

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

STODGE, INC. D/B/A POSTSCRIPT

Petitioner

v.

ATTENTIVE, INC.

(record) Patent Owner

IPR2023-01338
Patent No. 11,416,887

**PETITION FOR INTER PARTES REVIEW
UNDER 35 U.S.C. §§ 311-319 AND 37 C.F.R. § 42.200 ET SEQ.**

TABLE OF CONTENTS

TABLE OF EXHIBITS	5
NOTICE OF LEAD AND BACKUP COUNSEL.....	8
NOTICE OF THE REAL-PARTIES-IN-INTEREST	8
NOTICE OF RELATED MATTERS.....	8
NOTICE OF SERVICE INFORMATION	8
GROUND FOR STANDING	9
STATEMENT OF PRECISE RELIEF REQUESTED.....	9
THRESHOLD REQUIREMENT FOR INTER PARTES REVIEW	9
I. INTRODUCTION	10
A. Technical Background.....	10
B. Claims of the '887 patent.	17
C. Examination History of the '887 patent.	19
II. CLAIM CONSTRUCTION	19
III. DETAILED EXPLANATION OF THE REASONS FOR UNPATENTABILITY	20
Ground 1. Claims 1-2, and 13-14 were obvious over Haggerty in view of Demsey and Khanna.....	20
A. Prior Art Status	20
B. Overview of the Ground.....	20
1. Overview of Haggerty (two-tap method).....	21
2. Overview of Demsey (ad network).....	24
3. Overview of Khanna (mobile app deeplinking)	26
C. Rationale (Motivation) Supporting Obviousness.....	28

D.	<i>Graham</i> Factors.....	32
E.	Reasonable Expectation of Success	32
F.	Analogous Art	33
G.	Claim Mapping.....	33
Ground 2.	Claims 1-4, 6-10, 13-14, and 19 were obvious as in Ground 1, in further view of Rakavy.....	54
A.	Prior Art Status.....	55
B.	Overview of the Ground and Rationale for the Combination.....	55
C.	Reasonable Expectation of Success	56
D.	Analogous Art	56
E.	Claim Mapping.....	56
Ground 3.	Claims 1-2, 5, and 13-18 were obvious as in Ground 1, in further view of <i>Ad Serving Technology</i>	62
A.	Prior Art Status.....	62
B.	Overview of the Ground.....	62
C.	Reasonable Expectation of Success	63
D.	Analogous Art	63
E.	Claim Mapping.....	64
Ground 4.	Claims 1-2, 11-14, and 20 were obvious as in Ground 1, in further view of Molinet.	67
A.	Prior Art Status.....	67
B.	Overview of the Ground.....	68
C.	Overview of Molinet	68
D.	Reasonable Expectation of Success	69

E.	Analogous Art	69
F.	Claim Mapping.....	69
IV.	SECONDARY CONSIDERATIONS	73
V.	DISCRETIONARY INSTITUTION.....	73
A.	The Board should not deny the petition under 35 U.S.C. §325(d)	73
B.	The Board should not deny the petition under 35 U.S.C. §314(a).....	74
VI.	CONCLUSION.....	76
	CERTIFICATE OF SERVICE	77
	CERTIFICATE OF WORD COUNT	78

TABLE OF EXHIBITS

Exhibit No.	Description
1001	U.S. Patent No. 11,416,887 (“ the ’887 patent ”).
1002	Declaration of Dr. Henry Houh.
1003	C.V. of Dr. Henry Houh.
1004	File History of U.S. App. Ser. No. 17/669,114 (issued as the ’887 patent).
1005	U.S. Pat. App. Pub. 2014/0379482 (“ Demsey ”).
1006	U.S. Pat. App. Pub. 2013/0111328 (“ Khanna ”).
1007	U.S. Pat. No. 8,532,283 (“ Haggerty ”).
1008	U.S. Pat. App. Ser. No. 13/461,541 (“ the Haggerty ’541 application ”).
1009	U.S. Pat. App. Pub. 2016/0142858 (“ Molinet ”).
1010	U.S. Prov. App. Ser. No. 62/079,512 (“ The Molinet ’512 Provisional ”).
1011	U.S. Pat. App. Pub. 2015/0178784 (“ Oliver ”).
1012	Cristal, G., “Ad Serving Technology, Understand the marketing revelation that commercialized the internet”, ISBN: 1484867572 (2014)(“ Ad Serving Technology ”).
1013	U.S. Pat. App. Pub. 2015/0142568 (“ Hsu ”).
1014	Plaintiff’s Answering Brief in Opposition to Defendant’s Motion to Dismiss Complaint Pursuant to Fed. R. Civ. P. 12(B)(6) in <i>Attentive Mobile Inc. v. Stodge Inc. d/b/a Postscript</i> , Case, No. 1-23-cv-00087 (D. Del., Filed March. 15, 2023).
1015	File History of U.S. App. Ser. No. 15/986,569.

1016	Firtman, M., <i>“Programming the Mobile Web, Second Edition”</i> , O’Reilly (2013).
1017	U.S. Pat. App. Pub. 2009/0247140 (“ Gupta ”).
1018	U.S. Pat. No. 5,913,040 (“ Rakavy ”).
1019	RFC 5724, “URI Scheme for Global System for Mobile Communications (GSM) Short Message Service (SMS)”, January 2010.
1020	WO 2005/062596A1 (“ Helkio ”).
1021	Flanagan, D. <i>“JavaScript: The Definitive Guide, Sixth Edition”</i> , O’Reilly Media, Inc. (2011).
1022	Complaint for Patent Infringement in <i>Attentive Mobile Inc. v. Stodge Inc. d/b/a Postscript</i> , Case No. 1-23-cv-00087 (D. Del. Filed Jan. 25, 2023).
1023	Scheiner, M., <i>Attentive Gets the Message Out</i> , available at https://crm.org/news/attentive-gets-the-message-out
1024	Levine, B., <i>Led by TapCommerce founders, startup Attentive launches SMS/MMS marketing platform</i> , available at https://martech.org/led-tapcommerce-founders-startup-attentive-launches-smsmms-marketing-platform/
1025	Ha, A, <i>TapCommerce’s founders are back with Attentive</i> , a messaging startup that’s raised \$13M, available at https://techcrunch.com/2018/02/08/attentive-launch/
1026	United States District Courts — National Judicial Caseload Profile, available at https://www.uscourts.gov/sites/default/files/data_tables/fcms_na_distprofile0630.2023.pdf
1027	Scheduling Order in <i>Attentive Mobile Inc. v. Stodge Inc. d/b/a Postscript</i> , Case No. 1-23-cv-00087 (D. Del. March 8, 2023).
1028	Mobile Marketing Association, “US Consumer Best Practices

	for Messaging”, version 7.0, (2012), available at https://www.mmaglobal.com/documents/us-consumer-best-practices .
1029	U.S. Pat. No. 7,136,875 (“ Anderson ”).
1030	Excerpt from Amazon.com available at https://www.amazon.com/Serving-Technology-Understand-revelation-commercialized/dp/1484867572/
1031	Excerpt from Google Books available at https://www.google.com/books/edition/Ad_Serving_Technology/PHNHnwEACAAJ?hl=en

Petitioner respectfully requests *inter partes* review under 35 U.S.C. §311 of claims 1-20 of U.S. Pat. No. 11,416,887 (“the ’887 patent”).

NOTICE OF LEAD AND BACKUP COUNSEL

Lead Counsel Matthew A. Smith Reg. No. 49,003 SMITH BALUCH LLP 700 Pennsylvania Ave. SE, 2 nd Floor Washington, DC 20003 (202) 669-6207 smith@smithbaluch.com	Backup Counsel Andrew S. Baluch Reg. No. 57,503 SMITH BALUCH LLP 700 Pennsylvania Ave. SE, 2 nd Floor Washington, DC 20003 (202) 880-2397 baluch@smithbaluch.com
--	---

NOTICE OF THE REAL-PARTIES-IN-INTEREST

The real-party-in-interest for this petition is Stodge, Inc. d/b/a Postscript.

NOTICE OF RELATED MATTERS

The ’887 patent has been asserted in the following litigations:

- *Attentive Mobile Inc. v. 317 Labs, Inc. d/b/a Emotive*, Case No. 1-22-cv-01163 (D. Del. Filed Sep. 01, 2022).
- *Attentive Mobile Inc. v. Stodge Inc. d/b/a Postscript*, Case, No. 1-23-cv-00087 (D. Del. Filed Jan. 25, 2023).

NOTICE OF SERVICE INFORMATION

Please address all correspondence to the lead counsel at the addresses shown above. Petitioner consents to electronic service by email at: smith@smithbaluch.com, baluch@smithbaluch.com.

GROUND FOR STANDING

Petitioner hereby certifies that the patent for which review is sought is available for *inter partes* review, and that the Petitioner is not barred or estopped from requesting an *inter partes* review on the grounds identified in the petition.

STATEMENT OF PRECISE RELIEF REQUESTED

Petitioner respectfully requests that claims 1-20 of the '887 patent be canceled based on the following Grounds:

Ground 1: Claims 1-2, and 13-14 were obvious over Haggerty in view of Demsey and Khanna.

Ground 2: Claims 1-4, 6-10, 13-14, and 19 were obvious as in Ground 1, in further view of Rakavy.

Ground 3: Claims 1-2, 5, and 13-18 were obvious as in Ground 1, in further view of *Ad Serving Technology*.

Ground 4: Claims 1-2, 11-14, and 20 were obvious as in Ground 1, in further view of Molinet.

THRESHOLD REQUIREMENT FOR INTER PARTES REVIEW

As shown in the Grounds set forth below, the information presented in the instant petition, if unrebutted, demonstrates that “it is more likely than not that at least 1 of the claims challenged in the petition is unpatentable.” 35 U.S.C. § 314(a).

I. INTRODUCTION¹

A. Technical Background

The '887 patent is directed to a known business method, implemented using known Internet advertising technology.

The known business method of the '887 patent is simple. By the time the first application leading to the '887 patent was filed (May 2017), smartphones had been in widespread use for almost a decade. (Ex. 1002, ¶23). With the rise of smartphone use, text-based messaging (*e.g.* “SMS” messaging) became a primary form of communication. (Ex. 1002, ¶24). This form of communication was naturally adopted by advertisers, who wanted to send advertising texts to consumers. (Ex. 1002, ¶24).

Advertising texts, however, could be subject to legal or industry restrictions that required an advertiser to both obtain a user’s consent and provide proof of possession of a smartphone prior to sending advertising texts. (Ex 1001, 7:19-29)(Ex. 1002, ¶24). To prove both consent and possession of the phone, it was by 2012 considered a best practice to request, in an advertisement, that a user sign up for text messages by sending a text to the advertiser. (Ex. 1002, ¶¶25-26)(Ex. 1028,

¹ This introduction reflects the understanding of a person having ordinary skill in the art (“PHOSITA”), and is therefore relevant to each of the grounds below.

pp. 008-009, 0015-0021). For example, the 2012 Mobile Marketing Association document *U.S. Consumer Best Practices for Messaging* provided the following example of a best practice for advertising urging a user to text an advertiser:

Call to Action:

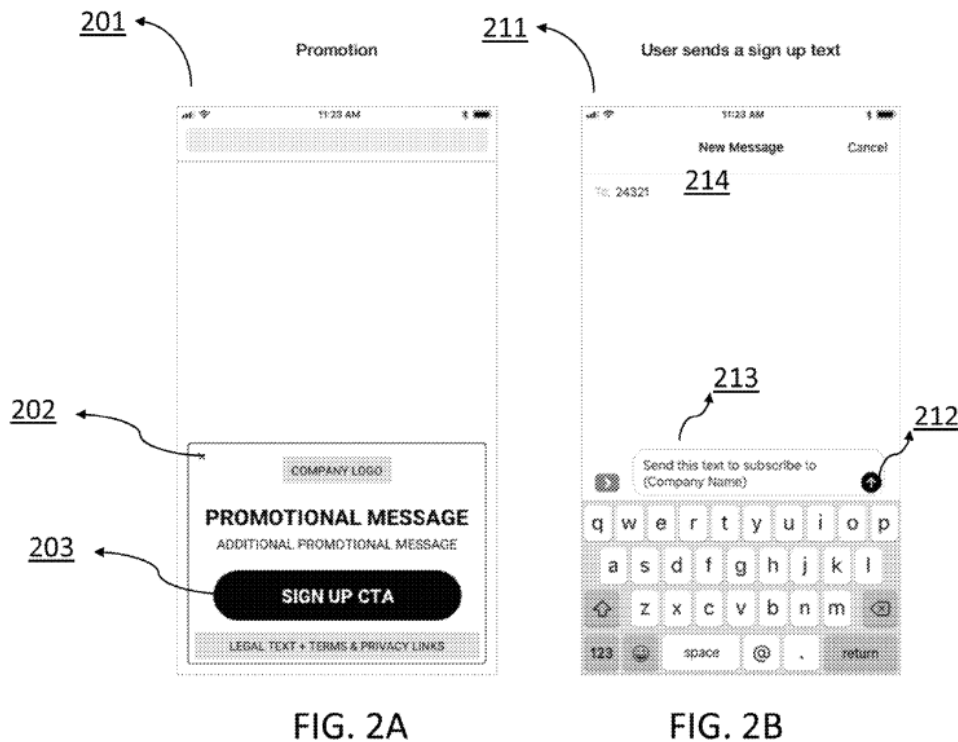
The following is advertised:

Program sponsor →	Upmobile Ski Alerts!
Service Description →	Send us the resort name, we'll send you the snow conditions. Txt 'Mammoth' to 12345 to receive an alert for Mammoth Resort.
Additional Carrier Costs →	Msg&Data Rates May Apply.
Terms & Conditions →	T&Cs avail at www.mammoth.com/mobile .

(Ex. 1028, p. 016)(Ex. 1002, ¶25).

Against a background of such standard marketing practices, the '887 purports to add known technology that allowed a device to *automatically* draft a text message (rather than asking a user to write it). (Ex. 1002, ¶27). The process starts with a Web advertisement that, instead of encouraging a user to type and send a text, encourages a user to click on a *link*. (Ex. 1001, 6:66-7:8)(Ex. 1002, ¶27). Clicking the link on a smartphone causes the device's text-messaging app to open. (Ex. 1001, 6:66-7:2)(Ex. 1002, ¶27). The text-messaging app presents the user with a pre-filled text message. (Ex. 1001, 7:13-17)(Ex. 1002, ¶27). The message provides both a destination telephone number and a message body, such that the user need only click "send" to send the text. (Ex. 1001, 7:17-29)(Ex. 1002, ¶27). When sent (to a designated server), the text message signs a user up for a promotion, such as a further promotional message. (*Id.*).

The process is shown in Figs. 2A-2B of the '887 patent:



(Ex. 1001, Figs. 2A-2B)(Ex. 1002, ¶28). These figures show an ad with a link 203 (Fig. 2A) that when clicked opens a pre-populated text message (Fig. 2B). (Ex. 1001, 7:30-65)(Ex. 1002, ¶28). The user need only click the send button (212) to send the text and sign up to receive marketing texts. (*Id.*). Thus, the user only needs to take *two actions* to sign up: clicking the link and then clicking the “send” button. (Ex. 1001, 6:66-8:5)(Ex. 1002, ¶28). Because only two actions are required, the '887 patent refers to this approach as a “two-tap” approach. (Ex. 1001, 3:3-7, 7:25-30)(Ex. 1002, ¶28).

The “two-tap” approach, according to the Patent Owner, was a “true leap” forward in the technology. (Ex. 1014, p. 2)(Ex. 1002, ¶29). Prior art technologies

were allegedly too cumbersome: requiring detailed sign-up forms or requiring the user to accurately type out a text message and respond to a confirming text. (*Id.*)(Ex. 1001, 1:36-47). The “two-tap” approach avoided potential mistakes in typing a text message and provided a record of consent with a single message. (Ex. 1014, p. 2).

Whatever the merits of the “two-tap” approach, however, the Patent Owner was not the first to do it. For example, U.S. Pat. No. 8,532,283 to Haggerty (Ex. 1007), the subject of Ground 1, below, teaches that a user can view an ad online with an embedded link. (Ex. 1007, Abstract, 2:28-31)(Ex. 1002, ¶¶30-31, 72-76). Clicking the link will cause the user’s mobile device to present to the user “a pre-populated text message addressed to the destination address with the contents indicated therein” together with “a prompt for confirmation for sending the message”. (Ex. 1007, 10:1-5, 11:19-21, 9:48-55, 13:12-18)(Ex. 1002, ¶31). Haggerty further states that its method provides a convenient way for a user to respond to an advertisement (Ex. 1007, 1:53-62) with lowered risk of mistyping information in the SMS (Ex. 1007, 9:48-55). (Ex. 1002, ¶31).

Other prior art references, including Oliver (Ex. 1011), Hsu (Ex. 1013) and Gupta (Ex. 1017) teach similar two-tap methods. (Ex. 1011, ¶¶0007-0009, 0027-0029, 0034, 0048)(Ex. 1013, ¶¶0039-0042)(Ex. 1017, ¶¶0026, 0069, 0088)(Ex. 1002, ¶¶31-38).

Central to these two-tap references was the ability to create links that could,

when selected, open a pre-populated a text message—technology that had long been available. (Ex. 1012, pp. 584-585)(*e.g.* Ex. 1011, ¶¶0027-0029)(Ex. 1013, ¶0040)(Ex. 1010, pp. 021-022, 015)(Ex. 1002, ¶¶39-44). Generally speaking, these links fell into the category of *mobile-app linking*. (Ex. 1002, ¶¶39-44). Mobile app linking allowed a link in a webpage, viewed on a smartphone browser, to open a different mobile app. (Ex. 1002, ¶39). Such links could simply open the app to the app’s opening screen, but they could also link to locations *within* the app, such as a page describing a specific product. (*Id.*). In that case, such links are called “deeplinks”, because they are directed to a point “deep” within the app, not just the opening screen. (Ex. 1006, ¶0030)(Ex. 1002, ¶39).

Deeplinking technologies for mobile apps were well-established years before 2017, including the use of Uniform Resource Identifiers (“URIs”) in webpage links to open a pre-populated text-message. (*Id.*). As the textbook “*Programming the Mobile Web*” stated in 2013, “[w]e all like the Short Message Service; that’s why **mobile browsers generally offer the ability to invoke the new SMS window from a link**. To do this, we have two possible URI schemes, sms:// and smsto://.” (Ex. 1016, p. 557)(Emphasis added)(Ex. 1002, ¶¶40-43). These URI schemes were used at the start of a link—so just as a standard web URL might begin with http://, a link to a text-messaging app would begin with sms://. (Ex. 1002, ¶40). The ability to use such URI links was built in to the major smartphone operating systems (iOS and

Android) in the relevant timeframe. (Ex. 1002, ¶40)(Ex. 1010, p. 015). These URI schemes were used to pass both the destination phone number and the message body to the text-messaging app (thereby “deeplinking” to the app). (Ex. 1016, pp. 557-558)(Ex. 1013, ¶0040)(Ex. 1011, ¶¶0027-0029)(Ex. 1002, ¶40). An appropriate URI protocol (*e.g.* “sms:”) to open text-messaging applications with pre-populated text and destination numbers was standardized by the year 2010 in IETF RFC 5724. (Ex. 1002, ¶41)(Ex. 1019, pp. 007-008, §2.2, p. 010, §2.5). The URIs specified in RFC 5724 could be used in clickable website links, such that when a user clicked on the link, a pre-populated SMS message would appear in the user’s text-messaging app. (Ex. 1002, ¶42)(Ex. 1020, pp. 009:33-010:33). Numerous prior art references discussed such URIs. (Ex. 1016, pp. 557-558)(Ex. 1013, ¶0040)(Ex. 1011, ¶¶0027-0029)(Ex. 1020, pp. 009:33-010:33)(Ex. 1010, pp. 015, 021)(Ex. 1002, ¶42).

The claims of the ’887 patent also pair the known two-tap business method to standard technology that had been used, *e.g.*, by Internet advertisers. (Ex. 1002, ¶¶45-47). This technology can be explained with a short background. In the early days of the Web, website publishers could contract directly with advertisers, who would provide ads (*e.g.* “banner ads”) to place on a website. (Ex. 1002, ¶45). The ads usually had links, such that a user who clicked on the ad would be directed to the advertiser’s home page. (Ex. 1002, ¶47).

This system had a variety of limitations. First, most website publishers did

not have the expertise to seek out advertisers. (Ex. 1002, ¶47). Only the largest publishers regularly had ads, but even these ads could not be quickly changed, may not have been placed optimally, and could not be targeted to specific users. (Ex. 1002, ¶47). Furthermore, it was difficult to perform accounting tasks like recording the number of views of an ad. (Ex. 1002, ¶48).

To address these concerns, *ad networks* arose. (Ex. 1002, ¶48). Ad networks would contract with website publishers, through a simple, one-time sign-up process, for the right to place ads on the publisher's site. (Ex. 1002, ¶48). To place ads on a publisher site, ad networks would transmit so-called "ad tags" to publishers. (Ex. 1002, ¶48). The ad tags were placeholder segments of code (*e.g.* JavaScript) that could be placed into publisher webpages (Dr. Houh provides an explanation of website technology in Ex. 1002, ¶¶49-62). The purpose of an "ad tag" was to allow the ad network to have some control over the placement of ads *at the time a page was being viewed*. (Ex. 1002, ¶59). The ad tag would execute when the user viewed a webpage, thereby requesting an ad from the ad-network server. (Ex. 1002, ¶59). The ad-network server would choose an ad in realtime and send the ad back for insertion into the webpage. (Ex. 1002, ¶59).

This process was advantageous for several reasons. (Ex. 1002, ¶63). With older technology (no ad tags), an advertiser would only know *which website* would contain the ad, not *which users* would eventually see the ad. (Ex. 1002, ¶63). Having

ad tags that execute *when the user views a page*, however, allowed the ad tag to collect information about the *specific user* and the *user's device*, and send this information to the ad-network server. (Ex. 1002, ¶63). This, in turn, allowed the ad-network server to target an ad directly to the *specific user*, formatted correctly for the specific user's device. (Ex. 1002, ¶63). Furthermore, because the publisher did not need to contract directly with advertisers, but only with the ad network, even small publishers could receive ads appropriate for their sites. (Ex. 1002, ¶63).

The '887 patent uses this standard ad-network technology to serve an ad to a user, where the ad has a clickable link that pre-populates a text message. (Ex. 1002, ¶64). How this technology relates to the claim language will be discussed next.

B. Claims of the '887 patent.

The '887 patent has 20 claims with a single independent claim (claim 1). Claim 1 can be divided into roughly three parts, corresponding to (1) *standard ad-network technology* that uses a script-based ad tag to gather user data to send to a server, (2) *standard deeplinking technology* that provides a way to link to a text-messaging application, and (3) the “two-tap” business method. Claim 1 is shown in the table below, with added numbers (1[a] [1b], etc.):

Claim element	Category
“1[a]. At least one non-transitory processor-readable medium storing code configured to be executed by a processor of a click-to-text server, the code including	Preamble

instructions configured to cause the click-to-text server to:”	
“[1b] send, to a client server, an integration tag configured to be served with a webpage hosted by the client server,”	Ad-network
“[1c] the integration tag configured to cause a mobile device loading the webpage via a first application to send user data to at least one of the client server or the click-to-text server;”	Ad-network
“[1d] define and send a uniform resource identifier (URI) to the mobile device in response to the mobile device executing the integration tag, the URI deeplinking to a messaging application different from the first application and”	Deeplinking
“[1e] configured to cause the mobile device to (1) automatically transition from the first application to the messaging application in response to the mobile device detecting a user interaction with a promotional message associated with the webpage, the messaging application being different from the first application, and”	Deeplinking / Business method
“[1f] (2) automatically populate a custom message in the messaging application that includes an address associated with the click-to-text server and a message body that includes an identifier associated with at least one of the webpage or the user data”	Business method
“[1g] such that, upon detecting a single user interaction with a send button of the messaging application, the mobile device sends the custom message to the click-to-text server;”	Business method
“[1h] receive the custom message at the click-to-text server; and”	Business method
“[1i] enroll the mobile device in a promotion associated with the promotional message based on receiving the custom message and without receiving any additional information from the mobile device.”	Business method

In claim 1, a “click-to-text server” (e.g. the central server of an ad network)

sends an “integration tag” (e.g. an ad tag) to a “client server” (a publisher) for inclusion in a website. The “integration tag” is configured to send user data to the central server, and gets a response. This is standard ad-network technology. (Ex. 1002, ¶66). The response contains a deeplinking URI, which is simply a link in an ad that deeplinks to a mobile app. This is standard deeplinking technology. (Ex. 1002, ¶66). Clicking on the deeplink causes a pre-filled text-message to appear that provides consent for further marketing activities, as was well-known in the art. (Ex. 1002, ¶66).

C. Examination History of the '887 patent.

The claims of the '887 patent issued after a single requirement for restriction. (Ex. 1004). In allowing the claims, the Examiner stated that the closest prior art of record did not “teach or suggest every limitation of the claims of the present invention, and particularly the interaction of the tag and user causing an enrollment with a *de minimis* amount of information having been provided, in combination with the other elements of the claims of the present invention”. (Ex. 1004, p. 041).

II. CLAIM CONSTRUCTION

Petitioner submits that the Board need not construe any claim terms to consider the prior art and arguments presented herein.

III. DETAILED EXPLANATION OF THE REASONS FOR UNPATENTABILITY

Ground 1. Claims 1-2, and 13-14 were obvious over Haggerty in view of Demsey and Khanna.

Claims 1-2, and 13-14 were obvious under (post-AIA) 35 U.S.C. § 103(a) over **Haggerty** (Ex. 1007), in view of U.S. Pat. App. Pub. 2014/0379482 (“**Demsey**”)(Ex. 1005) and U.S. Pat. App. Pub. 2013/0111328 (“**Khanna**”)(Ex. 1006).

None of these references was of record during the prosecution of the application leading to the ’887 patent.

A. Prior Art Status

Haggerty, Demsey, and Khanna and are U.S. patent publications with publication and effective filing dates before May 26, 2017, and are thus prior art under 35 U.S.C. §§102(a)(1) and (2).

B. Overview of the Ground

Haggerty teaches a “two-tap” method for enrolling users in promotions, where a single user action opens a pre-filled text message, which then only needs to be sent. (Ex. 1007, 10:1-5, 9:48-55, 13:12-18)(Ex. 1002, ¶¶66-70). This ground posits that it would have been obvious to implement Haggerty’s business method using well-known ad-serving technology, by adding the techniques of Demsey (an ad-network patent) and Khanna (a mobile app deeplinking patent). (Ex. 1002, ¶¶70-71).

Thus, in the combination, Haggerty teaches the known “two-tap” business method, Demsey provides the standard ad-network functionality, and Khanna provides the standard deeplinking functionality. (Ex. 1002, ¶¶70-71).

1. Overview of Haggerty (two-tap method)

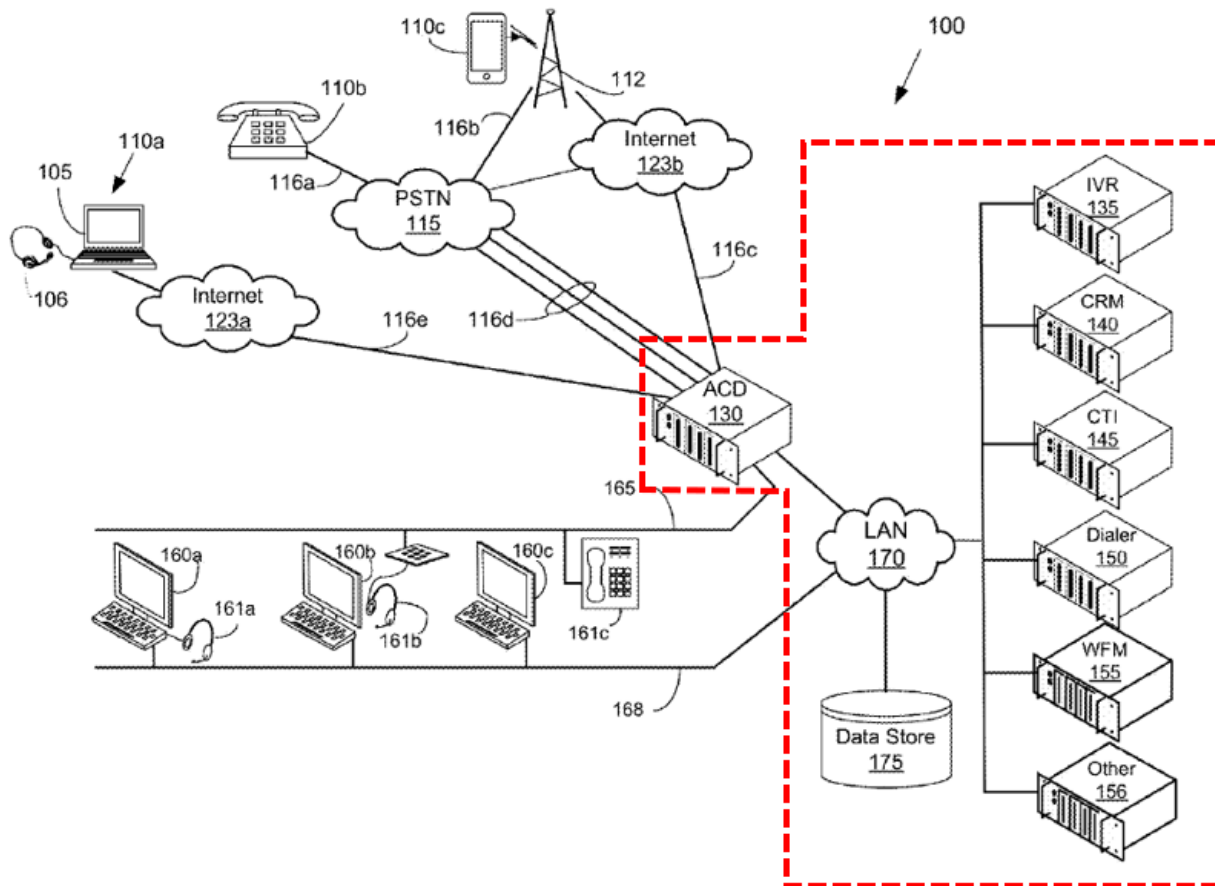
Haggerty teaches a “two-tap” business method similar to that of the ’887 patent. (Ex. 1002, ¶72). In Haggerty, a user can interact with an advertisement. (Ex. 1007, 14:41-49)(Ex. 1002, ¶72). The interaction can be scanning a QR code, but can also be clicking a hyperlink. (Ex. 1007, 14:41-49, *see also* 1:36-62, 2:1-5, 8:51-56, 10:59-11:21, 16:26-31, Figs. 3C, 4G)(Ex. 1002, ¶72). Once clicked, the link can cause the user’s mobile device to pre-populate an SMS message. Haggerty states:

“[U]pon reading the QR code using a smart phone, a **pre-populated text message** addressed to the destination address **with the contents indicated therein** is presented to the user, along with a prompt for **confirmation for sending the message.**”

(Ex. 1007, 10:1-5)(Emphasis added)(Ex. 1002, ¶72). Note that Haggerty teaches that a QR code is not necessary, and that “[i]n general, any of the actions that may be initiated by a QR code **could also be associated with a link.**” (Ex. 1007, 11:19-21)(Emphasis added)(Ex. 1002, ¶72).

The pre-populated text message is sent to a “call center” for processing, as shown in Fig. 4B. (Ex. 1007, Fig. 4B, 12:39-47)(Ex. 1002, ¶73). The “call center”

is not limited to handling or dealing with *voice* calls, however, and Haggerty also considers an SMS text message to be an “SMS call”. (Ex. 1007, Figs. 4B-4C, 4:38-56)(Ex. 1002, ¶73). The call center methods are carried out using a central server system, as shown in the red-dashed box from Fig. 1, reproduced here:



(Ex. 1007, Fig. 1, 3:44-63, 5:45-7:62)(Ex. 1002, ¶¶73-75).

This central server system, upon receiving a SMS message from a user, processes the message according to the particular advertising campaign, and sends a response SMS. (Ex. 1007, 12:39-67)(Ex. 1002, ¶¶71-75). An example message flow is shown in Fig. 4B, reproduced below, with yellow highlighting added to show

the text messages:

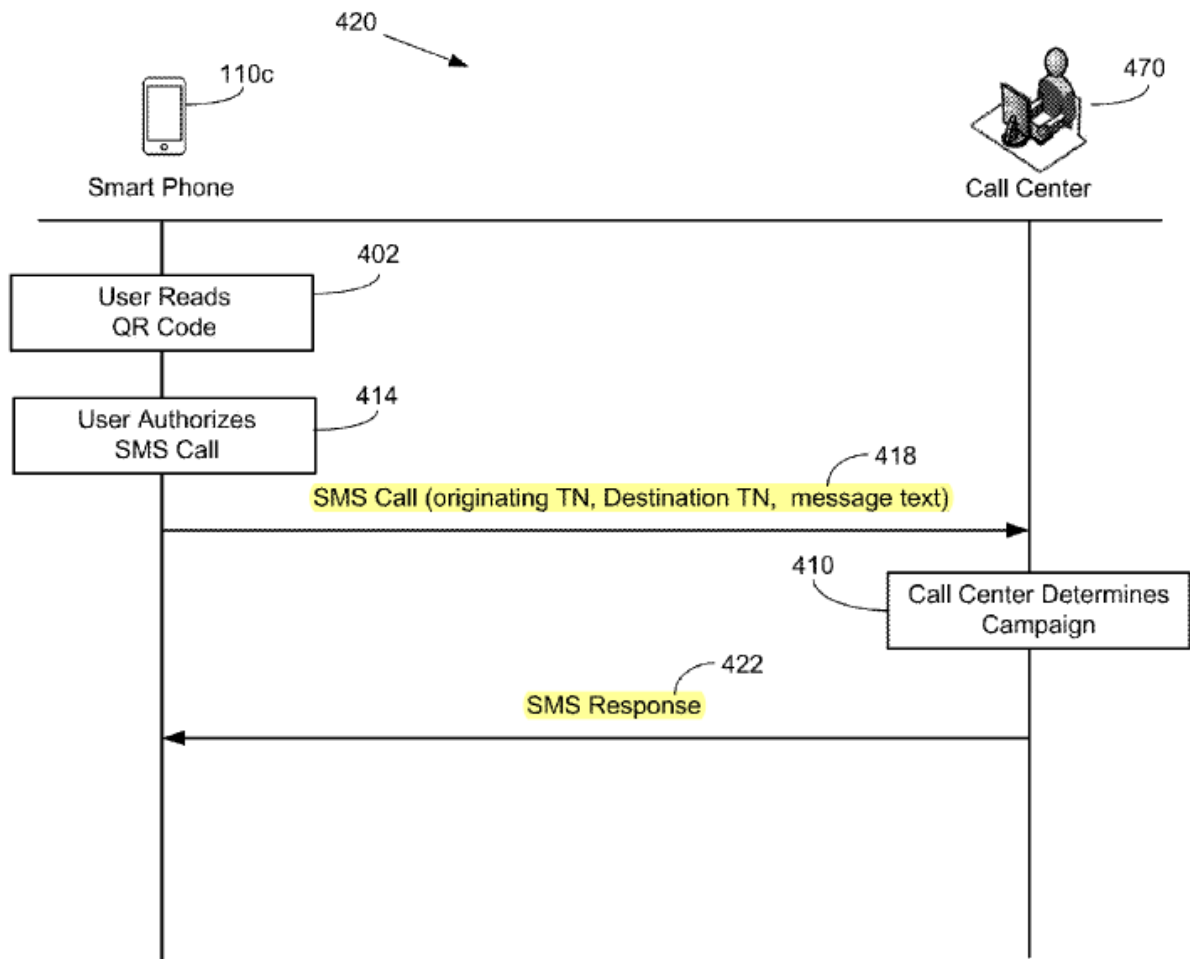


FIG. 4B

(Ex. 1007, Fig. 4B, 12:39-67)(Ex. 1002, ¶73).

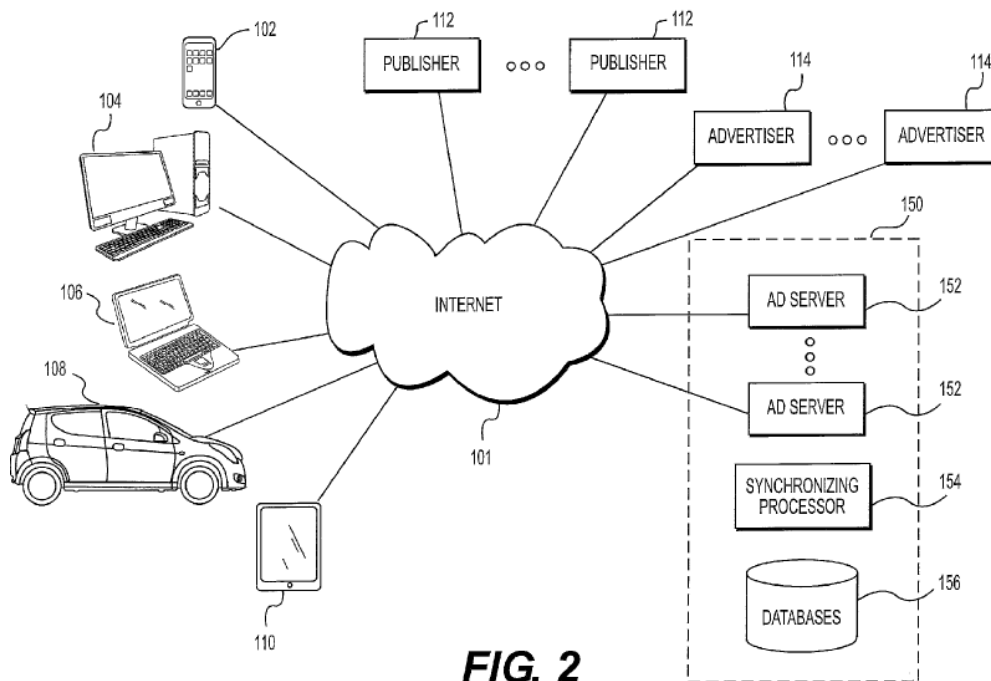
Just like the '887 patent, Haggerty teaches a user response to an advertisement with only two actions: activating a link, and then sending a pre-populated SMS. (Ex. 1007, 10:1-5, 14:41-49, 1:36-62, 2:1-5, 8:51-56, 10:59-11:21, 16:26-31, Figs. 3C, 4G)(Ex. 1002, ¶74). Haggerty further states that its method provides a convenient way for a user to respond to an advertisement (Ex. 1007, 1:53-62) with lowered risk

of mistyping information in the SMS (Ex. 1007, 9:48-55). (Ex. 1002, ¶74).

While Haggerty assumes advertisements with clickable links (Ex. 1007, 2:28-31), it does not describe the technology used to serve those ads. This technology, however, was standard in the relevant timeframe, as described in **Demsey**. (Ex. 1002, ¶76).

2. Overview of Demsey (ad network)

Demsey is an ad-network system that has a central server, advertisers, publishers, and users who download webpages with ads, as shown in Fig. 2, reproduced here:



(Ex. 1005, Fig. 2, ¶¶0024-0025)(Ex. 1002, ¶77). In Fig. 2, the central server is the synchronization server system 150, advertisers are represented by servers 114, publishers are represented by servers 112, and user devices are shown by devices

102 through 110. (Ex. 1005, ¶¶0024-0025)(Ex. 1002, ¶77).

Demsey sends “ad tags” to a publisher for inclusion in webpages. (Ex. 1005, ¶0005)(Ex. 1002, ¶78). When a webpage is downloaded to a user device, the ad tag (which uses JavaScript, a common programming language for webpages) executes and requests that an advertisement be sent back to the user’s device for display. (Ex. 1005, ¶¶0045-0048, Fig. 5)(Ex. 1002, ¶78). This is shown, for example, in Fig. 5 of Demsey, reproduced below with three added red-dashed boxes:

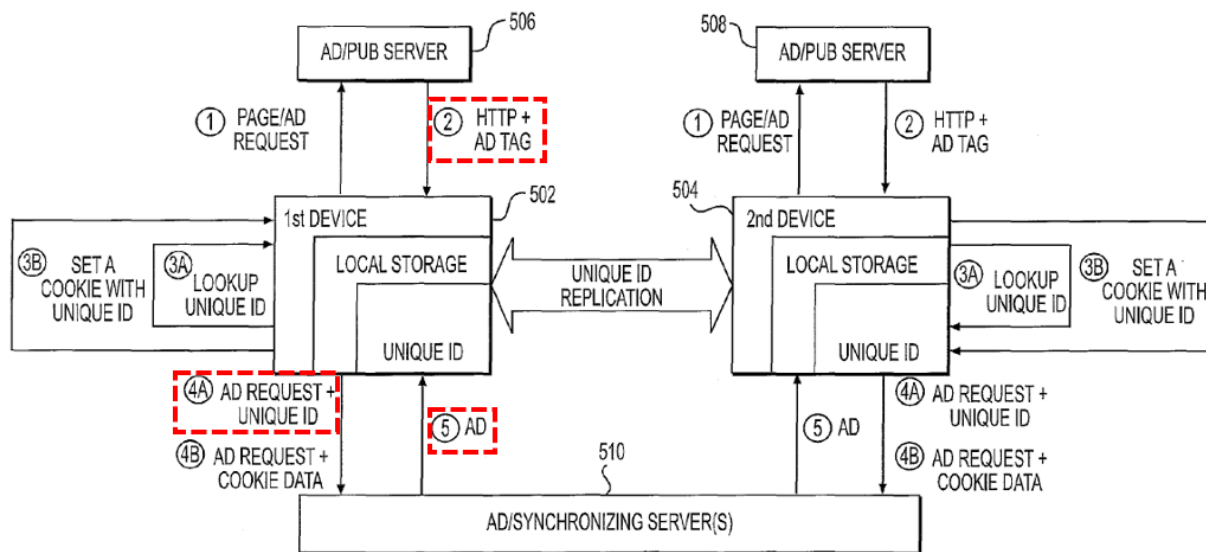


FIG. 5

(Ex. 1005, Fig. 5)(Ex. 1002, ¶78). As shown in Fig. 5, a user device 502 (left side, middle) downloads a webpage at action (2) (shown by a red-dashed box at the top). The webpage contains an “AD TAG”. (Ex. 1005, ¶0045)(Ex. 1002, ¶78). The “AD TAG”, when executed by the user’s device, causes the user’s device 502 to issue an “AD REQUEST” at action (4A) (red-dashed box, lower left) to the

AD/SYNCHRONIZING SERVER(S). (Ex. 1005, ¶0045)(Ex. 1002, ¶78). Along with the ad request, the user's device sends a user ID. (Ex. 1005, ¶0045)(Ex. 1002, ¶78). When the server receives the ad request, the server chooses an ad, and sends the ad back to the user device for incorporation into the webpage at action (5) (red-dashed box). (Ex. 1005, ¶¶0045, 0041)(Ex. 1002, ¶78).

Demsey's particular method involves assigning each user a unique identifier (like a number or code), and having the user device store the identifier in a cookie or other local storage. (Ex. 1005, ¶¶0009, 0039, 0045)(Ex. 1002, ¶79). Upon subsequent viewing of an ad, the unique identifier is retrieved by the ad tag and sent back to a central server, so that the central server can ascertain which user is requesting the ad. (Ex. 1005, ¶0045)(Ex. 1002, ¶79). Demsey also teaches providing "instance identifiers", which are identifiers for a *particular device* of a given user, understanding that a user could view webpages from multiple devices. (Ex. 1005, ¶¶0022, 0036)(Ex. 1002, ¶79).

3. Overview of Khanna (mobile app deeplinking)

A natural extension of Demsey's ad-network is provided by **Khanna**. (Ex. 1002, ¶80). Khanna teaches a way to allow mobile app deeplinking using links in ads that are served by an ad network (like Demsey's). (Ex. 1006, ¶¶0031-0044)(Ex. 1002, ¶80). Like Demsey's system, Khanna's system also has a central server, called the Advertisement Management System 110 ("AMS 110"), shown in Fig. 1,

reproduced here:

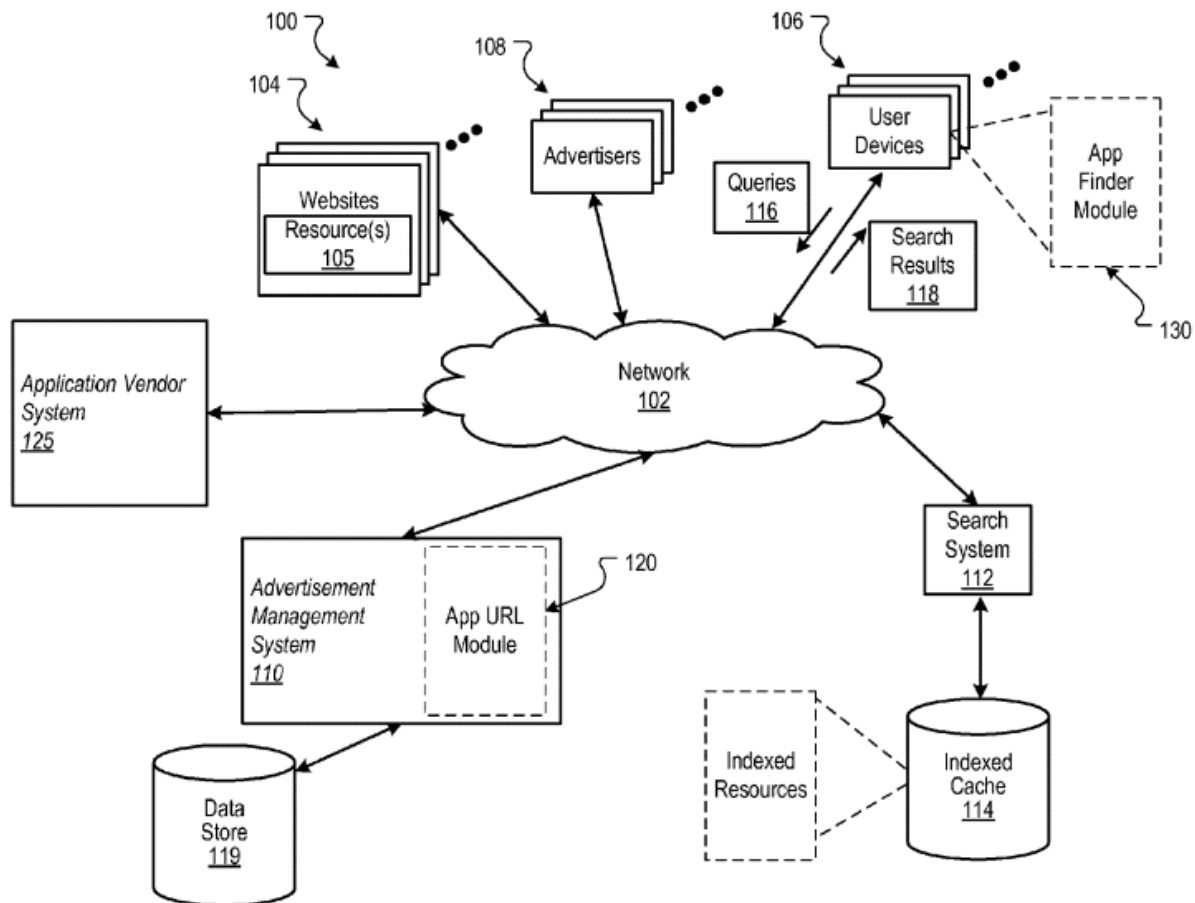


Fig. 1

(Ex. 1006, Fig. 1, ¶¶0031)(Ex. 1002, ¶80). The system also has user devices 106 (*e.g.* smartphones), which can download “resources” 105 (*e.g.* websites). (Ex. 1006, ¶¶0033-0035, 0103)(Ex. 1002, ¶80). The websites have advertisement slots that are intended to be filled with ads. (Ex. 1006, ¶0035)(Ex. 1002, ¶80). When a user views a webpage on a browser, the webpage—just as in Demsey—makes a request to the central server for advertisements to fill the ad slots. (Ex. 1006, ¶0038)(Ex. 1002,

¶80). And just as in Demsey, the central server sends back a response with the ads. (Ex. 1006, ¶0040)(Ex. 1002, ¶80).

To this basic ad-network system, Khanna adds mobile app deeplinking capability. (Ex. 1006, Abstract, ¶0025)(Ex. 1002, ¶81). Specifically, Khanna allows the ads to have links that specify a target mobile app, and a target product. (Ex. 1006, ¶0025)(Ex. 1002, ¶81). The links are generated when the central server selects an ad. (Ex. 1006, ¶0044)(Ex. 1002, ¶81). The central server uses an application URL module 120 (shown above in Fig. 1) to generate or modify the linked URL in the ad such that it will open a target mobile app to a desired starting point relating to the desired product (note that “product” is understood broadly, and could be, for example, information about a song and artist, ¶0051). (Ex. 1006, ¶0044)(Ex. 1002, ¶81). Khanna also provides code on the client device to check whether the target application is installed (¶¶0028, 0067), and if so, to carry out the deeplinking function (¶¶0046-0047). (Ex. 1002, ¶81).

C. Rationale (Motivation) Supporting Obviousness²

It would have been obvious carry out Haggerty’s method (on Haggerty’s central server) by using Demsey’s techniques for Internet advertising, extended using Khanna’s techniques for mobile-app deeplinking. (Ex. 1002, ¶82).

² The Overview (§B) and Introduction (§I) are relevant to this section.

First, it was obvious to use Demsey and Khanna together. (Ex. 1002, ¶83). Both references are intended for use with web-based advertising. (Ex. 1005, Abstract)(Ex. 1006, ¶¶0003, 0038-0039)(Ex. 1002, ¶83). Demsey describes an ad-network central server that provides ads to webpages that can be read on mobile devices. (Ex. 1005, ¶¶0005, 0023)(Ex. 1002, ¶83). Khanna picks up at this point, operating on ads, to be incorporated into a webpage for reading on a mobile device, that are served from a central server based on an ad request. (Ex. 1006, ¶¶0031, 0038-0039)(Ex. 1002, ¶83). Given the natural technological relationship between the two references, it would have been obvious to a PHOSITA to extend Demsey's capability with Khanna's. (Ex. 1002, ¶83).

Additionally, a PHOSITA with Demsey's system would have desired the mobile-app-deeplinking technology of Khanna, because Khanna states that "[a]llowing publishers or advertisers to create content items that deep-link directly to the content within their applications may lead to better user experience as well as create opportunities for the publisher to drive more business." (Ex. 1006, ¶0025)(Ex. 1002, ¶84). Similarly, a PHOSITA considering Khanna's system would have desired to use the system of Demsey, in order to be able to serve advertisements and to track users and user interactions with advertisements (Ex. 1005, ¶0021), as expressly desired by Khanna (Ex. 1006, ¶0043). (Ex. 1002, ¶84).

Second, it was obvious to implement Haggerty's method using the techniques

of Demsey and Khanna. (Ex. 1002, ¶85). Haggerty's method would have appealed to businesses that sell goods and services and that seek new potential customers. (Ex. 1007, 1:19-26, 1:53-62)(Ex. 1002, ¶85). In particular, it would have been attractive to have a streamlined method for acquiring the contact information and for acquiring consent to receive marketing communications and proof of possession from people interested in the business's offerings. (Ex. 1007, 1:19-26, 1:53-62, 17:4-18, 18:46-59)(Ex. 1002, ¶¶85, 24-26).

A PHOSITA would have found the combination of Haggerty with the techniques of Demsey/Khanna advantageous. (Ex. 1002, ¶86). Haggerty teaches a two-step sequence for acquiring new leads, in which a potential customer first interacts with a hyperlink, and second, clicks "send" on a pre-filled text message. (Ex. 1007, 10:1-5, 1:53-62, 14:41-49, 1:36-62, 2:1-5, 8:51-56, 10:59-11:21, 16:26-31, Figs. 3C, 4G)(Ex. 1002, ¶86). Haggerty further teaches that its links can be incorporated into electronic advertising. (Ex. 1007, 14:41-49)(Ex. 1002, ¶86). A PHOSITA would have found obvious that the hyperlinks of Haggerty that open a text-messaging application to a particular (pre-populated) state are exactly the kind of function that Khanna's deeplinks provide, and would have been aware of the standard URI protocols available to deeplink to SMS applications, as discussed in the Introduction. (Ex. 1002, ¶86). A PHOSITA would have found it advantageous to use Internet ad techniques like Demsey/Khanna in order to increase the reach of

Haggerty's marketing efforts, to better target advertisements to particular users (as expressly taught by Demsey), and to account for conversions of deeplinks (as expressly taught by Khanna). (Ex. 1005, ¶¶0002, 0021)(Ex. 1006, ¶¶0043, 0051, 0066)(Ex. 1002, ¶86).

In particular, it would have been obvious to carry out the server-based software techniques of Demsey and Khanna at Haggerty's call center server. (Ex. 1002, ¶87). Haggerty teaches that its call center functionality can be expanded with other server-based functions. (Ex. 1007, 7:34-36)(Ex. 1002, ¶87). Each of Demsey and Khanna, moreover, provides functionality with a central server (like Haggerty's call-center), such as Demsey's Synchronization Server 150 and Khanna's AMS 110 (Ex. 1005, ¶¶0021-0028, 0045-0048)(Ex. 1006, ¶¶0031, 0038-0039)(Ex. 1002, ¶87). Both Demsey and Khanna also express flexibility in the arrangement of the software, and user and server devices that make up their systems. (Ex. 1005, ¶¶0026-0032)(Ex. 1006, ¶¶0108, 0120-0122)(Ex. 1002, ¶87). It would have been obvious to combine the functionalities of these references into a single "server" as that term is understood in the '887 patent (Ex. 1001, 5:4-6:4) in order to obtain the benefits of each system as described above. (Ex. 1002, ¶87).

The combination further would have been obvious under *KSR Int'l Co. v. Teleflex, Inc.*, 550 U.S. 398, 417-21 (2007), as the application of known techniques (Demsey and Khanna), for their known functions, in order to improve Haggerty,

with no unpredictable results. (Ex. 1002, ¶88).

D. *Graham* Factors

This discussion of *Graham* factors applies to all Grounds.

The **level of ordinary skill** encompassed a person having a Bachelor's Degree in computer science and three years of experience in Web-based programming methods, where a higher level of education may substitute for experience and vice-versa. (Ex. 1002, ¶¶68, 89-93).

The **scope and content of the prior art** are discussed throughout the Ground.

The **differences between the prior art and the claims** are discussed above and in the claim mapping, below.

Secondary considerations are discussed in §IV, below.

E. Reasonable Expectation of Success

A PHOSITA in the relevant timeframe would have had a reasonable expectation of success in using the prior art in the manner discussed in this petition. (Ex. 1002, ¶94). As Dr. Houh explains, the methods of the prior art are implemented in software, are compatible, and were predictable in the relevant timeframe (May 2017). (Ex. 1002, ¶94). A PHOSITA would have been able to make any necessary modifications to implement the Ground, and in particular would have been able to implement Khanna's and Demsey's software technique together with Haggerty's teachings. (Ex. 1002, ¶94).

F. Analogous Art

Haggerty, Demsey, and Khanna are analogous art because they are in the same field as the '887 patent (Internet and mobile device technology). (Ex. 1001, 1:30-35)(Ex. 1007, Abstract)(Ex. 1005, Abstract)(Ex. 1006, Abstract)(Ex. 1002, ¶95). Furthermore, the methods of Haggerty, Demsey, and Khanna as described above would have been reasonably pertinent to the problems facing the named inventors, for example, the problems of providing an application deeplinking system and a more convenient text-message signup process. (Ex. 1001, 1:48-52)(Ex. 1002, ¶95). *See Wyers v. Master Lock Co.*, 616 F.3d 1231, 1238 (Fed. Cir. 2010)(“The Supreme Court’s decision in *KSR* [cite omitted], directs us to construe the scope of analogous art broadly....”).

G. Claim Mapping

This section maps the challenged claims to the relevant disclosures of the prior art, where the claim text appears in bold-italics, and the relevant mapping follows the claim text. Petitioner has added numbering and lettering in brackets (*e.g.* 1[a], [1b]) to certain claim elements, to facilitate the discussion. Each mapping for a dependent claim combination also renders obvious the claims from which the dependent claim depends.

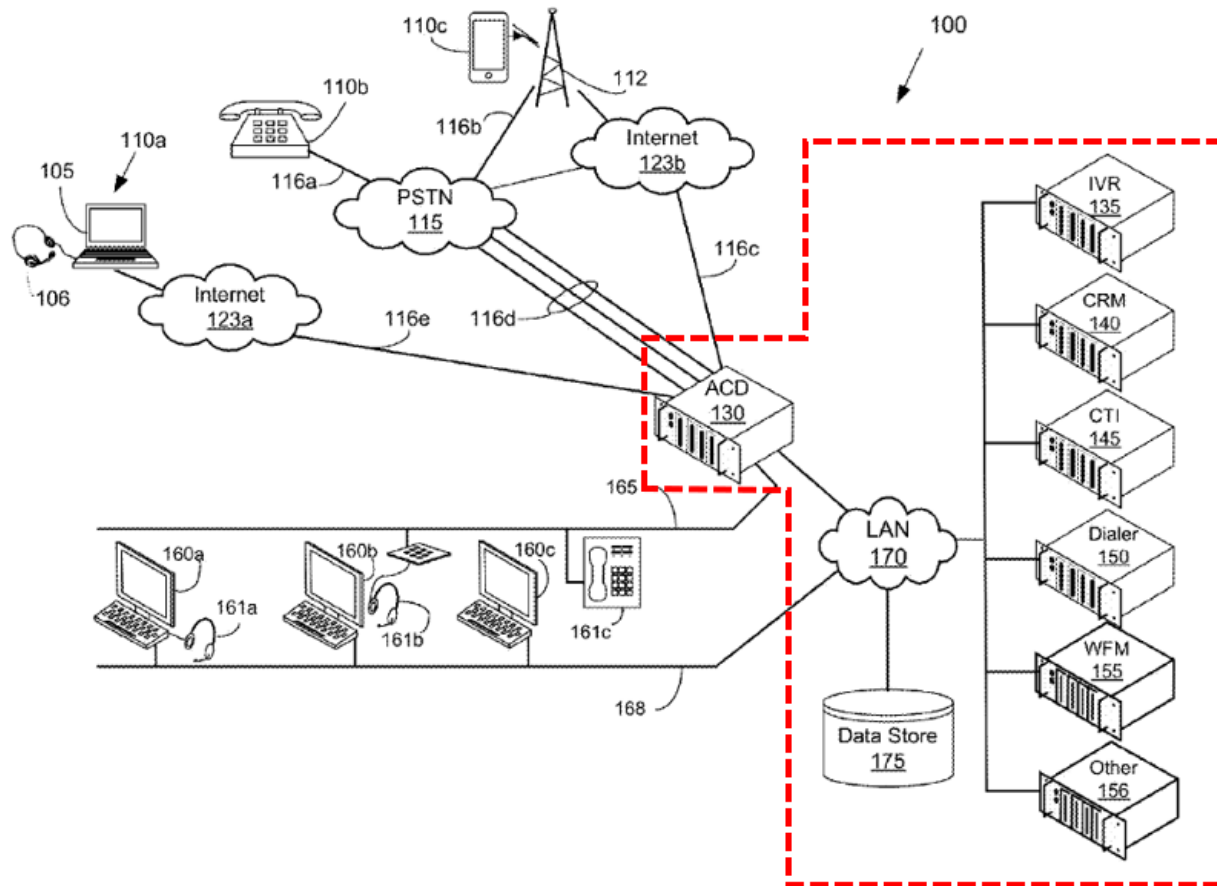
CLAIM 1

“1[a]. At least one non-transitory processor-readable medium storing code configured to be executed by a processor of a click-to-

text server, the code including instructions configured to cause the click-to-text server to:”

(a) non-transitory processor-readable medium storing code configured to be executed by a processor of a click-to-text server

Haggerty teaches a **non-transitory processor-readable medium storing code configured to be executed by a processor of a click-to-text server**. Specifically, Haggerty teaches a “call center” for carrying out its methods. (Ex. 1007, Fig. 1, 3:44-62)(Ex. 1002, ¶¶97-98). The call center has several computing functions 130-156 and a data store 170, each of which can be implemented in software operating on a server. (Ex. 1007, Fig. 1, 3:44-63, 5:45-7:62)(Ex. 1002, ¶¶98). These computing functions are shown in Fig. 1 of Haggerty, reproduced below, with an added red-dashed line around the **click-to-text server**:



(Ex. 1007, Fig. 1, 3:44-63, 5:45-7:62)(Ex. 1002, ¶98). The server is a “**click-to-text**” server because, in the combination, it performs the click-to-text functionality required by this claim in the same manner as the click-to-text server of the ’887 patent, as further described under elements [1b]-[1i], below. (Ex. 1002, ¶98). Furthermore, in Haggerty, the server receives the pre-populated text message and responds to it, as discussed below under limitations [1g]-[1i]. (Ex. 1002, ¶98).

Haggerty teaches that the various computing functions can be implemented in software running on a server (Ex. 1007, 7:42-62), which can be, for example, a single hardware platform (Ex. 1007, 7:52-56, 27:5-42, Figs. 10, 14)(Ex. 1002, ¶99).

Haggerty further teaches that the functions of the **click-to-text server** (as identified above) may be carried out using a **non-transitory processor-readable medium storing code configured to be executed by a processor**. (Ex. 1007, 28:16-42)(Ex. 1002, ¶¶99-100). Petitioner here notes the '887 patent's teachings regarding the hardware and software arrangement of the click-to-text server. (Ex. 1001, 5:4-6:4)(Ex. 1002, ¶99).

(b) code including instructions configured to cause the click-to-text server to

For the reasons described above in the section entitled “Rationale (Motivation) for the Combination”, it would have been obvious to combine certain functions of Demsey and Khanna with Haggerty’s call center, and it thus would have been obvious, as taught by Haggerty, to arrange the **code** of the click-to-text server to **include instructions configured to cause the click-to-text server to perform the functions of the click-to-text server recited in each of the claim limitations below**. (Ex. 1007, 28:16-42)(Ex. 1002, ¶101).

“[1b] send, to a client server, an integration tag configured to be served with a webpage hosted by the client server,”

Demsey discloses **sending, to a client server, an integration tag configured to be served with a webpage hosted by the client server**. (Ex. 1005, ¶0005)(Ex. 1002, ¶102). The integration tags used in Demsey are called “ad tags”. (Ex. 1005, ¶0005)(Ex. 1002, ¶102). Demsey states:

“By way of background, ad networks typically have hundreds or thousands of partners (often referred to as ‘**publishers**’) that **publish online content along with ‘ad tags’ provided by their partner ad network**. For example, each of a news website (www.news.com), a blog (www.blog.com), and an informational website (www.info.com), among others, may engage an ad network to efficiently and profitably advertise to those websites’ viewers. The **ad network provides each of those sites with an ad tag** having the ad network’s domain, e.g., ‘adnetwork.com.’”

(Ex. 1005, ¶0005)(Emphasis added)(*see also* Ex. 1005, ¶¶0030, 0045, 0047). Ad tags of Demsey are “**integration tags**” because they are code (JavaScript, ¶¶0045, 0047) that performs the functions as an integration tag. (Ex. 1002, ¶¶103, 105)(*compare* Ex. 1001, 5:35-42)(*see also* discussion under limitations [1c] and [1d] below for the functions carried out by the ad tag).

The “publishers” of Demsey are publishers of websites from servers 112 (as shown in Fig. 2, *e.g.*). (Ex. 1005, ¶¶0025-0026, 0005, 0045)(Ex. 1002, ¶103). Because the publishers partner with ad networks for ads, they are “clients”, and their servers are **client servers**. (Ex. 1005, ¶¶0005-0006)(Ex. 1002, ¶103).

As discussed above, the ad tags are **configured to be served with a webpage hosted by the client server**. (Ex. 1005, ¶¶0005, 0045, 0047)(Ex. 1002, ¶104). This is shown, *e.g.*, in Fig. 5, with an added red-dashed box to show the service of a web page with an ad tag from a publisher server 506:

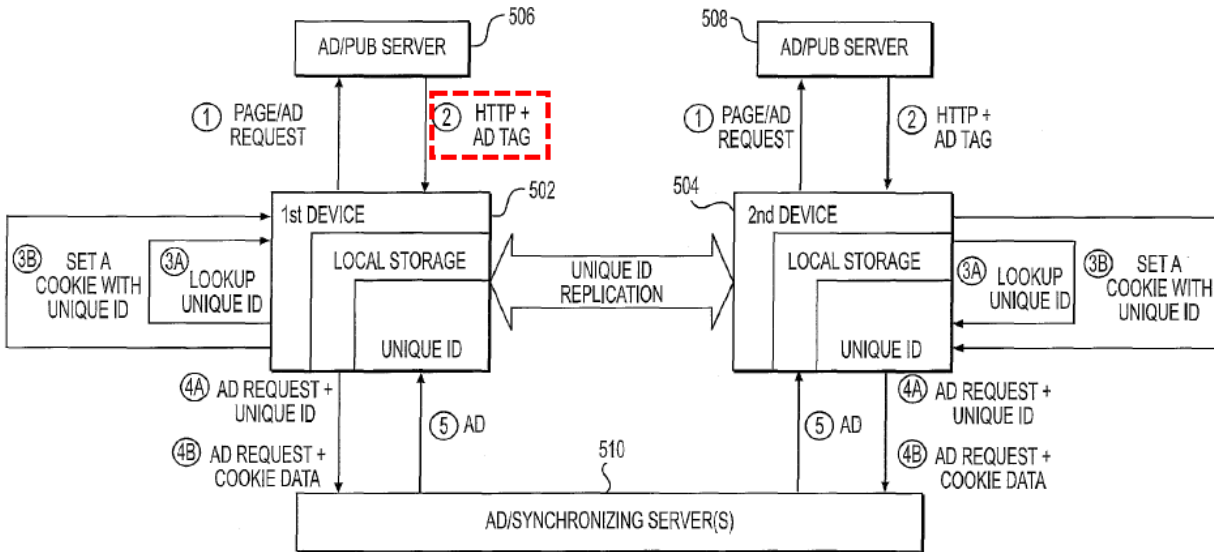


FIG. 5

(Ex. 1005, Fig. 5, ¶¶0045, 0047)(Ex. 1002, ¶104).

It would have been obvious, based on Demsey’s disclosures, to implement the **sending** of the integration tag by the **click-to-text server** (as defined above under limitation [1a]) **to the client server**. (Ex. 1002, ¶106). Demsey already teaches that “[t]he ad network provides each of those sites with an ad tag.” (Ex. 1005, ¶0005)(Ex. 1002, ¶106). It would have been apparent and obvious that providing ad tags to a site means sending the ad tag **to the client server** that serves the webpage, because the ad tag will be integrated *into* the webpage, and thus served with the webpage from that same server. (Ex. 1005, ¶¶0045, 0047, 0005, Fig. 5)(Ex. 1002, ¶106). It was further apparent and obvious to **send the integration tag by the click-to-text server**, which in the combination performs the function of Demsey’s central server. (Ex. 1002, ¶106). Demsey discloses that publisher servers communicate with the ad

network via the central server of the ad network (the “synchronizing server” in Demsey)(Ex. 1005, ¶0026). (Ex. 1002, ¶106). The functionality of the ad network is implemented at its servers (Ex. 1005, ¶0028), and thus it would have been obvious to provide ad tags from the click-to-text server in the combination. (Ex. 1002, ¶106). Furthermore, it would have been obvious to a PHOSITA to arrange to have the click-to-text server perform the sending of the ad tags at scale. (Ex. 1002, ¶106). As Demsey explains, there may be hundreds or thousands of partner publishers (Ex. 1005, ¶0005), and these publishers may change their websites frequently. (Ex. 1002, ¶106). Allowing the publisher servers to download ad tags from the click-to-text server would have facilitated the quick deployment of ad tags. (Ex. 1002, ¶106). An ad tag must also send data and an ad request to the correct address at the click-to-text server (Ex. 1005, ¶¶0045, 0047, Fig. 5) and in the correct form for the click-to-text server to use the data. (Ex. 1002, ¶107). The click-to-text server would have been best able to define the ad tag to send data in the *correct form* to the *correct place*. (Ex. 1002, ¶107).

Furthermore, the technique of sending over the Internet (*i.e.* downloading) a piece of code, from one server to another, was well-known in the relevant timeframe (early 2017), would have been predictable, and was one of only a few possible choices available to a PHOSITA seeking to implement Demsey’s teaching that “ad network provides each of those sites with an ad tag”. (Ex. 1005, ¶0005)(Ex. 1002,

¶108). *See KSR*, 550 U.S. at 416-21.

“[1c] the integration tag configured to cause a mobile device loading the webpage via a first application to send user data to at least one of the client server or the click-to-text server;”

(a) mobile device loading the webpage via a first application

Demsey discloses this limitation. First, Demsey discloses **mobile devices** (Ex. 1005, ¶0023, Figs. 1 and 2, rn 102, 110) that **load webpages** (Ex. 1005, ¶¶0005, 0045, 0047) for viewing **via a first application** (*e.g.* a browser) (Ex. 1005, ¶¶0005, 0045, 0047). (Ex. 1002, ¶109). The webpages contain integration tags (as discussed above under limitation [1b]) comprising, *e.g.*, JavaScript. (Ex. 1005, ¶¶0005, 0045, 0047, Fig. 5)(Ex. 1002, ¶109). Demsey describes and renders obvious that mobile device browsers loading webpages that contain JavaScript ad tags will execute when the webpage is **loading**. (Ex. 1005, ¶¶0005, 0045, 0047)(Ex. 1002, ¶¶109). As Dr. Houh explains, browsers in the relevant timeframe executed scripts present within a webpage at the time of loading the webpage. (Ex. 1002, ¶109).

(b) integration tag configured to cause...to send user data to at least one of the client server or the click-to-text server

Demsey renders obvious that the **integration tags** are **configured to cause a mobile device loading the webpage via a first application to send user data to at least one of the client server or the click-to-text server**. (Ex. 1002, ¶110). Specifically, the ad tags cause a unique ID to be passed to the central server (*i.e.* the

click-to-text server in the combination). Demsey states:

“As shown in FIG. 5, in one embodiment, the user’s first device 502 may (1) request a web page or ad from an ad or publisher server 506; and (2) **receive HTTP and an ad tag** from the ad or publisher server 506. The user’s first device 502 may **execute the ad tag**, such as by executing any type of client side script (e.g., **JavaScript**) to (3A) **look-up a unique ID** stored on the device. As described above, the unique ID may be stored within local storage associated with a browser of the device 502. The user’s first device 502 may then (4A) **issue an ad request to ad or synchronizing servers 510, where the request includes the looked-up unique ID.**”

(Ex. 1005, ¶0045)(Emphasis added)(Ex. 1002, ¶110). A PHOSITA would have understood this to mean and found obvious that Demsey’s JavaScript ad tag is causing an ad request containing a unique ID to be sent to the central server, which in the combination would be the **click-to-text server**. (Ex. 1002, ¶110). Demsey teaches that the unique ID can be an ID identifying a user, or an instance ID identifying a user in combination with a particular device. (Ex. 1005, ¶¶0035-0036)(Ex. 1002, ¶110). Because the unique ID identifies a user or a user and device, it is **user data**. (*Compare* Ex. 1001, 5:18-23)(Ex. 1002, ¶110).

Demsey further discloses that **integration tags** can **cause the mobile device to send** cookie data to at least one of the client server or the click-to-text server. (Ex. 1005, Fig. 5, rn 4B, ¶¶0045, 0047)(Ex. 1002, ¶111). Cookie data is also **user data**.

(Compare Ex. 1001, 5:18-23)(Ex. 1002, ¶111).

“[1d] define and send a uniform resource identifier (URI) to the mobile device in response to the mobile device executing the integration tag, the URI deeplinking to a messaging application different from the first application and”

(a) define and send a uniform resource identifier (URI) to the mobile device in response to the mobile device executing the integration tag

While Demsey does not teach mobile app deeplinking, Khanna teaches mobile app deeplinking from ads served from an ad server. (See Overview, above). Khanna’s deeplinking would have been obvious to use for the reasons discussed under the section “Rationale (Motivation) for the Combination”. (Ex. 1002, ¶112).

Specifically, Khanna’s central server, AMS 110 (in the combination, the click-to-text server), receives ad requests when users browse websites (just as in Demsey). (Ex. 1006, ¶¶0035-0039)(Ex. 1002, ¶113). In response to an ad request, Khanna teaches returning an advertisement (called a “content item”). (Ex. 1006, ¶¶0031, 0035-0039)(Ex. 1002, ¶113). In addition, Khanna teaches that the content item (advertisement) can be augmented by **defining and sending a uniform resource identifier (URI) to the mobile device**, where the **URI deeplinks to an application different from the first application**. (Ex. 1006, ¶¶0043-0044, *see also* ¶¶0025-0030)(Ex. 1002, ¶113). Khanna explains:

“The advertisement management system 110 may also include an

application URL module 120. The application URL module may be used to **generate or modify** a formatted URL for a content item....In some implementations, the **application URL module may generate a deep-linking application specific landing page URL** based on unformatted data elements associated with the content item received from an advertiser 108.”

(Ex. 1006, ¶0044)(Emphasis added)(Ex. 1002, ¶113). A URL (as used in Khanna) is simply a more-specific form of a **URI**. (Ex. 1002, ¶113). After the formatted URI of Khanna is “generate[d] or modif[ied]” at the server, it is **sent** with the content item to the mobile device for display. (Ex. 1006, ¶¶0044, 0051, 0038-0041, Fig. 2)(Ex. 1002, ¶113). Khanna’s URIs **deeplink to an application different from the first application**, because the first application is a browser (*see* limitation [1b]), while the second application is a specific application associated with the content item, and the browser first checks to see if the application is installed before launching it. (Ex. 1006, ¶¶0067, 0005)(Ex. 1002, ¶113). To transition from one application to another, Khanna teaches using “**deep-linking**” technology. (Ex. 1006, ¶¶0015, 0025-0030, 0044)(Ex. 1002, ¶113).

(b) the URI deeplinking to a messaging application different from the first application

In the combination, the URI **deeplinks to a messaging application different from the first application**. (Ex. 1002, ¶114). As discussed above, it would have been obvious to use the concepts of Demsey and Khanna to implement the business

method of Haggerty (Ex. 1002, ¶114), wherein the deeplinking technology of Khanna is used to open a pre-populated text message in a text messaging application. (Ex. 1007, 9:39-44, 9:56-10:6)(Ex. 1008, p. 030, ¶¶0066-0067)³(Ex. 1002, ¶114). This use of URIs was well-known in the relevant timeframe, as discussed in the Introduction. (Ex. 1002, ¶114). The messaging application is **different from the first application**, because the first application is a browser. (Ex. 1002, ¶114).

“[1e] [the URI...] configured to cause the mobile device to (1) automatically transition from the first application to the messaging application in response to the mobile device detecting a user interaction with a promotional message associated with the webpage, the messaging application being different from the first application, and”

(a) automatically transition...in response to the mobile device detecting a user interaction with a promotional message associated with the webpage

Khanna teaches that when a user clicks on a portion of the content item (advertisement) that activates the URI (*i.e.* when the mobile device **detects a user interaction with a promotional message associated with the webpage**), the URI **causes** the mobile device **to automatically transition** from a first mobile

³ Ex. 1008 is U.S. Pat. App. 13/461,541 incorporated-by-reference into Haggerty at Ex. 1007, 1:7-15. Citations to Ex. 1008 are in the format {page}, {¶}, with {page} indicating the three-digit numbers (bottom) added per 37 C.F.R. §42.64(d)(2)(i).

application to a different mobile application. Khanna states:

“When a user selects (e.g., clicks on) the content item, if the application is installed, the application may be launched with the associated product related data elements as inputs so that the content (e.g., a landing page) related to the product is displayed in the application.”

(Ex. 1006, ¶0027)(Ex. 1002, ¶115). The “content item” is the **promotional message**. It is **associated with the webpage** because it is within a content slot of the webpage. (Ex. 1006, ¶¶0003, 0035)(Ex. 1002, ¶115). Khanna teaches that the transition to a second application is **automatic** after detecting user input. (Ex. 1006, ¶¶0006, 0008, 0010, 0013-0014, 0046, 0062-0063)(Ex. 1002, ¶115). It was obvious in the relevant timeframe that a device **detects** user input. (Ex. 1002, ¶115).

It would further have been obvious to cause a user click on an advertising link in Demsey (*i.e.* a **promotional message**) to launch a text **messaging application**, as taught by Haggerty, for the reasons provided above under the section entitled “Rationale (Motivation) for the Combination”. (Ex. 1002, ¶116). In the combination, the app that is automatically opened is a **messaging application** per Haggerty. (Ex. 1007, 9:39-44)(Ex. 1002, ¶116). The messaging application is **different from the first application**, because the first application is a browser. (Ex. 1002, ¶116).

“[1f] [the URI...configured to] (2) automatically populate a custom message in the messaging application that includes an address associated with the click-to-text server and a message body that includes an identifier associated with at least one of the webpage or the user data”

(a) automatically populate a custom message in the messaging application that includes an address associated with the click-to-text server

The combination renders this limitation obvious by carrying out Haggerty’s business method using the functionality of Demsey and Khanna. (Ex. 1002, ¶117). As explained above under elements [1d] and [1e], Khanna describes deeplink URIs for inclusion into an advertisement, where the deeplinks (when clicked by a user) cause the mobile device to transition to a second application. (Ex. 1006, ¶¶0044-0046)(Ex. 1002, ¶117). Specifically, it would have been obvious to arrange a link using Khanna that would (per Haggerty) initiate “a pre-populated text message addressed to the destination address with the contents indicated therein is presented to the user, along with a prompt for confirmation for sending the message.” (Ex. 1007, 10:2-5, *see also* 14:41-49, 1:36-62, 2:1-5, 8:51-56, 10:59-11:21, 16:26-31, Figs. 3C, 4G)(Ex. 1002, ¶117). Here, the “destination address” as described in Haggerty (Ex. 1007, 10:2-5) was **an address associated with the click-to-text server**. (Ex. 1007, Fig. 4B, 12:39-67)(Ex. 1002, ¶117).

The “pre-populated text message” would be an **automatically-populated custom text message** as taught by Haggerty, which could be carried out using the

automatic transition and passing of data of Khanna as discussed under limitation [1d]. (Ex. 1002, ¶118).

(b) and a message body that includes an identifier associated with...the user data

Haggerty expressly teaches incorporating information about the **user** into the **pre-populated message body**, stating:

“[T]he user may be prompted for authorization for the publisher 490 to use **information about the user**. Accordingly, **this information may be sent along in the body of the message** to the publisher 490. Further, in particular embodiments, the advertisement with the embedded link may also request **information from the reader** that may be **included in the body of the message**. For example, the advertisement may request a **telephone number** at which the reader may be reached. **Those of ordinary skill in the art can envision several pieces of information that may be included along with the message** 482 sent to the publisher 490 in light of this disclosure.”

(Ex. 1007, 14:60-15:4)(Emphasis added)(*see also* Ex. 1007, 15:5-15)(Ex. 1002, ¶119). Information about the user is an **identifier** associated with the user, and therefore **associated with the user data**. (Ex. 1007, 14:60-15:15)(Ex. 1002, ¶119). For example, a telephone number of the user is a specific **identifier** of a user that is **associated with user data**, because it also serves to identify the user. (Ex. 1002, ¶119). Khanna teaches a method to pass such data to a target application in a deeplink (Ex. 1006, ¶0061) and other methods were known in the art. (Ex. 1002,

¶121).

Furthermore, Haggerty (in the quote above) expressly suggests including other information associated with the user in the body of the message. (Ex. 1007, 14:60-15:15)(Ex. 1008, p. 031, ¶0069)(Ex. 1002, ¶120). It would have been obvious based on Haggerty's express suggestion to include user-identifying information (Ex. 1007, 14:60-15:15) to incorporate information about the user that is already available, such as the unique identifier or unique instance (device) identifier from Demsey (*see* limitation [1c], above) in the body of the text message, to allow a server receiving a text message to identify the user or the user's device for further marketing activities and to track the conversion of an advertisement through the sending of a text message. (Ex. 1006, ¶0042)(Ex. 1002, ¶120). A unique ID or instance ID is **associated with the user data** because it *is* user data. (Ex. 1002, ¶120).

(c) and a message body that includes an identifier associated with...the webpage

Haggerty also teaches including **identifiers** that are **associated with the webpage**. For example, Haggerty teaches including a product or service name ("Widget") in the body of the text message (Ex. 1007, 9:64-10:6, Fig. 3A). (Ex. 1002, ¶122). The product or service name is an **identifier associated with the webpage** because it is the name of the product or service that the ad on the webpage is promoting. (Ex. 1002, ¶122). Where the promotion occurs on a webpage

published by the seller of the product or service, the product or service name is also associated with the proprietor of the webpage of the product or service. (Ex. 1002, ¶122). Similarly, Haggerty teaches including an alphanumeric “campaign identifier” in the body of the text message. (Ex. 1007, 12:55-67, 15:5-15, 19:55-63, 20:39-40)(Ex. 1008, p. 033, ¶0078, p. 037, ¶0092)(Ex. 1002, ¶122). The campaign identifier identifies the advertising campaign, and is **associated with the webpage** because the advertisement is placed in the webpage. (Ex. 1002, ¶122).

“[1g] such that, upon detecting a single user interaction with a send button of the messaging application, the mobile device sends the custom message to the click-to-text server;”

Haggerty teaches that:

“a pre-populated text message addressed to the destination address with the contents indicated therein is presented to the user, along with **a prompt for confirmation for sending the message**. In other embodiments, the message can be immediately sent.”

(Ex. 1007, 10:1-6)(Emphasis added)(*see also* Ex. 1007, 9:39-55)(Ex. 1002, ¶123).

Haggerty teaches two cases, one in which a message is sent immediately (without user input), and a second case in which “a prompt for confirmation for sending the message” is displayed first. (*Id.*). This latter case discloses and renders obvious that only a **single interaction** (*i.e.* the user’s confirmation action) need be **detected** by the mobile device before sending the message. (Ex. 1002, ¶123). Furthermore, it

would have been obvious to have the **single interaction be with a send button**, because it was well-known in the relevant timeframe that mobile SMS applications had send buttons that already performed the function of confirming that a message should be sent. (Ex. 1002, ¶123)(Ex. 1011, Fig. 1, ¶0007). This would have been obvious as the most natural and user-intuitive implementation of a confirmation (because users were already used to pressing “send”) and, independently, would have been obvious as a known method for implementing a send confirmation as taught by Haggerty that could have been implemented with no unpredictable results. (Ex. 1002, ¶123). *See KSR*, 550 U.S. at 417-21.

Haggerty further teaches sending the SMS to the click-to-text server (as discussed under limitation 1[a]). (Ex. 1007, 19:33-40, 12:39-47, Figs. 4B-4C, 18:38-42)(Ex. 1002, ¶124). This is shown, for example, in Fig. 7A, reproduced below, with highlighting added to show the sending of the SMS sent **to the click-to-text server**:

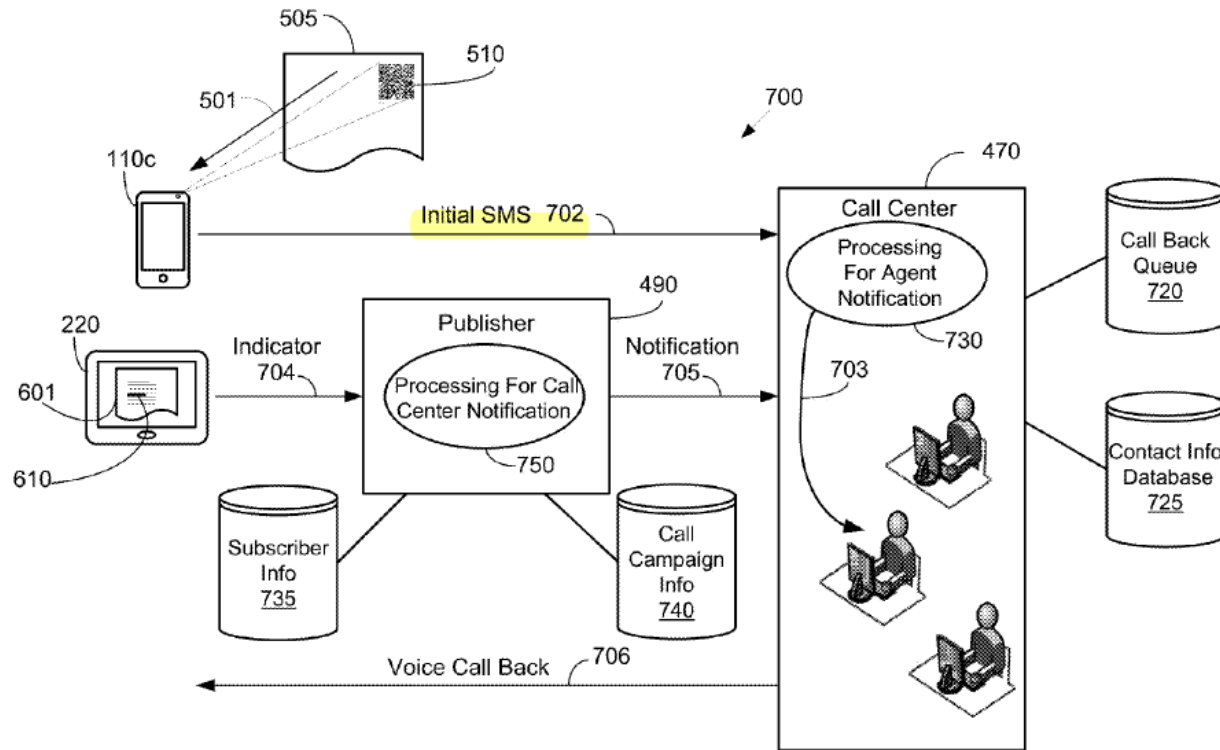


FIG. 7A

(Ex. 1007, Fig. 7A, 19:33-40, Figs. 4B-4C, 12:39-47, 18:38-42)(Ex. 1002, ¶124).

“[1h] receive the custom message at the click-to-text server; and”

Haggerty teaches that its call-center server (the click-to-text server) **receives the custom text message**. (Ex. 1007, 23:1-12, Fig. 9, rn 905, Fig. 7A, 19:33-40, Figs. 4B-4C, 12:39-47, 13:9-35, 18:38-42)(Ex. 1002, ¶125).

“[1i] enroll the mobile device in a promotion associated with the promotional message based on receiving the custom message and without receiving any additional information from the mobile device.”

Haggerty teaches that, when the message is received at the click-to-text server,

it **enrolls the mobile device in a promotion** by recording user information in a database, recording the user's consent to receive further communications in a consent database, and placing the user in a queue for a response (a call, text or email back). (Ex. 1007, Fig. 9, 22:47-23:21, Fig. 4B, 12:39-13:8)(Ex. 1002, ¶126).

The communication to the user back from the click-to-text server can involve any type of conventionally-known **promotion**, such as the provision of a coupon, discount, or further information about a product or service. (Ex. 1007, 10:27-30, 13:1-8, Fig. 9, 23:8-21)(Ex. 1008, p. 030, ¶0068)(Ex. 1002, ¶127). The promotion is **associated with the promotional message**, because the click-to-text server first identifies the campaign associated with the promotional message prior to enrolling the user in the promotion. (Ex. 1007, Fig. 9, 22:47-23:21, Fig. 4B, 12:39-13:8)(Ex. 1002, ¶127). The enrollment in the promotion happens **without receiving any additional information from the mobile device**. This is shown, for example, in Fig. 4B, reproduced below, where only the initial SMS message (418) is received from the mobile device prior to the response SMS (422):

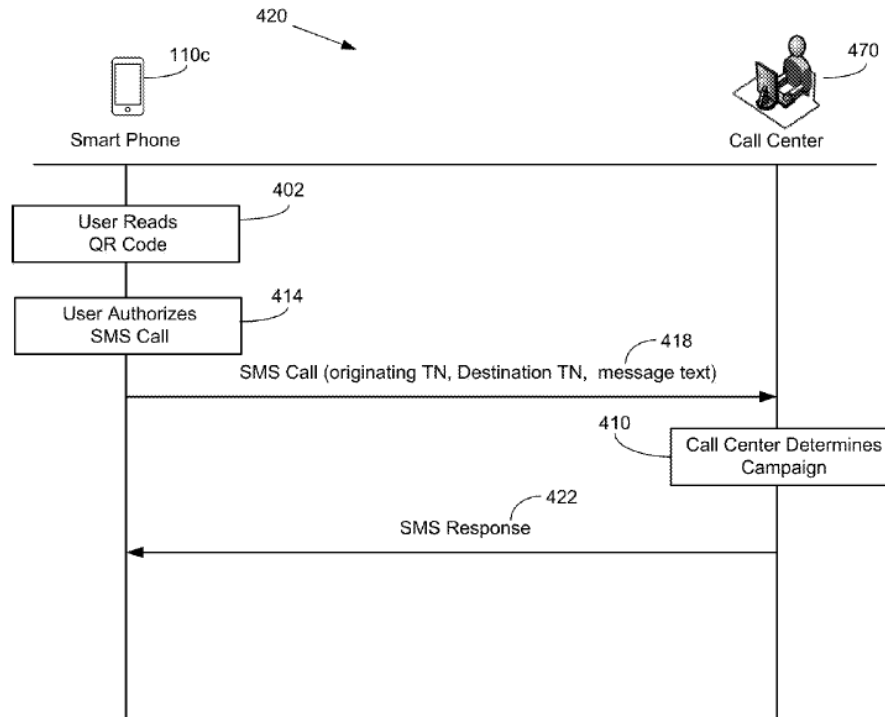


FIG. 4B

(Ex. 1007, Fig. 4B, 12:39-13:8)(*see also* Fig. 9, 22:47-23:21)(Ex. 1002, ¶127). *See AC Techs., S.A. v. Amazon.com, Inc.*, 912 F.3d 1358, 1367 (Fed. Cir. 2019)(“[A] reference need not state a feature’s absence in order to disclose a negative limitation.”).

CLAIM 2

“2. The non-transitory processor-readable medium of claim 1, wherein the mobile device is enrolled in the promotion without the click-to-text server charging a fee to a user of the mobile device.”

Haggerty’s method does not charge a fee to the user for enrollment in the promotion. (Ex. 1007, Fig. 9, 22:47-23:21, Fig. 4B, 12:39-13:8)(Ex. 1002, ¶128). *See AC Techs.*, 912 F.3d at 1367.

CLAIM 13

“13. The non-transitory processor-readable medium of claim 1, wherein the user data includes at least one of a mobile device identifier, a user viewing history, user tap history, cookie information, an identification of a referring webpage, or a user agent of the first application.”

Demsey teaches that the user data can include an instance identifier, which is a **mobile device identifier**. (Ex. 1005, ¶¶0039, 0036, 0048)(Ex. 1002, ¶129).

Demsey further discloses that the user data can be **cookie information**. (Ex. 1005, Fig. 5, rn 4B, ¶¶0045, 0047)(Ex. 1002, ¶129).

CLAIM 14

“14. The non-transitory processor-readable medium of claim 1, the code further comprising code to cause the click-to-text server to: send a signal to cause the promotional message to be displayed at the mobile device via the first application.”

Both Demsey and Khanna teach having the central server (the **click-to-text server** in the combination) send an ad to the mobile device browser in response to an ad request. (Ex. 1005, Fig. 5, rn 4A-5, ¶¶0045-0048, 0030)(Ex. 1006, ¶¶0039-0040, Fig. 1). This is a **signal to cause the promotional message to be displayed at the mobile device via the first application**. (Ex. 1002, ¶130).

Ground 2. Claims 1-4, 6-10, 13-14, and 19 were obvious as in Ground 1, in further view of Rakavy.

Claims 1-4, 6-10, 13-14, and 19 were obvious as in Ground 1, in further view

Patent No. 11,416,887

of U.S. Pat. No. 5,913,040 (“**Rakavy**”)(Ex. 1018).

Rakavy was not cited in the application leading to the ’887 patent.

A. Prior Art Status

Rakavy was published in 1999 and is thus prior art under 35 U.S.C. §102(a)(1).

B. Overview of the Ground and Rationale for the Combination

Claims 1-2, and 13-14 are obvious as in Ground 1. Claims 3-4, 6-10 and 19 all claim various forms of authentication or confirmation of a user, and would have been obvious in further view of motivation provided by Rakavy. (Ex. 1002, ¶132)

Rakavy is an early web-advertising patent. (Ex. 1002, ¶133). Like Demsey, Rakavy teaches an ad network with a central server (Advertising System Server 600) that can provide ads to user devices. (Ex. 1018, Abstract, Fig. 1, 4:39-45, 5:31-47)(Ex. 1002, ¶133). Like Demsey, Rakavy teaches assigning each user a user ID. (Ex. 1018, 5:31-47)(Ex. 1002, ¶133). Like Demsey, the user ID is used to identify a particular user to be better able to track user activities and target ads to the user. (Ex. 1018, 5:38-47)(Ex. 1002, ¶133).

As relevant to the present Ground, Rakavy teaches *authenticating* the user. Specifically, Rakavy states that advertisements might offer incentives such as contests and rewards. (Ex. 1018, 3:34-39, 6:21-26, 15:14-21)(Ex. 1002, ¶134). To avoid fraudulent behavior related to such ads, Rakavy teaches that it is advisable to

authenticate the user, and thus provides motivation to include the limitations of claims 3-4, 6-10 and 19. (Ex. 1018, 15:14-21, 6:21-26)(Ex. 1002, ¶134). As one example, Rakavy further specifically teaches to use the user ID to authenticate the user. (Ex. 1018, 6:16-26)(Ex. 1002, ¶134).

The combination and its rationale are further detailed in the claim mapping, below.

C. Reasonable Expectation of Success

A PHOSITA would have had a reasonable expectation of success in implementing ideas from Rakavy in the manner described above and in the claim mapping, below. (Ex. 1002, ¶136).

D. Analogous Art

Rakavy is in the same field as the '887 patent and would have been reasonably pertinent to the problems facing the named inventors. (Ex. 1018, 5:32-7:11, 15:14-21)(Ex. 1002, ¶137).

E. Claim Mapping

CLAIM 3

“3. The non-transitory processor-readable medium of claim 1, the code further comprising instructions to cause the click-to-text server to authenticate the mobile device based on the user data received in response to the mobile device loading the webpage and the identifier received with the custom message.”

(a) authenticate the mobile device based on the user data received in response to the mobile device loading the webpage

Demsey teaches that the **user data** can be a unique ID (*see* Ground 1, claim 1, limitation [1c]). (Ex. 1002, ¶138). Rakavy teaches to use a user ID to **authenticate** the user to avoid fraud. (Ex. 1018, 6:16-20, 15:14-21)(Ex. 1002, ¶138). It would have been obvious in the combination to use Demsey's unique ID or instance ID (**user data received in response to the mobile device loading the webpage**) to **authenticate the mobile device** in order to mitigate the potential fraud mentioned by Rakavy. (Ex. 1018, 15:14-21, 6:16-20)(Ex. 1002, ¶138). Authentication as suggested by Rakavy would have been most obvious to perform by checking provided information (*e.g.* a unique ID) against a database of information. (Ex. 1002, ¶138). Rakavy expressly teaches a database for authenticating, *e.g.*, using a user ID. (Ex. 1018, 6:16-20)(Ex. 1002, ¶138).

(b) authenticate the mobile device...based on the identifier received with the custom message

A PHOSITA would have found it obvious to use a separate **identifier received with the custom message** (*e.g.* a telephone number) to assist in the authentication of the user. (Ex. 1002, ¶139). For example, Haggerty (via its incorporated '541 application) teaches to use the user's telephone number to check against national wireless number databases, which is a form of authentication. (Ex. 1008, p. 031, ¶0069)(Ex. 1002, ¶139). Haggerty's incorporated '541 application further teaches that the SMS message can contain an additional security code associated with the user (**identifier received with the custom message**) to "aid the

recipient in identifying and verifying the sender”, *i.e.* to **authenticate the user**. (Ex. 1008, p. 031, ¶0069)(Ex. 1002, ¶139). Each of these methods would have provided additional and complementary protection against the potential for fraud described by Rakavy, and thus would have been obvious to use. (Ex. 1018, 15:14-21, 6:16-20)(Ex. 1002, ¶139).

Furthermore, the methods of authentication discussed under this claim 3 were well-known in the relevant timeframe and could have been used to improve Haggerty without unpredictable results. (Ex. 1002, ¶140). *See KSR*, 550 U.S. at 416-21.

CLAIM 4

“4. The non-transitory processor-readable medium of claim 1, the code further comprising instructions to cause the click-to-text server to: authenticate the mobile device based on the user data received in response to the mobile device loading the webpage; and send the promotional message to the mobile device in response to authenticating the mobile device.”

See above, claim 3 regarding **authenticating based on user data**. (Ex. 1002, ¶141). It would have been obvious, based on Rakavy, to **send the promotional message to the mobile device only in response to authenticating the mobile device** in order to avoid the fraud warned against by Rakavy, and to save resources associated with transmission to potentially fraudulent or spurious users. (Ex. 1018, 15:14-21, 6:16-20)(Ex. 1002, ¶141).

CLAIM 6

“6. The non-transitory processor-readable medium of claim 1, the code further comprising instructions to cause the click-to-text server to authenticate a user of the mobile device based on the custom message satisfying a predefined criteria, the mobile device enrolled in the promotion based on the user being authenticated.”

See above, claim 3. In the analysis of claim 3, the **predefined criteria** [sic] required by this claim (6) is that the user data and/or identifier matches corresponding stored data, such that the **user** can be **authenticated**. (Ex. 1002, ¶142). It would have been obvious to enroll the mobile device in the promotion (e.g. contest) based on the **mobile device being authenticated** to avoid the fraud described by Rakavy. (Ex. 1018, 15:14-21, 6:16-20)(Ex. 1002, ¶142).

CLAIM 7

“7. The non-transitory processor-readable medium of claim 1, the code further comprising instructions to cause the click-to-text server to authenticate a user of the mobile device based on information in the custom message appearing on a white list or being absent from a black list, the mobile device enrolled in the promotion based on the user being authenticated.”

See above, claim 3. Authentication using any of the methods discussed above under claim 3 would most obviously have required checking provided information against a database of information. (Ex. 1018, 6:16-20)(Ex. 1002, ¶143). Such checking constitutes **checking whether information in the custom message appear[s] on a white list or [is] absent from a black list**. (Ex. 1002, ¶143). It

would have been obvious to enroll the user in the promotion based on the user being authenticated for the reasons described above under claim 6. (Ex. 1018, 15:14-21, 6:16-20)(Ex. 1002, ¶143).

Furthermore, Haggerty, via its incorporated '541 application, teaches that the user's telephone number (obtained in the SMS message) can be checked against various lists, including "state or federal do-not-call ('DNC') databases or wireless number lists." (Ex. 1008, p. 025, ¶0051)(Ex. 1002, ¶144). It would have been obvious to check the telephone number against telephone numbers that have been registered to not receive marketing communications (a **blacklist**) or wireless mobile numbers (a **whitelist**) prior to enrolling the user in the promotion to determine whether the user has been spoofed or whether the user has otherwise indicated it should not be contacted, and **enrolling the user in the promotion** only on the **basis** of such **authentication**, in order to avoid the potential fraud described by Rakavy. (Ex. 1018, 15:14-21, 6:16-20)(Ex. 1002, ¶144).

CLAIM 8

"8. The non-transitory processor-readable medium of claim 1, the code further comprising instructions to cause the click-to-text server to determine that the user mobile device is a new subscriber, the mobile device enrolled in the promotion based on the mobile device being a new subscriber."

See above, claim 3. It further would have been obvious to determine that the mobile device is a **new subscriber, the mobile device enrolled in the promotion**

based on the mobile device being a new subscriber. (Ex. 1002, ¶145). Rakavy teaches that an advertisement can use an award or contest (Ex. 1018, 3:37-49) and that the server should “check the user’s participation and any awards from contests”. (Ex. 1018, 6:16-26)(Ex. 1002, ¶145). Rakavy further teaches to prevent fraud in promotions. (Ex. 1018, 15:14-21). Based on this, it would have been obvious to check whether the user is already participating in a promotion where some incentive (a contest, discount, etc.) is offered, and to permit new enrollment upon the **determination that the mobile device is a new subscriber**, and further to ensure that the user is not enrolling in the promotion multiple times. (Ex. 1002, ¶145).

CLAIM 9

“9. The non-transitory processor-readable medium of claim 1, the code further comprising instructions to cause the click-to-text server to authenticate the mobile device based on the user data sent in response to the mobile device loading the webpage.”

See above, claim 3, under which authentication based on user data is explained. (Ex. 1002, ¶146).

CLAIM 10

“10. The non-transitory processor-readable medium of claim 1, the code further comprising instructions to cause the click-to-text server to authenticate the mobile device based on the identifier received with the custom message.”

See above, claim 3, under which authentication based on the identifier received with the custom message is explained. (Ex. 1002, ¶147).

CLAIM 19

“19. The non-transitory processor-readable medium of claim 1, wherein the mobile device is enrolled in the promotion based on a determination, by the click-to-text server that the mobile device is not an existing subscriber to the promotion based on the identifier.”

See above, claim 8. (Ex. 1002, ¶148).

Ground 3. Claims 1-2, 5, and 13-18 were obvious as in Ground 1, in further view of *Ad Serving Technology*.

Claims 1-2, 5, and 13-18 were obvious as in Ground 1, in further view of *Ad Serving Technology* (Ex. 1012).

Ad Serving Technology was not of record during the prosecution of the application leading to the '887 patent.

A. Prior Art Status

Ad Serving Technology was published in 2014 (Ex. 1002, ¶46)(Ex. 1012, p. 004)(Ex. 1030)(Ex. 1031), and is thus prior art under post-AIA 35 U.S.C. §102(a)(1).

B. Overview of the Ground

This ground challenges the claims at-issue under Ground 1, and additionally, claims 5 and 15-18 based on *Ad Serving Technology*, which is a general textbook related to Internet ad technology. (Ex. 1012, pp. 005-028)(Ex. 1002, ¶150).

Claim 15 requires that the integration tag include code to cause the first

application to display the promotional message in an **iframe**. An iframe was a well-known HTML tag in the relevant timeframe frequently used to display ads, as documented in *Ad Serving Technology*. (Ex. 1012, pp. 191, 442)(Ex. 1002, ¶151). It thus would have been obvious to use an iframe for its well-known purpose described in *Ad Serving Technology*. (Ex. 1002, ¶151).

Each of claims 5 and 16-18, in turn, requires the integration tag be configured to execute in a certain way: by distinguishing between mobile and desktop devices (claim 5) or by using business rules to decide whether to display the ad (claims 16-18). *Ad Serving Technology* teaches that ad tags were well-known to contain JavaScript that could determine whether or not to display the ad, and provides certain well-known business rules, thus providing motivation for building known business rules into ad tag code in the combination. (Ex. 1012, p. 054)(Ex. 1002, ¶152).

C. Reasonable Expectation of Success

A PHOSITA would have had a reasonable expectation of success in using the teachings of *Ad Serving Technology* to use iframes for the presentation of promotional messages and ad tag JavaScript to implement business rules. (Ex. 1002, ¶155).

D. Analogous Art

Ad Serving Technology, as a general textbook describing Internet techniques for Advertising was in the same field as the '887 patent and would have been

reasonably pertinent to the problems facing the named inventors. (Ex. 1012, pp. 005-028)(Ex. 1002, ¶156).

The combination and rationales therefor will be further explained in the claim mapping section below.

E. Claim Mapping

CLAIM 5

“5. The non-transitory processor-readable medium of claim 1, the code further comprising instructions to cause the click-to-text server to define the integration tag, the integration tag configured to be executed by mobile devices and not desktop computers or laptop computers.”

The combination renders claim 5 obvious. Regarding the **click-to-text server defining the integration tag**, it would have been obvious to do so for the reasons provided under Ground 1, claim 1, limitation [1b], namely the server serving many customers at scale. (Ex. 1002, ¶156). Having the click-to-text server define the integration tag allows quicker deployment and exact structuring of the tag to deliver user data in the form and to the location needed by the click-to-text server. (Ex. 1002, ¶156). It further would have been obvious to have the integration tag **configured to be executed by mobile devices and not desktop computers or laptop computers** given the purpose of Haggerty to present ads with links that will pre-populate SMS messages (*see above*, Ground 4, claim 1, limitation [1f]), and the fact that mobile devices, not desktop computers or laptop computers, typically have

SMS apps that can respond to such links. (Ex. 1002, ¶156). *Ad Serving Technology* adds to this by teaching that different ads are appropriate for mobile devices versus other devices (Ex. 1012, p. 282), and that JavaScript can “easily” distinguish between mobile and non-mobile users. (Ex. 1012, pp. 282-283)(Ex. 1002, ¶156). Because Demsey’s **integration tags** include JavaScript, it would have been obvious to use this JavaScript to determine whether the user is using a mobile device, and to only present ads intended for mobile devices when mobile devices are being used. (Ex. 1002, ¶156).

CLAIM 15

“15. The non-transitory processor-readable medium of claim 1, wherein the integration tag includes code to cause the first application to display the promotional message in an iframe.”

Ad Serving Technology teaches that iframes were a known technology for displaying ads in the relevant timeframe. (Ex. 1012, pp. 442, 191)(Ex. 1002, ¶157). A PHOSITA would have understood how to use iframes predictably, and would have found it obvious to use an iframe to display a promotional message (ad) because this was a known technique for displaying ads that would have been implementable without unpredictable results. (Ex. 1002, ¶157). *See KSR*, 550 U.S. at 417-21.

CLAIM 16

“16. The non-transitory processor-readable medium of claim 1, wherein the integration tag includes code to cause the first application to display the promotional message in response to a

business rule being met.”

See above, claim 5, where the **business rule** executed in the **integration tag** to determine if the **promotional message is displayed** is that the ad should be displayed when the user is using a mobile device. (Ex. 1002, ¶158).

Furthermore, it would have been obvious based on *Ad Serving Technology* to cause the integration tag to execute a business rule that only shows a promotional message when a user has not seen the promotional message within a certain prior time period, because *Ad Serving Technology* teaches that the system should limit the frequency of ad display to users. (Ex. 1012, pp. 131-132)(“[f]requency caps are usually set to limit the user’s exposure to just a campaign to a handful of ‘impressions’ over a given time period of twenty four hours or less.”)(Ex. 1002, ¶159). *Ad Serving Technology* likewise teaches using JavaScript in ad tags to control when ads should be displayed. (Ex. 1012, p. 054)(“JavaScript code allows the ad to be pulled directly into a Publisher’s web page from the third party ad server and make decisions about what ad to display in real time.”)(*see also* pp. 051-054)(Ex. 1002, ¶159). It would have been obvious from these disclosures to **include code** in the integration tag **to cause the first application to display the promotional message in response to a business rule being met**, *e.g.* limiting views based on recent prior viewing by the same user. (Ex. 1002, ¶159).

CLAIM 17

“17. The non-transitory processor-readable medium of claim 16, wherein the business rule is at least one of a whitelist, a blacklist or a timeout.”

See above, claim 16, where preventing re-serving of an ad within a given timeframe is the **timeout**. (Ex. 1012, p. 132)(Ex. 1002, ¶160).

“18. The non-transitory processor-readable medium of claim 16, wherein the business rule is that a predefined event occurred within a predefined period of time before the integration tag was executed.”

See above, claim 17. In the case of claim 18, the business rule is a rule that an ad was not served more than a specified number of times (the “frequency cap”) within a prior, predefined period of time (e.g. 24 hours). (Ex. 1012, p. 132)(Ex. 1002, ¶161).

Ground 4. Claims 1-2, 11-14, and 20 were obvious as in Ground 1, in further view of Molinet.

Claims 1-2, 11-14, and 20 were obvious as in Ground 1, in further view of U.S. Pat. App. Pub. 2016/0142858 A1 (“**Molinet**”)(Ex. 1009).

Molinet was not of record during the prosecution of the application leading to the ’887 patent.

A. Prior Art Status

Molinet was published on May 19, 2016 from an effective filing date at least

as early as Nov. 6, 2015, and is thus prior art under post-AIA 35 U.S.C. §§102(a)(1) and (2).

B. Overview of the Ground

This Ground challenges the claims at-issue under Ground 1, and additionally, claims 11-12 and 20. Each of claims 11-12 and 20 requires that the URI that is sent from the click-to-text server to the mobile device be customized for the operating system of the mobile device (or simply for the mobile device as in claim 20). Molinet teaches this feature. (Ex. 1002, ¶163).

C. Overview of Molinet

Molinet teaches a system that can use mobile app deeplinking with “contextual deeplinks”. (Ex. 1009, ¶¶0016-0018)(Ex. 1002, ¶164). “Contextual deeplinks” are app deeplinks that are customized for the particular operating environment of a user’s mobile device. (Ex. 1009, ¶0037)(Ex. 1002, ¶164). This customization is desirable, according to Molinet, because different operating environments may not use or allow all deeplinking protocols. (Ex. 1009, ¶0005)(Ex. 1002, ¶164). Molinet further incorporates (Ex. 1009, ¶0001) its provisional application, U.S. Prov. App. 62/079,512 (Ex. 1010), which is relevant to certain aspects of claims 11-12. For example, the ’512 provisional teaches the use of standard URIs for mobile app deeplinking used by the iOS and Android operating systems. (Ex. 1010, pp. 021, 015)(Ex. 1002, ¶164).

D. Reasonable Expectation of Success

A PHOSITA would have had a reasonable expectation of success in using the teachings of Molinet in the manner described herein for the reasons described in Ground 1. (Ex. 1002, ¶166).

E. Analogous Art

Molinet is analogous art because it is in the same field as the '887 patent and is reasonably related to the problems facing the named inventors. (Ex. 1009, ¶0003)(Ex. 1002, ¶167).

F. Claim Mapping

CLAIM 11

“11. The non-transitory processor-readable medium of claim 1, wherein: the user data includes an indication of an operating system of the mobile device; the messaging application is a default messaging application for the operating system; and the URI is customized for the operating system such that the URI deeplinks to the default messaging application.”

The combination renders claim 11 obvious. First, Molinet, like Demsey, teaches collecting user data (as discussed above under claim 1, element [1c]) using JavaScript on a webpage. (Ex. 1009, ¶0065). The user data includes **an indication of an operating system of the mobile device**. (Ex. 1009, ¶0065)(“These characteristics may include network address (*e.g.*, IP address), operating system version, operating system type, device model information, and screen size.”)(*see also* Ex. 1009, ¶¶0017, 0025, 0037, 0054, 0069, 0085, 0105)(Ex. 1002, ¶168).

Molinet further teaches that its system **customizes the URI** (as discussed above under claim 1, element [1d]) based on the user data, including the operating system, such that the URI is operable on the particular mobile device. (Ex. 1009, ¶0037)(*see also* Ex. 1009, Abstract, Fig. 5, rn 550-560, ¶¶0018, 0037, 0065, 0080, 0107)(Ex. 1002, ¶169). Note that it is clear from Molinet that the “operating system” type and version are part of the contextual information used to ensure that an operable response (including a URI) is sent to the client device. (Ex. 1009, ¶¶0065, 0037, 0017, 0025, 0054, 0069, 0085, 0105)(Ex. 1002, ¶169).

Molinet’s incorporated provisional application (the ’512 provisional) furthermore describes direct deeplinking URI formats built into iOS and Android operating systems. (Ex. 1010, p. 021)(Ex. 1002, ¶170). It would have been obvious to use such URI formats to implement the deeplinking of Khanna, where the particular direct deeplink is chosen according to the operating system of the mobile device. (Ex. 1010, pp. 021-022, 015)(Ex. 1002, ¶170). As discussed in the Introduction, above, the deeplinking URI formats were also widely known for SMS deeplinking in the relevant timeframe. (Ex. 1002, ¶170).

Furthermore, it would have been obvious to configure the URI to deeplink to the default messaging application, such that the **messaging application is a default messaging application for the operating system**. Specifically, Haggerty teaches using a text-messaging (SMS) application, and in the relevant timeframe, most

smartphones came with a default text messaging (SMS) app, making that the most obvious choice (the app that comes with the phone is likely to be installed, whereas with a non-default app there is a risk that the app is not installed (Ex. 1006, ¶0046)). (Ex. 1002, ¶171). Furthermore, there were already standard URI formats available to deeplink to default SMS apps (including sms: and smsto:), making the default app the most obvious choice. (Ex. 1010, pp. 015, 021)(Ex. 1019, p. 010)(Ex. 1002, ¶171).

CLAIM 12

“12. The non-transitory processor-readable medium of claim 1, wherein: the user data includes a user agent associated with the first application; and the messaging application is a default messaging application for an operating system of the mobile device, the code further comprising instructions to cause the click-to-text server to: identify the operating system of the mobile device based on the user agent, the URI customized for the operating system such that the URI deeplinks to the default messaging application.”

See above, claim 11. Claim 12 differs from claim 11 in that the user data includes a user agent associated with the first application, and the software identifies the operating system of the mobile device based on the user agent. As Dr. Houh explains, “user agent” refers to any of a variety of data about the user’s device and software that is available to webpages and servers interacting with the user’s device. (Ex. 1002, ¶172).

Molinet teaches that **user agent** information can be collected as part of the user data. (Ex. 1009, ¶0085)(Ex. 1010, p. 022)(Ex. 1002, ¶173). As Molinet’s

incorporated provisional explains, the user agent information sent by a browser refers to the “User-Agent” field of an HTTP request, which included an identification of the operating system, operating system version, and name and version of the browser itself. (Ex. 1010, p. 022)(Ex. 1009, ¶0065)(Ex. 1002, ¶173). It would have been obvious to collect user agent information as “user data” sent together with the ad request of Demsey (*see* above, Ground 1, claim 1, element [1c]), both because the user agent data is sent with the HTTP header in the ad request (Ex. 1010, p. 022)(Ex. 1005, Fig. 5, ¶0045), and because the user agent data could be accessed using JavaScript executed by a browser by using, for example, the Navigator object’s userAgent field. (Ex. 1002, ¶173)(*citing* Ex. 1021, pp. 364-365, 992). In this way, it would have been obvious to **identify the operating system of the mobile device based on the user agent** (to configure the URI, as explained under claim 11), because the user agent states the identity of the operating system directly. (*Id.*). (Ex. 1002, ¶173). Claim 12 is otherwise obvious for the same reasons as claim 11. (Ex. 1002, ¶173).

CLAIM 20

“20. The non-transitory processor-readable medium of claim 1, wherein the URI is customized for the mobile device based on the user data.”

See above, claim 11. (Ex. 1002, ¶174). If a URI is customized for the operating system of the mobile device, it is **customized for the mobile device**. (Ex.

1002, ¶174).

IV. SECONDARY CONSIDERATIONS

Petitioner is not aware of any secondary considerations that would support an inference of non-obviousness. In a complaint for patent infringement, Patent Owner alleged (Ex. 1022, ¶51) that the patent has “garnered industry praise”, citing three articles (submitted as Exs. 1023-1025). Two of the articles, however, simply quote or characterize statements of the Patent Owner itself (Exs. 1024 and 1025). A third article calls “two-tap” technology “novel”, which, as explained in the introduction, it clearly was not. (Ex. 1011, ¶¶0029, 0034, 0048, 0007)(Ex. 1007, 10:1-5, 9:48-55, 13:12-18)(Ex. 1013, ¶¶0039-0042)(Ex. 1017, ¶¶0026, 0069, 0088). *See Iron Grip Barbell Co. v. USA Sports, Inc.*, 392 F.3d 1317, 1324 (Fed. Cir. 2004). In an opposition to a motion to dismiss, Patent Owner also alleged copying (Ex. 1014, p. 3), but provided no details, citing only to an irrelevant paragraph in its complaint. Patent Owner has thus not made a case for secondary considerations. (Ex. 1002, ¶¶175-177).

V. DISCRETIONARY INSTITUTION

A. The Board should not deny the petition under 35 U.S.C. §325(d)

The Board should not exercise its discretion to deny the petition under 35 U.S.C. §325(d). *See Advanced Bionics, LLC v. MED-EL Elektromedizinische Geräte G.m.b.H.*, IPR2019-01469, Paper 6, *8-9 (PTAB Feb. 13, 2020) (precedential). The principal references of Grounds 1-4, namely Demsey, Khanna,

Haggerty, Rakavy, *Ad Serving Technology*, and Molinet, were not of record during prosecution of the application leading to the '887 patent. *See Advanced Bionics* at *8-9.

Furthermore, the Examiner in the application that issued as the '887 patent appears to have clearly misapprehended the prior art's teachings. Specifically, as discussed above in §I.C., the Examiner allowed the claims because the prior art allegedly did not teach "the interaction of the tag and user causing an enrollment with a de minimis amount of information having been provided". (Ex. 1004, p. 041). However, the Haggerty, Oliver, Hsu and Gupta references cited in the Background demonstrate that the "two-tap" method was well-known, and would have been highly relevant to this understanding of the claim scope. (Ex. 1011, ¶¶0029, 0034, 0048, 0007)(Ex. 1007, 10:1-5, 9:48-55, 13:12-18)(Ex. 1013, ¶¶0039-0042)(Ex. 1017, ¶¶0026, 0069, 0088)(Ex. 1002, ¶¶31-38).

B. The Board should not deny the petition under 35 U.S.C. §314(a)

Nor should the Board deny the petition under 35 U.S.C. §314(a). *See Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 (PTAB Mar. 20, 2020) (precedential)(*"Fintiv"*).

The *Fintiv* factors weigh in favor of institution:

Factor 1. Postscript intends to move for a stay pending *inter partes* review, should the review be instituted.

Factor 2. The district court case against Postscript is still in early stages. No trial date has been set, only a tentative trial month (March 2025), and even that is subject to the Court's availability. (Ex. 1027, p. 021). The District of Delaware as a whole, for all civil trials, currently has a median time of 36.7 months to trial, which would result in a trial in February 2026. (Ex. 1026, p. 014). It is thus likely that the Board will reach a decision prior to any trial on the merits in district court. Petitioner also filed this Petition expeditiously after receiving the Patent Owner's infringement contentions on June 29, 2023.

Factor 3. Because the district court case against Postscript is early stages, the parties and the court have invested very little in the proceeding. There have been no motions for summary judgment of invalidity based on prior art, no expert reports, and no Markman briefing at the time of filing of this petition. (Ex. 1027).

Factor 4. Because the district court case against Postscript is early stages, it is too early to say what the extent of overlap at trial will be. Postscript has asserted each of the references at issue here in its invalidity contentions, among others.

Factor 5. Postscript is a party only to the parallel litigation filed in the District of Delaware on Jan. 25, 2023.

Factor 6. The merits of the petition are compelling, and the Board should institute on this basis alone.

VI. CONCLUSION

Petitioner thus respectfully requests that claims 1-20 of the '887 patent be canceled.

Date: <u>August 19, 2023</u>	<u>/Matthew A. Smith/ (RN 49,003)</u> Matthew A. Smith SMITH BALUCH LLP 700 Pennsylvania Ave. SE, 2nd Floor Washington, DC 20003 (202) 669-6207 smith@smithbaluch.com <i>Counsel for Petitioner</i>
------------------------------	--

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the foregoing petition for *inter partes* review, together with all exhibits and other documents filed therewith, was served by USPS Priority Mail - Express on August 19, 2023 on the Patent Owner's counsel of record at the United States Patent & Trademark Office having the following address:

COOLEY LLP
ATTN: IP Docketing Department
1299 Pennsylvania Avenue, NW
Suite 700
Washington DC 20004
UNITED STATES

Date: August 19, 2023

/Matthew A. Smith/ (RN 49,003)

CERTIFICATE OF WORD COUNT

The undersigned hereby certifies that the foregoing petition for *inter partes* review contains 13,913 words according to the word processing program used to prepare it.

Date: August 19, 2023

/Matthew A. Smith/ (RN 49,003)