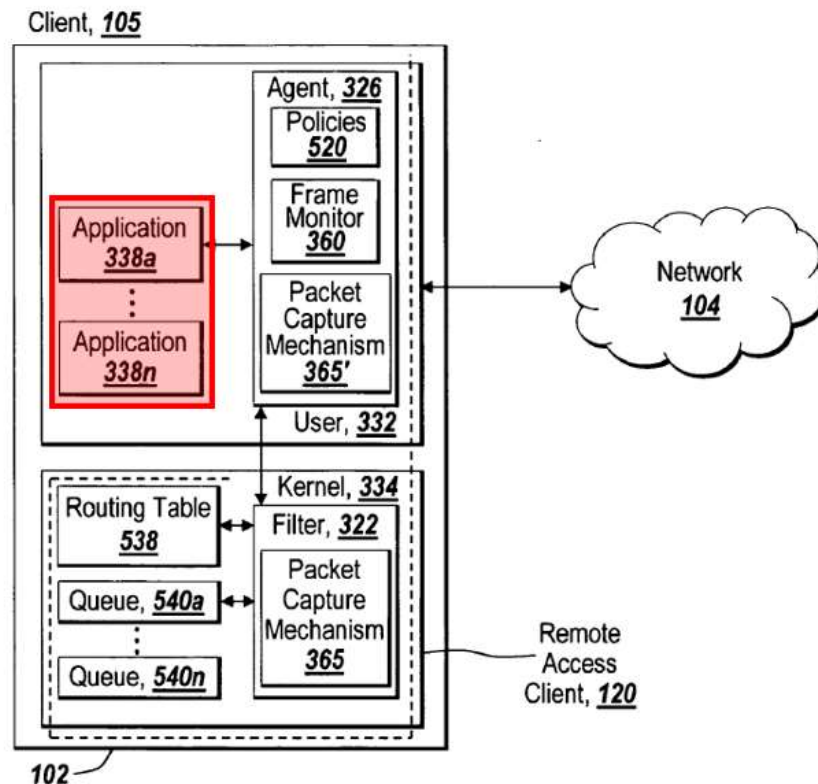


Fig. 5A

(*Rao*, Fig. 5A (annotated).) These applications “provide one or more real-time data communications, such as VoIP,” or “may provide email, collaboration, online meeting, and/or desktop[-]sharing[-]related services or functionality.” (*Rao*, [0179]; *see also Rao*, [0088] (describing breadth of applications); Bims, ¶73.) A POSITA would understand that these applications are associated with Internet access requests because they must request network access to access the Internet and carry out their respective services or functionality. (Bims, ¶73.) Each application is an “end-user application,” as claimed. (*Id.*)

Rao’s CPU runs a remote access client configured to classify whether a given application is running in a device’s foreground or background. (Bims, ¶74.) According to *Rao*, “remote access client 120 may determine whether the application 338a-338n associated with the network packet is running in the foreground or the background of the client 105.” (*Rao*, [0188]; *see also Rao*, [0189] (disclosing how “agent 326 may use characteristics of the application 338a-338n, such as running in the foreground or background”).) A POSITA would have understood that *Rao*’s remote access client runs on the CPU. (*See Rao*, [0118], [0119]; Bims, ¶¶74-75.)

A POSITA would have understood an application running in the foreground in *Rao* to be “interacting with a user in a user interface foreground of the device” because *Rao*’s remote access client classifies an application as “running in the

foreground” based on a user’s current interaction with that application. (Bims, ¶76.) According to *Rao*, an “application is running in the foreground” where it is “in active use by the user.” (*Rao*, [0003].) In the context of *Rao*’s device, an application running in the foreground would present an interface foreground to facilitate the interaction, as claimed. (Bims, ¶76.) For example, a POSITA would have understood that an application such as email would be capable of interacting with a user in a device’s user interface foreground (e.g., through an application like Microsoft Outlook) to create and send new messages. (*Id.*; *see also Rao*, [0090]; *Wyld*, 5:7-19.)

A POSITA would also have understood *Rao*’s applications to be “capable of both interacting with a user in a user interface foreground of the device, and at least some Internet service activities when not interacting with a user in the device user interface foreground” because it would have been well understood as of the ’076 patent’s priority date that an application (including *Rao*’s applications) would be **capable of** at least some Internet activities, even when running in the background. (Bims, ¶77.) For example, it was well known that an e-mail application was capable of using Internet access not only when in the foreground (e.g., when a user is sending emails) but also when in the background (e.g., to receive emails over the Internet even when the user is not interacting with the mobile device). (*Id.*; *see also Rao*, [0090]; *Wyld*, 5:7-19; *Kelz*, [0040].) Similarly,

an application such as a news aggregator may access the Internet in the background while the user attends to other unrelated tasks in the foreground. (Bims, ¶77; *Freund*, 10:24-27.) This is consistent with other references known to a POSITA. (Bims, ¶77 (citing *Oestvall*, [0006]).)

Rao's CPU is therefore configured to classify whether an end-user application—that is capable of both interacting with a user in a user interface foreground and at least some Internet service activities when not interacting with a user in the device user interface foreground—is interacting with the user in the device user interface foreground. (Bims, ¶78.) And the CPU is configured to perform this classification when data communication for Internet service activities is provided by the WWAN modem and the at least one WWAN, as discussed in Section V.A.4.b (claim 1[a]). (Bims, ¶78.)

e. 1[d(1)]: “allow or disallow the Internet access request based on the classification as to whether the first end-user application is interacting with the user and based on a power control state,”

Rao/Araujo renders obvious element 1[d(1)]. (Bims, ¶¶79-86.) As explained in Section V.A.3, a POSITA would have been motivated and found it obvious to augment the application-specific policies taught by *Rao* to consider not only whether the application is currently in the foreground of the device (“is interacting with the user”), but also a power state of the device (“a power control state”).

(Bims, ¶79.) Thus, in an example implementation, *Rao/Araujo*'s remote access client would allow or disallow an application's request for network access based both on (i) whether the application is currently in the device's foreground (taught by *Rao*) and (ii) the device's power state (taught by *Araujo*). (*Id.*) A POSITA would have understood that the operations of *Rao/Araujo*'s remote access client are undertaken by *Rao/Araujo*'s CPU ("one or more processors"). (*Id.*)

Indeed, *Araujo* expressly recognizes the value of considering an application's status in connection with a power state when determining whether to allow or disallow network access. (Bims, ¶80.) For instance, *Araujo* discloses that a "[n]etwork card" (e.g., modem) is "a device that may draw power while it is being used, and thus may have its power state managed." (*E.g., Araujo*, [0056].) This may include, for example, either a WWAN modem or a WLAN modem. *See* Sections V.A.9 (claim 9), V.A.14 (claim 14). Given that a network card "may use more power when transmitting," a "power-management policy" may assess whether a given application (or program) may consume power by accessing the Internet. (*Araujo*, Abstract, [0021]; Bims, ¶80.)

According to *Araujo*, "[w]hether a program is permitted to consume power," e.g., by powering on the network card to access the network, "could be based on a finding as to whether the program's status justifies the consumption of power under the circumstances that are present." (*Araujo*, [0027].) *Araujo* discloses that

“some programs may have a ‘VIP’ status that allows them to consume power in situations where other programs would not be permitted to consume power.” (*Id.*; *see also Araujo*, [0028], [0029], [0033].) A POSITA would have understood that an application (“program” in *Araujo*) having a VIP status—a status that, like *Rao*’s foreground status, indicates “high priority”—would be permitted to consume power (e.g., power on the network card to access the network), regardless of the device’s or network card’s power state. (Bims, ¶81.)

In contrast, *Araujo* discloses that some programs may have a non-VIP status that limits their ability to consume power in situations where VIP programs would be permitted to consume power. (*Araujo*, [0027], [0033] (explaining a non-VIP program cannot power on the network card); Bims, ¶82.) A POSITA would have understood that an application having a non-VIP status—a designation that, like *Rao*’s background status, indicates “low priority”—may not be permitted to consume power (e.g., power on the network card to access the network) when the device is in a particular power state, such as when the device or network card is off or in a low-power state. (Bims, ¶82.) Indeed, it was well known to a POSITA as of the priority date to use “priority[-]based functional processing capabilit[ies] ... to shut down if desired, lower priority applications to allow enough battery capacity to operate the primary or higher priority applications.” (*Aleksic*, [0033].) As the prior art explained, “[a]s the power capacity of the device battery decreases, lower

priority functions are either shut down or the quality of service of a feature is reduced in an attempt to conserve battery power.” (*Aleksic*, [0049].)

A POSITA would also have understood from *Araujo* that a non-VIP program would still be permitted to consume power (e.g., use the network card) when the network card is already powered on. (Bims, ¶84; *Araujo*, [0033] (explaining the non-VIP program may use the network card if it has already been powered on).)

From *Araujo*, a POSITA would have been motivated and found it obvious to augment *Rao*’s policies to consider an application’s foreground/background status together with a device’s power state, just as *Araujo* describes considering an application’s VIP/non-VIP status together with a device’s power state. (Bims, ¶85.) Just as *Araujo* describes allowing a VIP application to use the network card to access the network regardless of the device’s power state, *Rao/Araujo*’s augmented policies would allow a foreground application to access the network regardless of the device’s power state. (*Id.*) Further, just as *Araujo* describes disallowing a non-VIP application from using the network card to access the network when the network card is off, *Rao/Araujo*’s augmented policies would disallow a background application from accessing the network when *Rao/Araujo*’s network interface (or some other network access component) is powered off. (*Id.*) And just as *Araujo* describes allowing a non-VIP application to use the network card to

access the network when the network card is on, *Rao/Araujo*'s augmented policies would allow a background application to access the network when *Rao/Araujo*'s network interface (or some other network access component) is on. (*Id.*) It was known in the art that permitting network access by applications in the foreground—and thus, “currently in active use by the user” (*Rao*, [0003])—was desirable as doing so would reduce perceived latency. (Bims, ¶85.)

Thus, *Rao*'s policies, augmented according to *Araujo*, would allow or disallow an application's request for network access (“Internet access request”) (1) based on whether an application is in the foreground (“interacting with the user”) and (2) based on a device's power state (“power control state”). (Bims, ¶86.) In an example implementation, e.g., as described in Section V.A.3, these augmented policies would be applied by *Rao/Araujo*'s remote access client (implemented to include *Araujo*'s request deflector functionality), which sits between the applications and a driver for the network interface card (or other network access components) in *Rao*'s client. (Bims, ¶86; *Rao*, [0101], Fig. 5A.) In this implementation, when a request from an application was received, *Rao*'s remote access client would “intercept[] requests” for network access “and respond[] to the requests ... before the request reach[es] [the] device driver.” (*Araujo*, [0024], [0050].) Applying the augmented policies, *Rao/Araujo*'s remote access client would allow or disallow (e.g., “block,” *Araujo*, [0040]) the request

(1) based on whether the application is in the foreground and (2) based on a power control state. (Bims, ¶86.)

f. 1[d(2)]: “such that, when the power control state is a first power control state and the first end-user application is classified as not interacting with the user, the Internet access request is disallowed, and”

Rao/Araujo renders obvious element 1[d(2)]. (Bims, ¶87.) As explained above, just as *Araujo* describes disallowing a non-VIP application from using the network card to access the network when the network card is off, *Rao/Araujo*’s augmented policies would disallow a background application from accessing the network when *Rao*’s network interface is powered off. (*Id.*) Thus, when *Rao/Araujo*’s network interface is powered off (“first power control state”) and the application requesting network access is in the background (“not interacting with the user”), *Rao/Araujo*’s remote access client (implemented to include *Araujo*’s request deflector functionality) would disallow (e.g., “block,” *Araujo*, [0040]) the request for network access. (Bims, ¶87; Sections V.A.3, V.A.4.e (claim 1[d(1)]); *see also Araujo*, [0038].) *Araujo* illustrates multiple options if a power state change is not allowed:

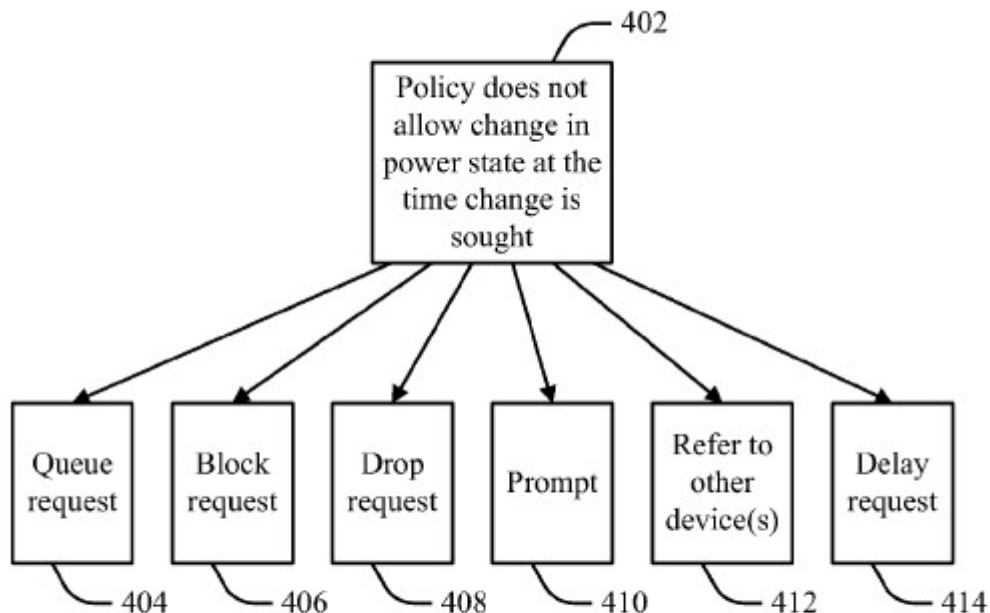


FIG. 4

(*Araujo*, Fig. 4.) A POSITA would have understood that the operations of *Rao*’s remote access client are undertaken by *Rao/Araujo*’s CPU (“one or more processors”). (Bims, ¶87.)

g. 1[d(3)]: “when the power control state is at least one other power control state, the Internet access request is allowed.”

Rao/Araujo renders obvious element 1[d(3)]. (Bims, ¶88.) As explained above, just as *Araujo* describes allowing a non-VIP application to use the network card to access the network when the network card is on (“at least one other power control state”), *Rao/Araujo*’s augmented policies would allow a background application to access the network when *Rao/Araujo*’s network interface is on. (*Id.*)

Thus, when *Rao/Araujo*'s network interface is powered on, *Rao/Araujo*'s remote access client (implemented to include *Araujo*'s request deflector functionality) would allow the network access request, regardless of whether the application is in the foreground or the background. (*Id.*; Sections V.A.3, V.A.4.e (claim 1[d(1)]); *Araujo*, [0037].) *Araujo* shows various actions if a power state change is allowed:

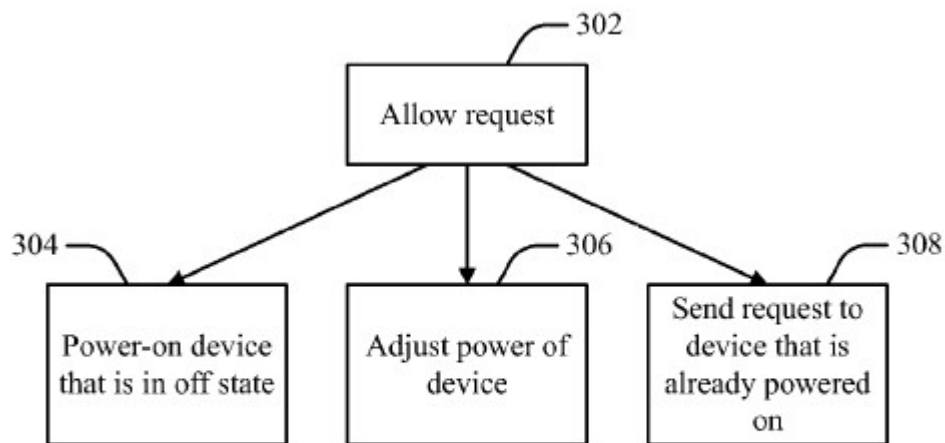


FIG. 3

(*Araujo*, Fig. 3.) A POSITA would have understood that the operations of *Rao/Araujo*'s remote access client are undertaken by *Rao/Araujo*'s CPU ("one or more processors"). (Bims, ¶88.)

5. Claim 2: “The wireless end-user device of claim 1, wherein the one or more processors are configured to classify that the first end-user application is interacting with the user in the device user interface foreground when the user of the device is directly interacting with that application or perceiving any benefit from that application.”

Rao/Araujo renders obvious claim 2. (Bims, ¶¶89-90.) Claim 2 recites an “or” between “directly interacting with that application” and “perceiving any benefit from that application,” so only one of the two limitations must be taught to satisfy claim 2. *Brown v. 3M*, 265 F.3d 1349, 1352-53 (Fed. Cir. 2001).

As discussed in Section V.A.4.d (claim 1[c]), *Rao/Araujo*’s CPU classifies whether a given application is running in a device’s foreground. (Bims, ¶89.) According to *Rao*, an “application is running in the foreground” where it is “in active use by the user.” (*Rao*, [0003].) A POSITA would have understood that “a user of the device is directly interacting with [an] application,” as claimed, when the application is in the foreground and in active use, as disclosed by *Rao*. (Bims, ¶89.) Specifically, *Rao/Araujo*’s CPU determines whether a user is directly interacting with a first application (e.g., by engaging in “active use” of the application, taught by *Rao*) and/or is perceiving any benefit from that application (e.g., by viewing the application as it is running in the foreground and/or perceiving that the application is ready for user input, as also taught by *Rao*). (*Rao*, [0003]; Bims, ¶90.)

6. Claim 5: “The wireless end-user device of claim 1, further comprising, when the Internet access request is disallowed, queuing the Internet access request until a power control state change occurs.”

Rao/Araujo renders obvious claim 5. (Bims, ¶¶91-94.) As discussed in Sections V.A.3 and V.A.4.e (claim 1[d(1)]), in an example implementation, *Rao/Araujo*’s remote access client would disallow certain requests for network access according to *Rao/Araujo*’s augmented policies. (Bims, ¶91.)

Araujo discloses multiple possible actions when a request for network access is disallowed (e.g., initially blocked):

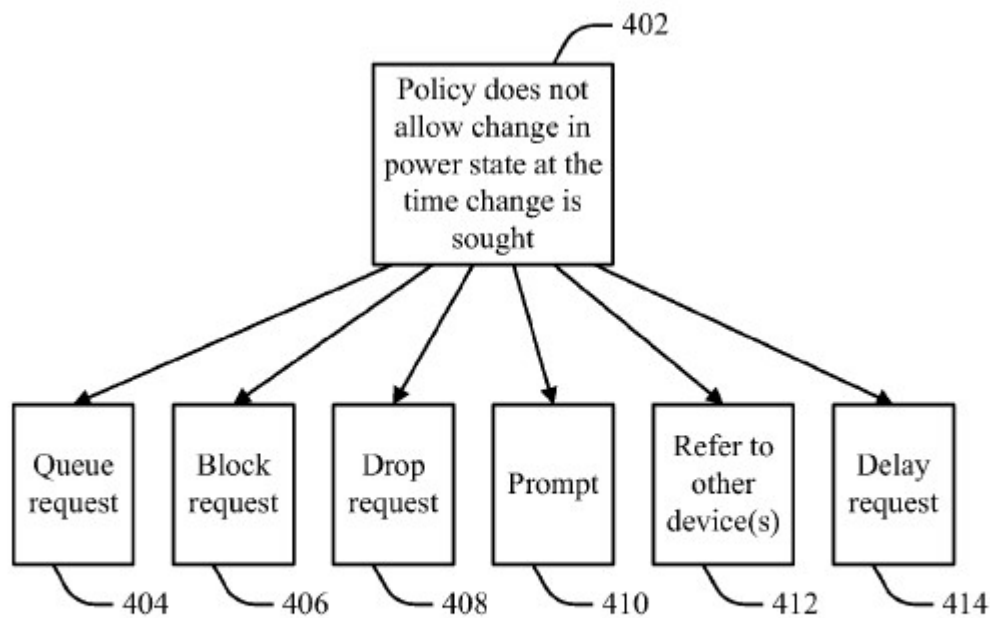


FIG. 4

(*Araujo*, Fig. 4, [0038].) A POSITA would have understood that the actions (illustrated in Figure 4) are not exclusive, and additional actions may be taken

when a request is disallowed. (Bims, ¶92.) “[B]lock[ing]” is one possible action, as discussed above. (*Araujo*, [0040].)

A further “such action is to queue a program’s request to be processed at some later time (block 404).” (*Araujo*, [0039].) For example, if an application “requests to use a device that is turned off (e.g., ... a network card ...), and the power management policy does not allow the device to be turned on at the time of the request, the request may be queued so that it can be serviced at some later [time].” (*Araujo*, [0039], claim 8.) Then, “if at some later time the power management policy allows the device to be turned on, then the program’s request may be dequeued and processed.” (*Araujo*, [0039].) A POSITA would have understood that a change in the power state (e.g., when the network card has been powered on (e.g., due to a request from a VIP application as taught by *Araujo*)) is an example of the sort of event that may lead to a queued request being processed. (Bims, ¶93.)

Similarly, *Araujo* discloses that another potential response to a disallowed request “is to impose a delay on carrying out the request (block 414).” (*Araujo*, [0043].) In some situations, the power management policy “might call for the network card to be turned on at specific intervals.” (*Id.*) “If a request to transmit data over a network is made between these intervals, the request may be delayed until the next time that the network card is scheduled to be powered on, at which

time the request could be allowed.” (*Id.*) The result is that, when the Internet access request is denied, *Rao/Araujo* also discloses delaying the request (“queuing”) via *Rao/Araujo*’s request deflector functionality until the device turns back on (“a power control state change occurs”). (Bims, ¶94.)

7. Claim 7: “The wireless end-user device of claim 1, wherein the one or more processors are further configured to allow the Internet access request when the power control state is the first power control state and the first end-user application is classified as interacting with the user.”

Rao/Araujo renders obvious claim 7. As explained above in Section V.A.3, *Rao*’s policies, when augmented to consider the device’s power state, would specify whether an application request is to be allowed or blocked based both on whether the application is currently in the device’s foreground (taught by *Rao*, and similar to *Araujo*’s VIP status) and the device’s power state. Just as *Araujo* describes allowing a VIP application to use the network card to access the network even when the network card is off, *Rao/Araujo*’s augmented policies would allow a foreground application (akin to *Araujo*’s VIP application) to access the network even when *Rao/Araujo*’s network interface is off (“the first power control state”). (Bims, ¶95.) Thus, when the application requesting network access is in the foreground (“interacting with the user”) and *Rao/Araujo*’s network interface is off (“the first power control state”), *Rao/Araujo*’s remote access client would allow the request for network access. (*Id.*)

8. Claim 8: “The wireless end-user device of claim 1, further comprising a user interface, wherein the user interface is to inform the user of the device when there are options to set, control, override, or modify service usage controls that affect the processor configuration that determines whether to allow or disallow Internet access requests in the first power control state.”

Rao/Araujo renders obvious claim 8. *Rao* discloses “agent 326 may provide a configuration mechanism such as a user interface, graphical or otherwise, design[ed] and constructed for configuring or specifying the policies 520.” (*Rao*, [0183].)

Araujo likewise discloses a user interface. (*Araujo*, [0041].) According to *Araujo*, when an application cannot access the Internet (due to the device’s power state), the CPU may “[p]rompt[] a user (or other person) for input.” (*Araujo*, [0041].) “[I]f a program requests to send data over a network card that has been turned off, a dialog box may be generated on a display that informs the person in front of the display that the card is off.” (*Id.*) A “dialog box” is a user interface, as claimed, because it allows interaction with a user. (Bims, ¶99.) *Araujo* further discloses that this dialog box “may also solicit input from the person (e.g., ‘Would you like to proceed anyway?’), or may simply inform the person that the card is off and suggest that the user try again later.” (*Araujo*, [0041].)

A POSITA augmenting *Rao*’s policies to include *Araujo*’s power management policy would have likewise incorporated *Araujo*’s user interface

functionality into *Rao*'s user interface to enable full use of *Araujo*'s power management policy. (Bims, ¶¶96-97.) Given *Rao*'s architecture already includes user interface functionality, such a modification would be well within the abilities of a POSITA, and a POSITA would have a reasonable expectation of success. (Bims, ¶97.)

A POSITA would have recognized that a user interface for specifying *Rao/Araujo*'s augmented policies would include “options to set, control, override, or modify service usage controls that affect the processor configuration that determines whether to allow or disallow Internet access requests in the first power control state” because any changes to the augmented policies would affect the processor configuration that determines whether to allow or disallow Internet access requests in the first power control state. (Bims, ¶98.)

Moreover, a POSITA would have been motivated and found it obvious to implement the user interface to “inform the user of the device when there are options” related to *Araujo*'s policy, because *Araujo* discloses a user interface that prompts the user for input on its policies. (*Araujo*, [0041].) A POSITA would have understood that soliciting user input, including asking whether the user would like to override the prevailing policy, would entail informing the user of an option to override the service usage control, as claimed. (Bims, ¶99.)

9. Claim 9

a. 9[a]: “The wireless end-user device of claim 1, further comprising: a wireless local area network (WLAN) modem to communicate data for Internet service activities between the device and at least one WLAN, when configured for and connected to the at least one WLAN,”

Rao/Araujo renders obvious element 9[a]. (Bims, ¶¶100-102.) As discussed in Section V.A.4.b (claim 1[a]), *Rao/Araujo*’s device includes a wireless wide area network (WWAN) modem to communicate data for Internet service activities. (Bims, ¶100.) The device “can connect to one or more networks 104.” (*Rao*, [0095]; *see also Rao*, [0125].) Thus, in addition to a WWAN modem (identified for claim 1), a POSITA would have understood that *Rao/Araujo*’s device would include other modems, including a WLAN modem (i.e., a second modem), to connect to other networks. (Bims, ¶100.)

Indeed, *Rao* discloses network interface 118 enables connectivity to a “Local Area Network (LAN) ... through a variety of connections,” such as “wireless connections.” (*Rao*, [0125]; *see also Rao*, [0095].) Network interface 118 includes a “modem ... suitable for interfacing the computing device 102 to any type of network capable of communication,” including WLAN. (*Rao*, [0125].) Given *Rao* discloses a network interface, including a second modem, that enables connectivity to a wireless Local Area Network (WLAN), *Rao/Araujo* discloses a “[WLAN] modem,” as claimed. (Bims, ¶¶101-102.)

b. 9[b]: “wherein the one or more processors are further configured to allow the Internet access request when the WLAN modem provides data communication for the Internet access request and the power control state is the first power control state.”

Rao/Araujo renders obvious element 9[b]. (Bims, ¶¶103-106.) As discussed in Section V.A.4 (claim 1), in *Rao/Araujo*, the CPU controls the remote access client to handle the Internet access requests according to *Rao*’s policies, augmented to include *Araujo*’s power management policy. (Bims, ¶103.) From *Araujo*, a POSITA would have been motivated to implement *Rao/Araujo*’s augmented policies to specify that, when *Rao*’s WWAN network interface card is off (“first power control state”) and an application’s request is disallowed for the WWAN network interface, the request may be referred to *Rao*’s WLAN network interface card for transmission. (*Id.*)

Indeed, as explained in Section V.A.6 (claim 5), *Araujo* discloses multiple possible actions when a request for network access is disallowed. (*Araujo*, [0038], Fig. 4.) One possible action is referring the request to another device:

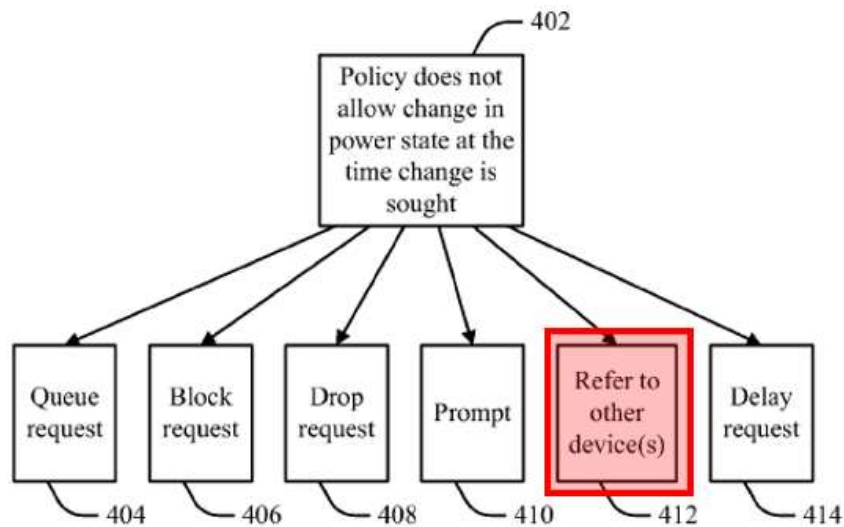


FIG. 4

(*Araujo*, Fig. 4 (annotated), [0042].) According to *Araujo*, “[t]here may be a table, or some other data structure, that lists various devices that could service a request if a particular device is not powered on,” and “[s]uch a table could be consulted.” (*Id.*) For example, *Araujo* explains, some mobile devices (like *Rao*’s device) may have “plural network cards,” and if a request to use a first network card (e.g., a WWAN modem) is denied because the card is turned off, then the request may be sent to a second network card (e.g., a WLAN modem). (*Id.*; Bims, ¶104.)

In evaluating an Internet access request, a POSITA would have understood that a WLAN modem may generally consume less power than a WWAN modem and would be preferred for Internet access in power conservation modes as part of a device’s power management policy. (Bims, ¶105.) Thus, in *Rao/Araujo*, a

background application's Internet access request via WWAN may be denied if the WWAN network card is powered off, but the power management policy may allow that same request via WLAN. (Bims, ¶¶105-106.)

10. Claim 10: “The wireless end-user device of claim 1, wherein the power control state is a power state of the device.”

Rao/Araujo renders obvious claim 10. (Bims, ¶¶107-110.) As explained in Section V.A.4 (claim 1), a POSITA would have been motivated and found it obvious from *Araujo* to augment *Rao/Araujo*'s policies to allow or disallow an application's request for network access (“Internet access request”) (1) based on whether an application is in the foreground (“interacting with a user”) and (2) based on a power control state, such as the power state of a network card, which is a power state of the device. (Bims, ¶107.)

To the extent “power state of the device” refers to the power state of the entire mobile device (rather than a single component), a POSITA would have understood from *Araujo* that the power management policies may implicate the device and/or a component of the device that employs power. (Bims, ¶108.) As *Araujo* explains, “[e]lectronic devices (e.g., computers, wireless telephones, audio/video equipment, etc.) normally employ some type of power management framework.” (*Araujo*, [0014]; *see also Araujo*, [0003] (“[p]ower management for a machine”).) For example, *Araujo* describes a “lower-power-consumption mode”

that applies to various components of the device (“a first power control state”).

(*Araujo*, [0014].) A POSITA would have understood that a device would likewise include a non-lower-power consumption mode (“at least one other power control state”). (Bims, ¶108.)

A POSITA, when reviewing *Araujo*, also would have read *Araujo*’s disclosure in the context of the prior art that formed a POSITA’s background knowledge. See *Ariosa Diagnostics v. Verinata Health, Inc.*, 805 F.3d 1359, 1365 (Fed. Cir. 2015). This would include, for example, *Cole*, which teaches that a “power state”—the same phrase used in *Araujo*—applies to a mobile device (rather than just a component, such as a modem). (Bims, ¶109; *Cole*, [0090]; *Araujo*, [0004].) Like *Araujo*, *Cole* teaches “chang[ing] the configuration of the communication functions in direct response to the power state of the mobile device 110.” (*Cole*, [0090].) As a result, a POSITA would have understood that in *Rao*’s policies, augmented to include *Araujo*’s power management policy, a power state (a “power control state”) includes a power state of the device. (Bims, ¶¶109-110.)

11. Claim 11: “The wireless end-user device of claim 1, wherein the first power control state is a power save state of the device.”

Rao/Araujo renders obvious claim 11. (Bims, ¶¶111-112.) As discussed in Section V.A.10 (claim 10), *Rao/Araujo*’s policies allow or disallow an

application's request for network access based, in part, on a power control state, such as the power state of a network card, which is a power state of the device. (Bims, ¶111.) As discussed in Section V.A.4.f (claim 1[d(2)]), the "first power control state" is when *Rao/Araujo*'s network interface is powered off. A POSITA would have understood that being powered off is a "power save state" of *Rao/Araujo*'s device because the device consumes less power when the network interface is powered off. (*Id.*)

In addition, *Araujo* discloses a "lower-power-consumption mode" in which a device (i.e., the entire mobile device) turns down "brightness" and "processor speed when its remaining battery power drops below a certain threshold." (*Araujo*, [0014].) A POSITA would have understood that this power state is a "power save state" because the amount of power consumed is less than during standard operations. (Bims, ¶112; *see also Cole*, [0099] (describing a "reduced power state" or "low power mode").) Thus, a POSITA would have understood that, in the context of *Rao/Araujo*'s augmented policies, a power save state ("the first power control state") is a power state of the device, as claimed. (Bims, ¶112.)

12. Claim 12: "The wireless end-user device of claim 1, wherein the power control state is a power state of the WWAN modem."

Rao/Araujo renders obvious claim 12. (Bims, ¶113.) As discussed in Section V.A.4.b (claim 1[a]), *Rao/Araujo*'s device includes a WWAN modem, as

claimed. (*Id.*) *Araujo* discloses that “power may be controlled for devices,” such as “disks, network adapters, processors, [and] monitors.” (*Araujo*, [0018], [0019].) A POSITA would have understood that a WWAN modem is an example of a network adapter. (Bims, ¶113.) A WWAN modem may be “on or off” or “an intermediate power usage between fully-on and fully-off.” (*Araujo*, [0019]; Bims, ¶113.) *Araujo* details how its power management policy is driven by the power state (the claimed “power control state”) of the WWAN modem. (*Araujo*, [0004], [0015], [0018]-[0022].) Thus, in the context of *Rao/Araujo*’s architecture, the power control state used to make the policy decisions is the WWAN modem’s power state. (Bims, ¶113.)

13. Claim 13: “The wireless end-user device of claim 1, wherein the one or more processors are further configured to, based on a device usage state, dynamically change a determination of whether to allow or disallow the Internet access request in the first power control state when the first end-user application is classified as not interacting with the user.”

Rao/Araujo renders obvious claim 13. (Bims, ¶¶114-117.) Applying *Rao/Araujo*’s augmented policies, *Rao/Araujo*’s remote access client would allow or disallow an application’s request (1) based on whether the application is in the foreground and (2) based on a power control state. (Bims, ¶114.) *Rao* further discloses that its remote access client “may determine any other characteristics or statistics of the application 338a-338n [for the purpose of assessing priority] such

as ... total execution time[] and/or frequency of use.” (*Rao*, [0188]-[0189].) A POSITA would understand these characteristics reflect a “device usage state” because they reflect how the device is being used. (Bims, ¶114.) A POSITA would therefore have been motivated to modify how *Rao/Araujo* assesses priority (to account for a device usage state) and would have had a reasonable expectation of success in doing so. (Bims, ¶¶115-117.)

From *Rao/Araujo*, a POSITA would understand that, based on these “device usage state” characteristics, *Rao*’s remote access client may dynamically change its determination of whether to allow or disallow an application’s request. (Bims, ¶117.) For instance, when the application requesting network access is in the background (“not interacting with the user”) and the network interface is powered off (“first power control state”), *Rao/Araujo*’s remote access client may determine that a request from the application for network access should be disallowed in light of how the device is being used (e.g., a total execution time or frequency of use does not exceed a predetermined threshold). (*Id.*) But *Rao/Araujo*’s remote access client may dynamically change this determination (e.g., to allow the application’s request) if how the device is being used changes (e.g., the total execution time or frequency of use now exceeds a predetermined threshold). (Bims, ¶117; Sections V.A.3, V.A.4.e (claim 1[d(1)]).) A POSITA would have understood that

the operations of *Rao/Araujo*'s remote access client are undertaken by

Rao/Araujo's CPU ("one or more processors"). (Bims, ¶117.)

14. Claim 14: "The wireless end-user device of claim 1, wherein the one or more processors are further configured to, based on power control state changes for the WWAN modem, dynamically change a determination of whether to allow or disallow the Internet access request in the first power control state when the first end-user application is classified as not interacting with the user."

Rao/Araujo renders obvious claim 14. (Bims, ¶¶118-120.) As discussed in Section V.A.4 (claim 1), *Rao/Araujo*'s CPU controls the remote access client to handle Internet access requests according to *Rao*'s policies, augmented to include *Araujo*'s power management policy. (Bims, ¶118.) From *Araujo*, a POSITA would have been motivated and found it obvious to implement *Rao/Araujo*'s augmented policies to specify that, where a first network card such as a WLAN modem is turned off, an Internet access request from a background application ("not interacting with the user") may be allowed or disallowed based on whether a second, WWAN modem is turned on. (*Id.*)

As explained in Section V.A.9 (claim 9), *Araujo* discloses multiple possible actions when a request for network access is disallowed. (*Araujo*, [0038], Fig. 4.) One possible action is referring the request to another device. (*Araujo*, [0042], Fig. 4.) For example, *Araujo* explains, some mobile devices (like *Rao*'s device) may have "plural network cards," and if a request to use a first network card (e.g.,

a WLAN modem) is denied because the card is turned off, then the request may be sent to a second modem (e.g., a WWAN modem). (*Id.*; Bims, ¶119.) If the WWAN modem is on, the request would be allowed because the application may access the Internet. (Bims, ¶119.)

Thus, from *Rao/Araujo*, a POSITA would have understood that, based on power control state changes for the WWAN modem, *Rao*'s remote access client may dynamically change its determination of whether to allow or disallow an application's request (1) based on whether the application is in the foreground and (2) based on a power control state. (Bims, ¶120.) For instance, when the application requesting network access is in the background ("not interacting with the user") and the WLAN modem is powered off ("first power control state"), *Rao/Araujo*'s remote access client may determine that a request from the application for network access should be disallowed if the WWAN modem is also powered off. (*Id.*) But *Rao/Araujo*'s remote access client may dynamically change this determination (e.g., to allow the application's request via the WWAN modem) if the WWAN modem is powered on. (*Id.*)

15. Claim 15: “The wireless end-user device of claim 1, wherein, when the Internet access request is disallowed, the one or more processors are configured to prevent the Internet access request from causing a change to a power state of the modem.”

Rao/Araujo renders obvious claim 15. (Bims, ¶¶121-122.) As discussed in Section V.A.4.f (claim 1[d(2)]), when *Rao/Araujo*’s network interface is powered off and the application requesting network access is in the background, *Rao/Araujo*’s remote access client would disallow (e.g., “block,” *Araujo*, [0040]) the request for network access. *Araujo* explains that such disallowing involves preventing the request for network access from causing a change to the network card. According to *Araujo*, “if a particular request does not justify powering a device under the power management policy, the request deflector may prevent the request from being passed to the device driver.” (*Araujo*, [0005]; *see also Araujo*, Fig. 2 (208, 212), Fig. 4.) The result is that “the change of a device’s power state [(the modem’s power state)] may be physically prevented or deferred.” (*Araujo*, [0050].) Indeed, as *Araujo* details to explain its power management policy, “state transitions” between power states entail “power usage,” and preventing such a state change preserves power. (*Araujo*, [0029].) Accordingly, in *Rao/Araujo*, when a request is disallowed, *Rao/Araujo*’s remote access client (controlled by the CPU) would prevent the request from causing a change to a power state of the network interface. (Bims, ¶¶121-122.)

16. Claim 16: “The wireless end-user device of claim 1, wherein the one or more processors are configured to, when the Internet access request is disallowed, instruct the first end-user application to transition to a different state.”

Rao/Araujo renders obvious claim 16. (Bims, ¶¶123-126.) *Araujo* discloses that “if the power state change is not allowable under the policy,” the device can “block the request” and “notif[y] [the program] that the request will not be processed.” (*Araujo*, [0038], [0040].) For example, “a program could be notified that a request to use a device is not being put through to the device because the request would be contrary to power-usage policy” (*Araujo*, [0017]) or because “the device is off” (*Araujo*, [0040]). As a result, “the program may receive an error code or result code indicating that the request [will not be put through].” (*Id.*)

In view of *Araujo*, a POSITA would have found it obvious that, when a request is blocked, *Rao/Araujo*’s remote access client (controlled by *Rao/Araujo*’s CPU) instructs an application to transition to a different state. (Bims, ¶124.) Indeed, *Araujo* details an example where a notification would be beneficial. According to *Araujo*, “if an application with a low status is requesting use [of] a network card that is presently turned off, the system might decide ... to wait until a high-status daemon makes a request to use the network card.” (*Araujo*, [0033].) “If such a request is expected (e.g., based on past patterns of usage) within, say, the next two seconds, then the low-status application’s request may be ... delayed in

view of the anticipated high-status daemon's request.” (*Id.*) From *Araujo*, a POSITA would have understood that, when an application request is disallowed, *Rao/Araujo*'s remote access client would instruct the application to move into an idle state and such an instruction would prevent the low-status application from making repeated requests or otherwise unnecessarily consuming device resources. (Bims, ¶125.) This idle state would only be temporary because the idled application would have its request allowed once the high-status daemon requests Internet access. (*Id.*)

As a result, a POSITA would have been motivated by *Araujo* to implement *Rao/Araujo*'s CPU to control *Rao/Araujo*'s remote access client to instruct an application to transition to an idle state where that application's Internet access request has been disallowed. (Bims, ¶126.) This modification would have been well within the abilities of a POSITA, and would merely require enabling *Rao/Araujo*'s remote access client to transmit a code (e.g., error code) to the requesting application indicating that the application's Internet access request calls for a device that is not turned on and instructing the application to enter a standby mode. (*Id.*) As a result, a POSITA would have a reasonable expectation of success in implementing this modification. (*Id.*)

B. Ground 2: Claim 3 Is Obvious over *Rao* in View of *Araujo* and *Singh*

1. Overview of *Singh*

Singh is directed to “displaying windows on a graphical user interface based on relative priorities.” (*Singh*, Abstract.) Like the ’076 patent, *Singh* recognizes that “[c]omputers and other electronic devices are typically operable to execute multiple applications concurrently.” (*Singh*, 1:21-22.) *Singh* is directed to addressing issues raised by a device’s ability to execute multiple applications concurrently and is analogous art to the ’076 patent. (Bims, ¶¶127-128.)

According to *Singh*, an issue with the “simultaneous display of multiple windows” on the GUI is that “[t]he display area of the graphical user interface available to [the] windows is limited.” (*Singh*, 1:21-28, 1:37-40.) Thus, “it is desirable to position, size, and/or configure the windows on the GUI in order to maximize the benefit of the display area of the GUI for the user.” (*Singh*, 1:40-42.) *Singh* thus teaches using the “relative priorities” of windows for arranging the windows on the GUI. (*Singh*, Abstract; Bims, ¶128.) “A higher priority window may cover a lower priority portion of a lower priority window to provide for a simultaneous viewing of the higher priority window and a higher priority portion of the lower priority window.” (*Singh*, Abstract.)

2. Rationale to Combine

It would have been obvious to a POSITA to modify *Rao*'s remote access client to include *Singh*'s windows management function. (Bims, ¶¶129-133.)

Rao describes a scenario where “an application is running in the foreground and currently in active use by the user, [and] a network packet generated or received for an application running in the background may be processed ahead of a network packet generated or received for the application running in the foreground.” (*Rao*, [0003].) According to *Rao*, this scenario is not desirable because “network packets for [background] applications may be processed ahead of the real-time network packets of [a foreground] VoIP telephone call[,] thereby increasing the latency and reducing the quality of the voice communication.” (*Id.*) To prevent network activity from background applications from interfering with network activity from foreground applications, *Rao* teaches policies that “define prioritization based on whether an application is running in the foreground or the background of the client” and “the network packets are communicated according to the determined priority.” (*Rao*, [0182], [0184].)

A POSITA would have recognized that, in cases where multiple windows are displayed simultaneously (e.g., with the highest-priority window having “focus” and covering portions of lower-priority windows), it is challenging for *Rao*'s device to prioritize based on determining which application is running in the

foreground. (Bims, ¶131.) In such cases, multiple windows are in the foreground but partially displayed and one window has the focus (and thus should have the highest priority). (*Id.*) For example, in the window with focus, a user is engaged in a video call that requires real-time communication. (*Id.*) Another foreground window displays the user's email, but the user interacts with email only occasionally (e.g., during pauses in the video call). (*Id.*) While both windows are in the foreground, network traffic related to the video call should receive priority when the video call has focus. (*Id.*) Thus, a POSITA implementing *Rao/Araujo* would have been motivated to look to other references to improve the prioritization of Internet access requests when multiple applications are displayed in a user interface of the device. (*Id.*)

A POSITA would have recognized “the multitude of ways to define client-side application priorities” and further recognized that *Singh*'s window priorities represent such an example. (*Rao*, [0182]; Bims, ¶132.) *Rao/Araujo*'s policies would thus be further augmented to permit the remote access client to classify whether an application is interacting with the user in the device user interface foreground using *Singh*'s window prioritization. (*Rao*, [0182], [0183]; *Singh*, 4:52-55, Abstract; Bims, ¶132.)

A POSITA would have a reasonable expectation of success in doing so, which would have required only routine programming knowledge well within the

skill of a POSITA before the priority date. (Bims, ¶133.) Indeed, the change would have been nothing more than the use of a known technique to improve similar mobile devices in a similar way and combining prior-art elements according to known methods to yield the predictable results described above. (*Id.*)

Rao/Araujo/Singh's CPU would be configured to perform the operations discussed herein. (*Id.*)

3. Claim 3: “The wireless end-user device of claim 1, wherein the one or more processors are configured to classify that the first end-user application is interacting with the user in the device user interface foreground based on a state of user interface priority for the application.”

Rao/Araujo/Singh renders obvious claim 3. (Bims, ¶¶134-135.) As discussed in Section V.A.4.d (claim 1[c]), *Rao/Araujo*'s CPU classifies “whether an application is running in the foreground or the background.” (*Rao*, [0182].)

Rao/Araujo's CPU would further use *Singh*'s windows management function to improve its classification of foreground applications where multiple windows are displayed simultaneously. (Bims, ¶¶134-135; *see* Section V.B.2.)

Singh teaches a “window management function [that] determine[s] a relationship of windows with one another.” (*Singh*, 9:19-21.) According to *Singh*, “when a user interacts with a window, the window becomes the highest priority window.” (*Singh*, 9:33-38.) A POSITA would have known that users interact with foreground windows, and in *Singh*, users only interact with the highest priority

window (because any interaction causes the window to be assigned the highest priority). (Bims, ¶135.) Therefore, a POSITA would have understood that *Singh*’s window management function—integrated into *Rao/Araujo*’s remote access client—uses a state of user interface priority (e.g., highest priority) to classify that the application is interacting with the user in the user interface foreground. (*Id.*)

C. Ground 3: Claim 4 Is Obvious over *Rao* in View of *Araujo* and *Freund*

1. Overview of *Freund*

Freund relates to using “access rules” for efficient use of computing resources and controlling network access. (*Freund*, 3:60-67.) *Freund* is analogous art to the ’076 patent. (Bims, ¶¶136-138.)

Freund teaches application-specific access rules to determine if an application or process “should have access to the Internet and what kind of access (i.e., protocols, Internet addresses, time limitations, and the like) is permissible.” (*Freund*, 4:45-63.) These “[a]ccess rules can include criteria such as total time a user can be connected to the Internet,” “time a user can interactively use the Internet,” and “a list of applications or application versions that a user can or cannot use in order to access the Internet.” (*Freund*, Abstract, 13:2-13.)

To apply the rules, *Freund* gives “[s]pecific consideration[] for detecting interactive or ‘active’ use by the user versus background or ‘passive’ use by certain

applications.” (*Freund*, 8:63-67.) *Freund* teaches the use of “a preferred user interface or ‘wizard’ dialogs for configuring rules.” (*Freund*, 24:16-17.) This interface displays “rules governing operation of the Internet access monitor” and allows a user to create, edit, or delete rules. (*Freund*, 24:17-24.)

2. Rationale to Combine

It would have been obvious to a POSITA to modify *Rao/Araujo*’s remote access client to further include *Freund*’s method for classifying whether an application is running in the background, to improve the operation of *Rao*’s remote access client. (Bims, ¶¶139-144.)

Rao contrasts “an application [that] is running in the foreground and currently in active use by the user” with “an application running in the background.” (*Rao*, [0003]; Bims, ¶140.) *Rao*, however, does not expressly describe how to determine whether an application is running in the background. (Bims, ¶140.) As a result, a POSITA implementing *Rao*’s teachings would have looked to similar references, including *Freund*, to understand how to classify an application as running in the background. (*Id.*)

Freund teaches network access policies that give consideration to whether an application is operating in a device’s foreground or background. (*Freund*, 10:5-43.) In particular, *Freund* teaches that “‘background’ use” should be

determined, at least in part, by evaluating “when an application accesses the Internet without user interaction.” (*Freund*, 3:12-16.)

Freund discloses a number of benefits to its classification method. (Bims, ¶142.) According to *Freund*, its method “for detecting interactive or ‘active’ use by the user versus background or ‘passive’ use by certain applications” “prevents the system from mistakenly blocking access of a user based on excessive use, when in fact a background application is instead responsible for the activity.” (*Freund*, 8:63-9:3.) Indeed, according to *Freund*, “[c]onventional technologies cannot distinguish between this ‘active’ versus ‘passive’ access and therefore cannot present an accurate picture of the actual time the user has spent on on-line activities.” (*Freund*, 10:30-33.)

The resulting combination would further *Rao*’s goal of prioritizing the transmission of network packets from foreground applications over background applications to reduce the detrimental latency impacts that result from a failure to prioritize. (E.g., *Rao*, [0003]; see also Section V.A.1.) *Freund*’s method allows for a more accurate distinction between background and foreground applications, including certain “bandwidth-intensive [background] applications” (*Freund*, 2:12-14) that should be de-prioritized relative to certain applications operating in the foreground (e.g., a VoIP application providing a telephone call). (Bims, ¶143.) As a result, a POSITA would have been motivated to integrate *Freund*’s method

for classifying whether an application is running in the background into

Rao/Araujo's remote access client. (*Id.*)

A POSITA would have also had a reasonable expectation of success in modifying *Rao/Araujo*'s architecture in view of *Freund*'s classification method.

(Bims, ¶144.) *Rao/Araujo*'s remote access client already evaluates whether a device is in active use (e.g., to classify whether an application is running in the foreground), and *Freund*'s classification method adds additional functionality for classifying whether an application is running in the background. (*Id.*) Such a modification would have been well within the abilities of an ordinary artisan to implement and would not have conflicted with the classification of applications performed by *Rao/Araujo*'s remote access client. (*Id.*)

3. Claim 4: “The wireless end-user device of claim 1, wherein the one or more processors are configured to classify that the first end-user application is not interacting with the user in the device user interface foreground when the application is providing or utilizing a background data service.”

Rao/Araujo/Freund renders obvious claim 4. (Bims, ¶¶145-147.) As discussed in Section V.A.4.d (claim 1[c]), *Rao/Araujo*'s CPU, via the remote access client, classifies whether a given application is running in a device's foreground or background. (Bims, ¶145.) *Rao/Araujo*'s remote access client would implement *Freund*'s method for determining when an application is running in the background. (*Id.*; Section V.C.2.)

Freund defines “‘background’ use” as “when an application accesses the Internet without user interaction.” (*Freund*, 3:12-16.) *Freund* further explains that background use “occurs when an application or process executing in the background (i.e., does not have ‘focus’) accesses the Internet, such as a mail client which intermittently polls an Internet-based mail server.” (*Freund*, 10:10-24; *see also Freund*, 10:40-43 (“A given application itself can be examined for determining whether it is ‘active’ by determining whether the application receives ‘focus’ and/or receives user input (e.g., mouse clicks or key strokes).”).) Since *Freund* defines background use as occurring when an application or process executing in the background accesses the Internet (when an application uses background data services), then by *Freund*’s definition, an application would be classified as not interacting with a user when the application uses background data services. (Bims, ¶146.)

A POSITA would have thus understood that when *Rao/Araujo*’s remote access client (controlled by its CPU) classifies a first end-user application as running in the background, as disclosed by *Freund*, the classification would be performed based on determining that the application is providing or utilizing a background data service (e.g., by generating network packets in the background). (Bims, ¶147.)

D. Ground 4: Claim 6 Is Obvious over *Rao* in View of *Araujo* and *Aleksic*

1. Overview of *Aleksic*

Like the '076 patent, *Aleksic* teaches monitoring and prioritizing usage of device resources on an application-by-application basis. (*See Aleksic*, [0022]-[0025], [0033].) *Aleksic* is analogous art to the '076 patent. (Bims, ¶¶148-149.)

Aleksic recognizes that when a portable device has limited device resources, “users are typically most concerned with ... the ability to use the portable device for at least an emergency call” (i.e., a voice application) or other critical applications. (*Aleksic*, [0002], [0050].) *Aleksic* thus teaches a controller that allows a user to “prioritize which functional processing capability features of the portable device [e.g., which applications] take priority over others.” (*Aleksic*, [0022].) *Aleksic* teaches “a graphic user interface 1100 that provides selectable priority information shown as ‘critical’ or ‘noncritical’ for various [device] applications.” (*Aleksic*, [0050].) Thus, according to *Aleksic*, if the available device resources are not sufficient “in comparison to an emergency call threshold, the device will enter the emergency call mode and effectively shut down all applications to reserve remaining power for voice communications (e.g., emergency calls)” or other critical applications, leaving those applications to run in either the background or foreground, as necessary. (*Aleksic*, [0043].)

2. Rationale to Combine

It would have been obvious to a POSITA to modify *Rao/Araujo*'s remote access client to include *Aleksic*'s prioritization of critical or emergency applications and further augment *Rao/Araujo*'s policies. (Bims, ¶¶150-153.)

The prior art recognized “a need ... for an improved power management operation in portable devices.” (*Aleksic*, [0004].) As described in Section V.C.3, a POSITA would have modified *Rao* in view of *Araujo* to further improve *Rao*'s application-aware power management. (See *Rao*, [0179]-[0195].) In implementing this modification, a POSITA would have recognized opportunities to provide further prioritization for applications designated by a device user as “critical” or “emergency” so that these applications would be available, even when device resources are limited. (Bims, ¶151; *Aleksic*, [0043], [0048].) This would have motivated a POSITA to seek solutions beyond *Rao* and *Araujo* to achieve such additional efficiencies and further augment *Rao*'s policies. (Bims, ¶151.)

Aleksic's teachings allow a user to “prioritize which functional processing capability features of the portable device [e.g., which applications] take priority over others.” (*Aleksic*, [0022], [0050].) This includes emergency or other critical applications, requiring prioritization of Internet access. (Bims, ¶152.) *Aleksic*'s prioritization would have enabled *Rao/Araujo*'s prioritization architecture to

further account—at a policy level—for “critical” and “emergency” applications.

(*Aleksic*, [0043], [0048].)

A POSITA would have had a reasonable expectation of success in modifying *Rao/Araujo*’s policies in view of *Aleksic*. (Bims, ¶153.) *Rao* discloses that a POSITA “will recognize and appreciate the multitude of ways to define client-side application priorities.” (*Rao*, [0182].) *Rao* explains that policies: may (1) “be provided by or downloaded via the gateway 340,” (2) “comprise any type and format of syntax and/or language for specifying a policy,” and (3) “be provided via any type and/or form of medium.” (*Rao*, [0183].) These policies also “may be configurable by a user.” (*Id.*) The result is that the modification would have been well within the abilities of an ordinary artisan to implement. (Bims, ¶153.)

3. Claim 6: “The wireless end-user device of claim 1, wherein the one or more processors are further configured to, for at least a second application or service, allow Internet access requests in the first power control state without regard to whether the second application or service is classified as interacting with the user.”

Rao/Araujo/Aleksic renders obvious claim 6. (Bims, ¶154.) As discussed in Section V.D.2, *Rao/Araujo/Aleksic*’s remote access client’s augmented policies allow for prioritizing certain critical applications regardless of the power control state of *Rao*’s network interface and whether the application is running in the background (“not interacting with the user”). Thus, when *Rao/Araujo/Aleksic*’s

network interface is powered off and a critical or emergency application requests network access, *Rao/Araujo/Aleksic*'s remote access client would allow an Internet access request regardless of whether the critical or emergency application is in the background ("not interacting with the user"). (*Id.*; Sections V.A.3, V.A.4.e (claim 1[d(1)]).) A POSITA would have understood that the operations of *Rao/Araujo/Aleksic*'s remote access client are undertaken by its CPU ("one or more processors"). (Bims, ¶154.)

E. Ground 5: Claim 16 Is Obvious over *Rao* in View of *Araujo* and *Montemurro*

1. Overview of *Montemurro*

Like the '076 patent, *Montemurro* teaches application-aware optimization of network transmissions from a mobile device. Just as *Rao* and *Araujo* teach application-aware prioritization of network access, *Montemurro* discloses application-aware "policy-based routing of communications among two or more modes of wireless communication." (*Montemurro*, [0001].) *Montemurro* is analogous art to the '076 patent. (Bims, ¶¶155-158.)

Devices with such connectivity capabilities are "dual or multi-mode devices," with "radio access technologies that provide access to" multiple network types, including WLAN and WWAN. (*Montemurro*, [0003].) But, according to *Montemurro*, "[t]here are costs associated with application access from these

different network[s],” and “[i]t is therefore desirable to have a mechanism that seeks to optimize communications” over the different network types.

(*Montemurro*, [0004].)

To optimize communications, *Montemurro* discloses “a rules engine ... to coordinate communications for applications 204 using communications interfaces [208].” (*Montemurro*, [0027].) This rules engine employs “a set of rules/policies that could include various factors such as radio access technology (e.g.,] for high/low bandwidth properties), cost, presence, time of day, location (e.g.,] geo-based policies, network roaming), destination IP address, application type, and Quality of Service (QoS) requirements.” (*Montemurro*, [0029].)

2. Rationale to Combine

As described in Section V.A.3, a POSITA would have modified *Rao* in view of *Araujo* to further improve *Rao*’s architecture to include application-aware power management. (*See Rao*, [0179]-[0195].) In implementing this modification, a POSITA would have recognized opportunities to provide further optimization in scenarios where *Rao/Araujo*’s device has access to multiple networks. (Bims, ¶159; *Rao*, [0095]; *see also Montemurro*, [0003], [0004].) A POSITA would have been motivated to seek solutions beyond *Rao* and *Araujo* to achieve additional efficiencies. (Bims, ¶¶159-162.)

According to *Montemurro*, devices with multiple radio access technologies can access more than one network, including at least “wireless local area networks (WLAN),” “wireless wide area networks (WWAN),” “GSM,” and “GPRS.” (*Montemurro*, [0002]-[0003].) But “[t]here are costs associated with application access from these different networks.” (*Montemurro*, [0004].) And “[t]here are also quality considerations.” (*Id.*) “It is therefore desirable to have a mechanism that seeks to optimize communications for multi-mode capable devices.” (*Id.*)

Given *Rao*’s disclosure of multi-mode devices that “connect to one or more networks” (*Rao*, [0095]), a POSITA would have been motivated to improve *Rao/Araujo*’s architecture to include *Montemurro*’s policy-based data routing for multimode operations. (Bims, ¶161.) A POSITA would have understood that incorporating these teachings into *Rao/Araujo* would have resulted in the ability to “offer different usage models depending on the mode of wireless operation selected.” (*Montemurro*, [0003].) *Montemurro*’s policies would have enabled *Rao*’s prioritization architecture to further account—at a policy level—for “radio access technology ..., cost, presence, time of day, location ..., destination IP address, application type, and Quality of Service (QoS) requirements.” (*Montemurro*, [0029].)

A POSITA would have had a reasonable expectation of success in modifying *Rao/Araujo*’s policies in view of *Montemurro*’s policies. (Bims, ¶162.)

Rao discloses that a POSITA “will recognize and appreciate the multitude of ways to define client-side application priorities.” (*Rao*, [0182].) *Rao* explains that policies: may (1) “be provided by or downloaded via the gateway 340,” (2) “comprise any type and format of syntax and/or language for specifying a policy,” and (3) “be provided via any type and/or form of medium.” (*Rao*, [0183].) These policies also “may be configurable by a user.” (*Id.*) The result is that the modification would have been well within the abilities of an ordinary artisan to implement. (Bims, ¶162.)

3. Claim 16: “The wireless end-user device of claim 1, wherein the one or more processors are configured to, when the Internet access request is disallowed, instruct the first end-user application to transition to a different state.”

Rao/Araujo/Montemurro renders obvious claim 16. (Bims, ¶¶163-165.)

Montemurro discloses that “applications can ... provide different usage models that may vary depending on the mode of operation that is available.” (*Montemurro*, [0024]; *see also Montemurro*, [0003].) The rules engine “configures the communication operations with a set of rules/policies that could include various factors such as radio access technology ..., cost, presence, time of day, location ..., destination IP address, application type, and Quality of Service (QoS) requirements, among others.” (*Montemurro*, [0029].) As discussed in

Section V.A.4.c-g (claim 1[b]-1[d(3)]), in *Rao/Araujo*'s remote access client, policies are administered via the CPU.

Montemurro teaches that a processor can, when an Internet access request is disallowed (e.g., because the available network for the network request is not allowable under the policy), instruct the application to transition to a different state by connecting to a different network and/or altering its behavior to account for the different network. (Bims, ¶164.) For example, “[a] natural language example of a policy rule could be ‘the MP3 download application 204D runs on the lowest cost network 104 and 106 available’.” (*Montemurro*, [0029].) Thus, if the MP3 download application is running on a WWAN network when a lower cost WLAN network becomes available, the MP3 download application would be directed to transition to a different state of using the WLAN network. (Bims, ¶164.)

Moreover, *Montemurro* discloses that, where connection “differences are significant (e.g., between GSM/GPRS and WAN 802.11 technologies),” the application may transition to a different state of behavior by “tailoring its behaviour to account for the new access technology in use,” such as changing “[b]uffer sizes, retransmission timers[,] etc.” (*Id.*) A POSITA would have recognized that these behavior changes are examples of transitions between different states, and these transitions can occur in response to a disallowed Internet access request (where a more suitable network becomes available). (Bims, ¶165.)

VI. The Petition Is Proper Under 35 U.S.C. § 325(d)

Discretionary denial under § 325(d) is not warranted. *See Advanced Bionics, LLC v. MED-EL Elektromedizinische Geräte GmbH*, IPR2019-01469, Paper 6, 7-9 (PTAB Feb. 13, 2020) (precedential) (“*Advanced Bionics*”) (citing *Becton, Dickinson, & Co. v. B. Braun Melsungen AG*, IPR2017-01586, Paper 8 (PTAB Dec. 15, 2017) and applying *Becton, Dickinson* factors (a)-(f) within the *Advanced Bionics* analysis).

Montemurro (issued as U.S. Patent No. 8,825,109) and a relative of *Rao* (WO 2006/012610A2) were each cited during prosecution of the ’076 patent. (’076 patent, References Cited.) Patent Owner submitted these references on an information disclosure statement, and they are among the over 1,200 references listed on the patent. (*Id.*) There is no evidence the Examiner substantively considered these references during prosecution. (*See* Ex-1002.) Simply citing a reference on an information disclosure statement, with nothing more, does not favor denial of institution. *See Navistar, Inc. v. Fatigue Fracture Tech., LLC*, IPR2018-00853, Paper 13, 16-17 (PTAB Sept. 12, 2018). Moreover, the same or substantially the same arguments were not previously presented to the Office given the Examiner issued no prior-art rejections.

To the extent it is argued that the same or substantially the same art or arguments were previously presented to the Office, the Examiner erred in a manner

material to the patentability of the claims. *Advanced Bionics*, 8-10. Although *Montemurro* and a relative of *Rao* were cited on the '076 patent's face, none were addressed during prosecution. As this Petition shows, the Examiner failed to appreciate *Montemurro*'s or *Rao*'s disclosures and how those references in combination with other prior-art references not previously before the Office (e.g., *Araujo*, *Singh*, or *Freund*) render the claims obvious.

Moreover, Petitioners have shown a reasonable likelihood that at least one of the challenged claims is unpatentable over the applied art based on the current record. Petitioners have demonstrated material error by the Office, and discretionary denial is not warranted.

VII. The Petition Is Proper Under *Fintiv*

The factors set forth in *Apple Inc. v. Fintiv Inc.*, IPR2020-00019, Paper 11, 5-6 (PTAB Mar. 20, 2020) (precedential), weigh in favor of instituting review under 35 U.S.C. § 314(a). Here, there is a parallel district court proceeding involving the same parties and patent. However, no trial date has been set. Moreover, the claim construction process has just begun, the district court has not analyzed any prior-art invalidity issues, and discovery is in the preliminary stages. (Ex-1022.)

The Petition's merits are compelling and clear evidence of the unpatentability of the claims. Where the Board "determines that the information

presented at the institution stage presents a compelling unpatentability challenge, that determination alone demonstrates that the PTAB should not discretionarily deny institution under *Fintiv*.” USPTO Memorandum, Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation 4-5 (June 21, 2022). Because the factors favor instituting review, the Board should institute under § 314(a).

VIII. Mandatory Notices

A. Real Party-in-Interest

The real parties-in-interest for this Petition are Lenovo (United States) Inc. (“Lenovo US”) and Motorola Mobility LLC (“Motorola”). Petitioners further identify Lenovo Group Ltd. as a real party-in-interest solely because Lenovo Group Ltd. was previously named as a defendant in the parallel district court case listed below.

B. Related Matters

Patent Owner has asserted the ’076 patent against Lenovo US and Motorola in *Headwater Research LLC v. Motorola Mobility LLC, et al.*, No. 4:23-cv-04496-JST (N.D. Cal.) (filed Aug. 30, 2023).

The ’076 patent is also related to U.S. Patent No. 10,749,700, which is the subject of an *inter partes* review petition filed concurrently with this Petition.

C. Lead and Back-Up Counsel

<u>Lead Counsel</u>	<u>Back-Up Counsel</u>
Joshua L. Goldberg Reg. No. 59,369 joshua.goldberg@finnegan.com Finnegan, Henderson, Farabow, Garrett & Dunner, LLP 901 New York Avenue, NW Washington, DC 20001-4413 Tel: 202-408-6092 Fax: 202-408-4400	Benjamin A. Saidman Reg. No. 69,325 benjamin.saidman@finnegan.com Finnegan, Henderson, Farabow, Garrett & Dunner, LLP 271 17th Street, NW, Suite 1400 Atlanta, GA 30363-6209 Tel: 404-653-6510 Fax: 404-653-6444 Daniel C. Cooley (Reg. No. 59,639) daniel.cooley@finnegan.com Finnegan, Henderson, Farabow, Garrett & Dunner, LLP 1875 Explorer Street Suite 800 Reston, VA 20190-6023 Tel: 571-203-2700 Fax: 202-408-4400 Grace K. Mills (Reg. No. 66,946) gracie.mills@finnegan.com Finnegan, Henderson, Farabow, Garrett & Dunner, LLP 901 New York Avenue, NW Washington, DC 20001-4413 Tel: 202-408-4000 Fax: 202-408-4400

D. Service Information

Petitioners consent to being served electronically at the e-mail addresses provided above and at Motorola-Headwater-IPRs@finnegan.com.

IX. Grounds for Standing

Petitioners certify that the '076 patent is available for IPR and that
Petitioners are not barred or estopped from requesting IPR challenging the patent
claims on the grounds in this Petition.

X. Conclusion

Petitioners request IPR and cancellation of claims 1-16. The Office may
charge any required fees to Deposit Account No. 06-0916.

Date: August 12, 2024

Respectfully submitted,

By: /Joshua L. Goldberg/
Joshua L. Goldberg, Lead Counsel
Reg. No. 59,369

CERTIFICATION OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24, Petitioners certify that the word count of Petitioners' Petition for *Inter Partes* Review (exclusive of any table of contents, table of authorities, mandatory notices under § 42.8, certificates of service and word count, and list of exhibits or claim listing) as measured by Microsoft Word is 13,993.

Dated: August 12, 2024

By: /Joshua L. Goldberg/
Joshua L. Goldberg, Lead Counsel
Reg. No. 59,369

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. §§ 42.6(e) and 42.105(a), the undersigned certifies that on August 12, 2024, a copy of the foregoing Petition for *Inter Partes* Review, the associated powers of attorney, and Exhibits 1001-1023 were served by FedEx Priority Overnight on the correspondence address of record indicated in the Patent Office's Patent Center website for U.S. Patent No. 9,198,076:

Headwater Research LLC
c/o Farjami & Farjami LLP
26522 La Alameda Ave., Suite 360
Mission Viejo, CA 92691

A courtesy copy has also been mailed to Patent Owner's litigation counsel at:

Mark Fenster
Russ August & Kabat
12424 Wilshire Boulevard, 12th Floor
Los Angeles, CA 90025

Date: August 12, 2024

By: /Lisa C. Hines/
Lisa C. Hines
Case Manager
FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, LLP