

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

METARAIL, INC.,
Patent Owner.

Case No. IPR2024-01271
Patent No. 10,262,342

**PETITION FOR INTER PARTES REVIEW
UNDER 35 U.S.C. §§ 311-319 AND 37 C.F.R. § 42.1 et seq**

TABLE OF CONTENTS

Mandatory Notices.....	vii
A. Real Party-In-Interest.....	vii
B. Related Matters.....	vii
1. United States Patent & Trademark Office	vii
2. U.S. District Court for the District of Delaware	vii
C. Counsel and Service Information - § 42.8(b)(3) and (4).....	viii
I. Introduction	1
II. Requirements for IPR	4
A. Grounds for Standing (37 C.F.R. § 42.104(a))	4
B. Specific Art on Which the Challenge is Based.....	5
C. Identification of the Challenge (37 C.F.R. §42.104(b))	5
III. Background.....	5
A. The '342 patent.....	5
B. Prosecution History.....	8
IV. Level of Ordinary Skill in the Art	8
V. Claim Interpretation.....	9
VI. Ground 1: Claims 1-21 are obvious over Belanger in view of Halevy.....	9
A. Belanger – Parameter-Based Deep Linking.....	9
1. Belanger’s Creation and Serving of Deep-Links	15
B. Halevy	17
1. Halevy Automatically Maps Data Fields of Website Forms for Creating Deep-links	18
2. Universal Variable Map (Form Database).....	19
3. Retrieving Deep-Links from the Form Database.....	22
C. POSA’s Motivations to Modify Belanger in view of Halevy	25
D. Claim-by-Claim Analysis	33
1. Claim 1	33

2. Claim 2: “The method according to claim 1, further comprising: applying, by the spidering module via programming code, a set of enhancing rules to enhance results of the obtaining of the computer-readable field identifiers.”	52
3. Claim 3: “The method according to claim 1, wherein the obtaining includes gathering information pushed from the sites or applications on the Internet.”	54
4. Claim 4: “The method according to claim 1, wherein the obtaining includes running a programmatic query on a site or application, receiving a response page from the site or application responsive to the programmatic query, and parsing the response page to obtain a set of computer-readable field identifiers from the site or application.”	54
5. Claim 5: “The method according to claim 1, wherein the mapping is performed when the computer-readable field identifiers are obtained from the sites or applications on the Internet, when the deep-linked ad is automatically programmatically generated, or when the deep-linked ad is about to be served in real-time.”	56
6. Claim 6: “The method according to claim 1, wherein the spidering module parses forms from the sites or applications on the Internet and form data associated with the forms, identifies any product associated with the forms, and obtains product attributes from the form data for indexing.”	57
7. Claim 7: “The method according to claim 1, wherein the first application and the second application are in same vertical or different verticals for which the spidering module is configured to search on the Internet.”	58
8. Claims 8-21	59
VII. Ground 2: Claims 1-21 are obvious over Belanger and APA in view of Halevy	62
VIII.No Basis Exists for Discretionary Denial	64
A. Section 314(a).....	64
1. Stay Potential.....	64
2. Trial Timing.....	64

3. Litigation Involvement.....	65
4. Issue Overlap	65
5. Litigation Defendants.....	65
6. Other Circumstances	65
B. Section 325(d).....	66
IX. Conclusion.....	67
X. Claim Listing	68

TABLE OF AUTHORITIES

CASES

<i>Advanced Bionics, LLC v. MED-EL Elektromedizinische Geräte GmbH</i> , IPR2019-01469, Paper 6 (PTAB Feb. 13, 2020).....	67
<i>Apple Inc. v. Fintiv, Inc.</i> , IPR2020-00019, Paper 11 (Mar. 20, 2020) (precedential)	64
<i>Markforged Inc. v. Continuous Composites</i> , IPR2022-00679, Paper 7 (Oct. 25, 2022).....	65
<i>Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.</i> , 868 F.3d 1013 (Fed. Cir. 2017).....	9
<i>Samsung Elecs. v. Netlist, Inc.</i> , IPR2022-00063, Paper 13 (May 5, 2022)	65
<i>Sand Revolution II, LLC, v. Cont’l Intermodal Group–Trucking</i> , IPR2019-01393, Paper 24 (June 16, 2020)	64, 65

REGULATIONS

37 C.F.R. § 42.100(b)	9
37 C.F.R. § 42.104(a).....	4
37 C.F.R. § 42.104(b)	5

STATUTES

35 U.S.C. § 102(b).....	5
35 U.S.C. § 103(a)	5
35 U.S.C. § 325(d).....	66

OTHER AUTHORITIES

Director’s Interim Procedure for Discretionary Denials (June 21, 2022).....	64, 65
MPEP § 2144.04(III)	4

APPENDIX LISTING OF EXHIBITS

Exhibit	Description
1001	U.S. Patent No. 10,262,342 (“the ’342 patent”)
1002	Prosecution History of U.S. Patent No. 10,262,342
1003	Declaration of Dr. Edward A. Fox (“Fox”)
1004	Curriculum Vitae of Dr. Edward A. Fox
1005	U.S. Pat. App. Pub. No. 2005/0033641 to Jha et al. (“Jha”)
1006	U.S. Pat. App. Pub. No. 2006/0230033 to Halevy et al. (“Halevy”)
1007	Hila Becker et al., <i>What Happens after an Ad Click? Quantifying the Impact of Landing Pages in Web Advertising</i> , Conference on Information and Knowledge Management 2009 (Nov. 2, 2009) (“Becker”)
1008	U.S. Pat. App. Pub. No. 2005/0144048 to Belanger et al. (“Belanger”)
1009	Reserved
1010	Docket Sheet, <i>Metarail, Inc. v. Google LLC</i> , No. 1:23-cv-01116 (D. Del.)
1011	United States District Courts — National Judicial Caseload Profile, <i>available at</i> https://www.uscourts.gov/sites/default/files/fcms_na_appprofile0630.2024.pdf
1012	First Amended Complaint, <i>Metarail, Inc. v. Google LLC</i> , No. 1:23-cv-01116 (D. Del. Jan. 12, 2024), ECF No. 15
1013	Michael Uschold, <i>Ontology and database schema: What’s the difference?</i> , <i>Applied Ontology</i> , vol. 10, no. 3-4, pp. 243-258 (2015) (“Uschold”)
1014	U.S. Pat. App. Pub. No. 2005/0119921 to Fitzgerald et al. (“Fitzgerald”)
1015	Trip Advisor Press Release, “New Advertising Technology Delivers Results for Travel Marketers” (published Dec. 3, 2001) (“TripAdvisor”)

1016	U.S. Patent Pub. No. 2007/0239494 to Stephens et al. (“Stephens”)
1017	Prosecution history of U.S. Patent No. 9,633,378
1018	Doug Tidwell, et al., Programming Web Services with SOAP (1st ed. 2001) (“Tidwell”)
1019	Reserved
1020	Reserved
1021	Exhibit C of First Amended Complaint, Metarail, Inc. v. Google LLC, No. 1:23-cv-01116 (D. Del. Jan. 12, 2024), ECF No. 15
1022	Exhibit G of First Amended Complaint, Metarail, Inc. v. Google LLC, No. 1:23-cv-01116 (D. Del. Jan. 12, 2024), ECF No. 15
1023	Exhibit I of First Amended Complaint, Metarail, Inc. v. Google LLC, No. 1:23-cv-01116 (D. Del. Jan. 12, 2024), ECF No. 15

MANDATORY NOTICES

A. Real Party-In-Interest

Petitioner is the Real Party-in-Interest.¹

B. Related Matters

A decision in this proceeding could affect or be affected by the following:

1. United States Patent & Trademark Office

The following U.S. patent applications claim priority to U.S. Patent No. 10,262,342 (“the ’342 patent”):

U.S. Patent Application No. 16/933,053, filed July 20, 2020.

The following *Inter Partes* Review proceedings involve patents in the same family as the ’342 patent:

IPR2024-01269 challenging U.S. Patent No. 10,789,626;

IPR2024-01270 challenging U.S. Patent No. 9,633,378; and

IPR2024-01272 challenging U.S. Patent No. 10,152,734.

2. U.S. District Court for the District of Delaware

A decision in this proceeding could affect or be affected by the following:

Metarail, Inc. v. Google LLC, Case No. 1:23-cv-01116 (DDE).

¹ Google LLC is a subsidiary of XXVI Holdings Inc., which is a subsidiary of Alphabet Inc. XXVI Holdings Inc. and Alphabet Inc. are not real parties-in-interest to this proceeding.

C. Counsel and Service Information - § 42.8(b)(3) and (4)

Lead Counsel	Scott A. McKeown, Reg. No. 42,866
Backup Counsel	Anant K. Saraswat, Reg. No. 76,060 Michael S. Parsons, Reg. No. 58,767 Gregory F. Corbett, pending admission pro hac vice
Service Information	E-mail: SMcKeown-PTAB@WolfGreenfield.com ASaraswat-PTAB@WolfGreenfield.com MParsons-PTAB@WolfGreenfield.com Gregory.Corbett@WolfGreenfield.com <u>Post and hand delivery:</u> Wolf, Greenfield & Sacks, P.C. 600 Atlantic Avenue Boston, MA 02210-2206 <u>Telephone:</u> 617-646-8000 <u>Facsimile:</u> 617-646-8646

A power of attorney is submitted with the Petition. Counsel for Petitioner consents to service of all documents via electronic mail.

I. INTRODUCTION

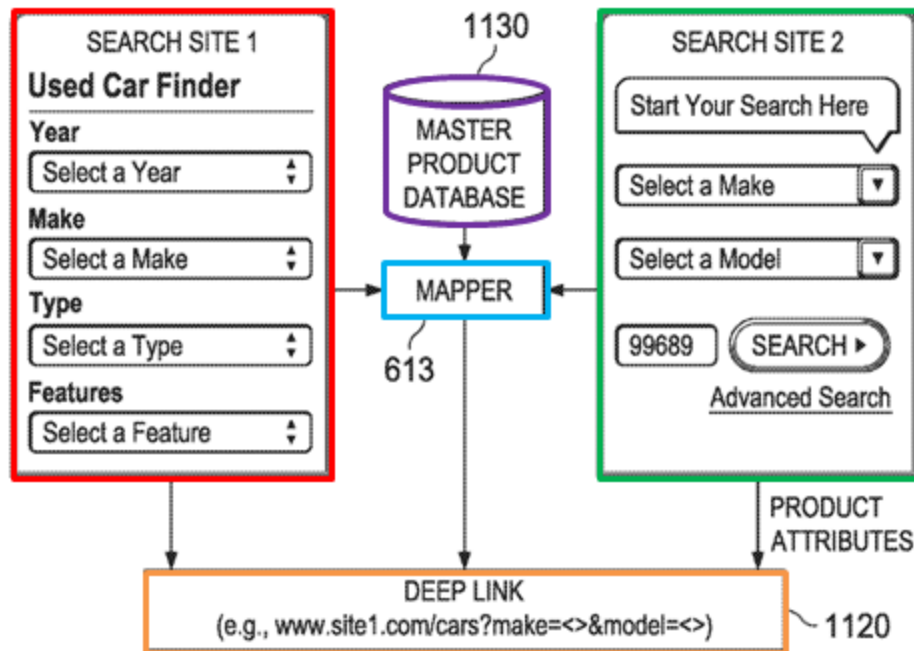
Google LLC (“Petitioner”) requests *inter partes* review and cancellation of claims 1-21 (the “challenged claims”) of U.S. Patent No. 10,262,342 (“the ’342 patent”) (Ex-1001).

The ’342 patent is directed to a deep-linking system for “automatically generat[ing]” online ads. Ex-1001, Abstract. By way of background, the ’342 patent acknowledges that “deep-linked” or “parameter-based” digital advertising technologies were known at the time of filing. *Id.*, 2:49-63. In such prior art systems, parameters provided by a user to one site (e.g., travel location and dates) are “passed along with each ad click to the appropriate advertiser.” *Id.*, 2:55-57. Advertisers use these passed-along parameters, for example, to “automatically activate searches” from the advertiser’s homepage, “thereby landing the web user ‘deep’” into a website rather than on the advertiser’s homepage. *Id.*, 2:57-63; Fox, ¶35.

The ’342 patent alleges that the problem with prior “deep-linked” or “parameter-based” ad systems was that the mapping between the parameter fields in a publisher’s site and the corresponding fields in an advertiser’s site had to be “manually created by developing programmatic integrations between” the fields on the publisher’s site and the fields on the advertiser’s site (e.g., “by manual code”).

Ex-1001, 3:2-16. The '342 patent simply purports to automate these mappings. *Id.*, Abstract, 1:33-37, 3:32-33, 3:44-51, 4:15-17; Fox, ¶36.

As shown in Fig. 11A below, the '342 patent claims a “universal variable map” 613 that is formed in a memory by crawling Internet websites to identify website form data. *Id.*, 8:49-62. Once a form is identified, the crawler “collects the names of variables and identifiers” on the form. *Id.*, 9:47-51. The names of the variables or identifiers “are then archived in” the “universal variable map.” *Id.*, 9:66-67. The universal variable map also includes “normalized variables” that are “linked” to the names of the variables and identifiers obtained from the crawled forms found on various sites to allow for the automated creation of deep links (e.g., from a Search Site 2 to a Search Site 1). *Id.*, 5:18-22. In this way the universal variable “mapper” automates the previously manual variable link process.



The '342 patent issued without substantive discussion in the examination record as to prior art. *See* Ex-1002, 200-01. Yet, as the '342 patent concedes, serving a parameter based deep-link on one site to repurpose the parameters to a target site of the deep-link was well-known. While the '342 patent purports to have invented automating the creation of such deep-links via its universal variable map, this too was known.

As explained herein, **Belanger** (Ex-1008), like the prior art described in the '342 patent, discloses a server that serves deep-links for reaching another site such that parameter data included in the deep-link at a first site pre-populates parameters at a second site. Belanger's exemplary system teaches using these deep-links in support of a car rental business; nevertheless, a broader applicability is also taught and was known in the art relative to the overall travel industry.

Automating parameter gathering through web spidering/crawling and linking parameter mappings to create deep-links was also well known as disclosed by **Halevy** (Ex-1006). As such, at the time of the '342 patent filing, automating the manual process of gathering and linking parameters to create deep-links across websites was obvious to a POSA to avoid tedious manual coding of such link associations. Ex-1001, 3:12-21 (evidencing knowledge known in the art). Indeed, automating a manual process is *de facto* obvious. MPEP § 2144.04(III). Moreover, automating this formerly manual parameter association and parameter linking offers greater scalability for deep-link creation to account for expanding business partners of a subject site, and was an emerging value add to travel industry booking sites. That is, travel data such as date range and location is ideally leveraged across travel reservation sites (e.g., car rental, hotel, and airline reservation sites). *Infra* §§ VI.A and VI.C.

Accordingly, Petitioner respectfully requests *Inter Partes* Review of claims 1-21 be instituted, and these claims be canceled as unpatentable.

II. REQUIREMENTS FOR IPR

A. Grounds for Standing (37 C.F.R. § 42.104(a))

Petitioner certifies that the '342 patent is available for IPR and Petitioner is not barred or estopped from requesting IPR as to the challenged claims.

B. Specific Art on Which the Challenge is Based

Petitioner relies on the following prior art:²

Name	Ex.	Filing Date	Issue/Pub. Date	pre-AIA §
Halevy	1006	2006-04-05	2006-10-12	102(b)
Belanger	1008	2005-01-20	2005-06-30	102(b)

C. Identification of the Challenge (37 C.F.R. §42.104(b))

Petitioner requests IPR of the challenged claims and that the Board cancel the same as unpatentable on the following grounds:

Ground Number and Reference(s)		Claims	Basis
1	Belanger in view of Halevy	1-21	Pre-AIA §103(a)
2	Belanger & APA in view of Halevy	1-21	Pre-AIA §103(a)

III. BACKGROUND

A. The '342 patent

The '342 patent is directed to automatically generating deep-linked ads via the use of a so-called “universal variable map.” Ex-1001, Abstract. As discussed above, the universal variable map is a collection of the “names” of the variables

² Petitioner does not challenge the priority chain of the '342 patent as the art predates the earliest date, but reserves the right to do so in other proceedings.

and identifiers of the fields on web forms found by crawling Internet websites and applications. *Id.*, 8:49-62; 9:47-59. These “names” are stored in the universal variable map and associated with “normalized variables.” *Id.*, 5:18-22, 9:66-10:4. These normalized variables serve as links between the fields of a first site and corresponding fields of a second site that were previously correlated manually to create parameter-value-passing deep-links. *Id.*, 5:18-22; Fox, ¶35.

Figure 11A (annotated below) illustrates the process for using the universal variable map in generating deep-linked ads for a user visiting a car search website (e.g., **Search Site 2**). Here, the **universal variable mapper 613** is used “to provide to its visitors a deep-linked ad for **Search Site 1**,” which is a search site for a “Used Car” in the same “vertical,” (i.e., website category):

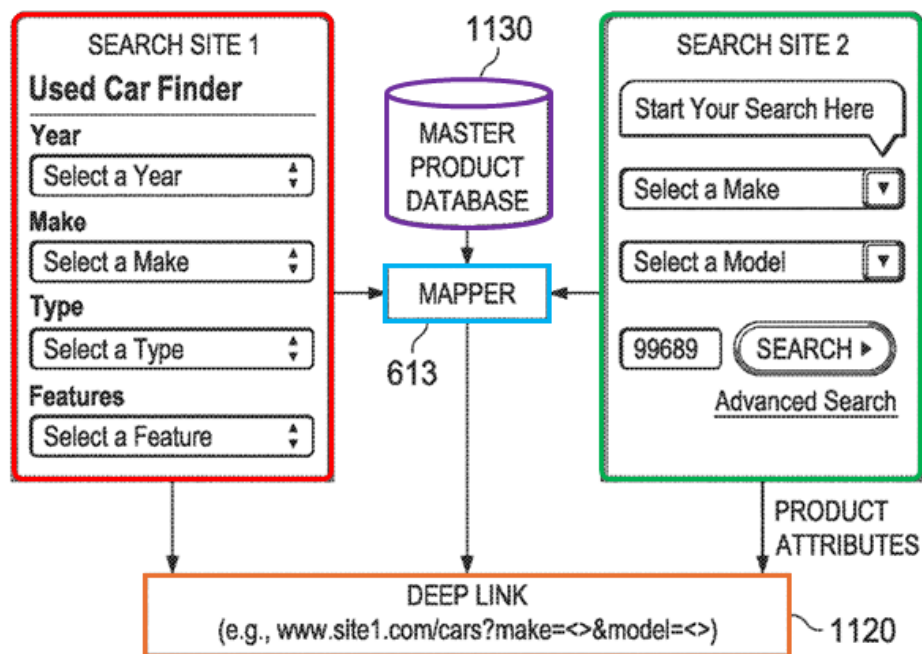


FIG. 11A

Ex-1001, 13:66-67; Fox, ¶37. To generate the deep-linked ad, [mapper 613](#) accesses the universal variable map of [Master Product Database 1130](#), “cross reference[s] programming variables³ entered into [Search Site 2](#) with those used by [Search Site 1](#), and automatically create[s] a deep-linked ad ([deep-link 1120](#) in this example).”

Ex-1001, 13:64-14:10. The [deep-linked ad](#) is placed on [Search Site 2](#) so that when a user selects the ad, “the user’s browser is automatically directed to a network address indicated by” the [deep-link 1120](#) to [Search Site 1](#), which provides content corresponding to the parameter data entered on [Search Site 2](#); for example, in directing the user’s browser to the deep link location, the form at [Search Site 1](#) may “be pre-populated with ... any user input provided at [Search Site 2](#),” for example, user entries in the field selectors such as “Toyota” and “Prius.” *See id.*, 14:10-16; Fox, ¶37.

The purported improvement that the above-described process provides over the prior art is the automating of the manual process of “mapping” programming variables (i.e., field identifiers) used by one site to corresponding programming variables on another site. Ex-1001, 3:22-23. The automation of this previously

³ The ’342 patent uses the terms “programming variable,” “variable,” “field identifier,” and “identifier” interchangeably. *See* Ex-1001, 4:27-28, 9:26-33, 9:47-51, 13:32-36.

manual process involves the universal variable map that stores a collection of website data organized by content (i.e., “verticals”). *Id.*, 8:32-45, Figs. 4, 5. As described in Figure 11A above, this data is used to automatically generate a deep-link that maps the data a user entered in form fields of a first website to the variables or field identifiers of corresponding fields of a second website. *Id.*, 13:64-14:14; Fox, ¶38.

B. Prosecution History

The '342 patent issued after Applicants filed a terminal disclaimer to overcome the only rejection entered by the Examiner, finding that the claims were not patentably distinct over the claims of its parent (U.S. 9,633,378). *See* Ex-1002, 198, 286-87, 323-28. No substantive prior art rejections were applied.

IV. LEVEL OF ORDINARY SKILL IN THE ART

The '342 patent relates to web-based e-commerce. A person of ordinary skill in the art (“POSA”) would have had a bachelor’s degree in computer science or a related field, and at least two years of experience with web programming and search engine systems. Fox, ¶32. More education could substitute for experience, and vice versa. This person would have been capable of understanding and applying the teachings of the '342 patent and the prior-art references discussed herein. *Id.*

V. CLAIM INTERPRETATION

Claim terms subject to IPR are to be construed in accordance with their ordinary and customary meaning as understood by a POSA in light of the specification and prosecution history. 37 C.F.R. § 42.100(b). Only terms necessary to resolve the controversy need to be construed. *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017). Petitioner reserves the right to argue different constructions in any other actions involving the '342 patent.

VI. GROUND 1: CLAIMS 1-21 ARE OBVIOUS OVER BELANGER IN VIEW OF HALEVY

A. Belanger – Parameter-Based Deep Linking

Belanger (Ex-1008) discloses “a reservation booking website” that uses “promotional deep-links” to assist users in booking car reservations. *Id.*, [0019], [0128], [0168-69], Fig. 37 (showing promotional deep-links 318). The website’s booking page includes various fields into which the user can enter data related to the reservation, such as pickup and return dates, pickup location, and requested vehicle type. *Id.*, [0165], Fig. 37 (below, annotated)

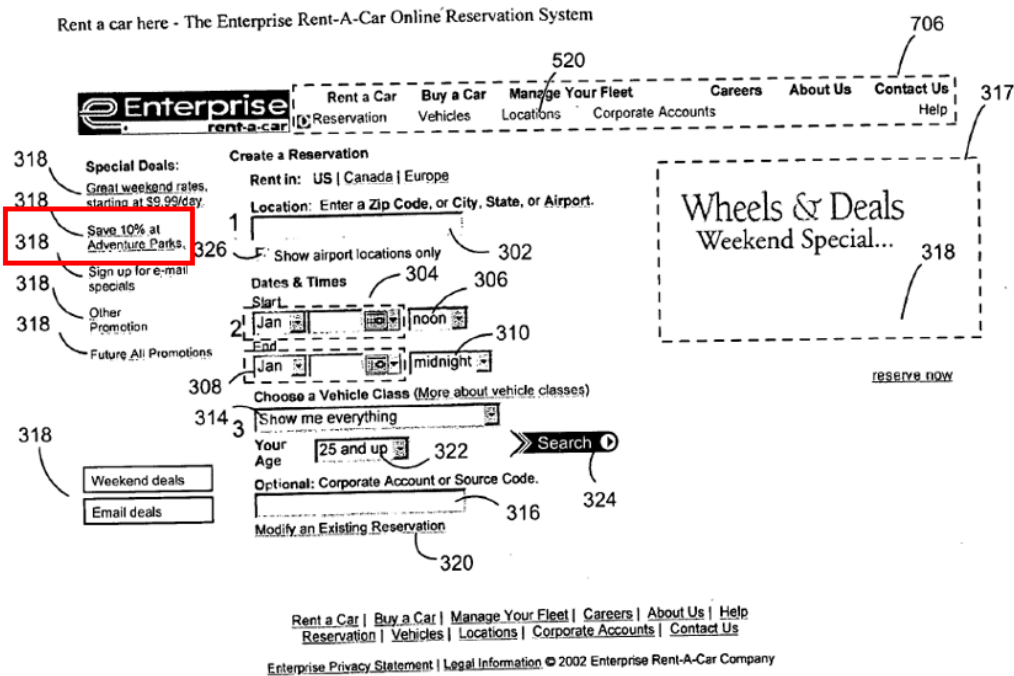


Fig. 37

Belanger explains that a “promotional deep-link” corresponds to a promotion with “one or more promotion conditions,” where each condition “correspond[s] to a particular data value or range of data values that the customer must choose for a particular reservation data type.” Belanger, [0019]; [0168-69]. The promotional deep-link directs the user to a webpage where “any reservation data types corresponding to promotional conditions ... have their data values set equal to those conditions/parameters.” Belanger, [0019]; *compare with* Ex-1001, 4:10-14, 14:10-15 (deep-links can take the user to a page with a “form” that is “pre-populated with field identifier values”). These promotional deep-links may be used in various ways, including being advertised on other websites. *Id.*, [0168]. These deep-links beneficially “provide[s] single click deep-link access to the

reservation booking website.” *Id.*, [0178]; *see also* Fitzgerald (Ex-1014) (parent of Belanger, incorporated by reference, and Titled: “Method and Apparatus for Customer Direct On-Line Reservation of Rental Vehicles Including Deep-Linking.”); Fox, ¶¶46-47.

Belanger’s Figure 53 (below, annotated) shows an example of “promotional link 318” on a confirmation web page that is presented upon completion of the booking process. This link 318 offers a discount at an amusement park attraction. *Id.*, [0128], Fig. 53; Fox, ¶48.

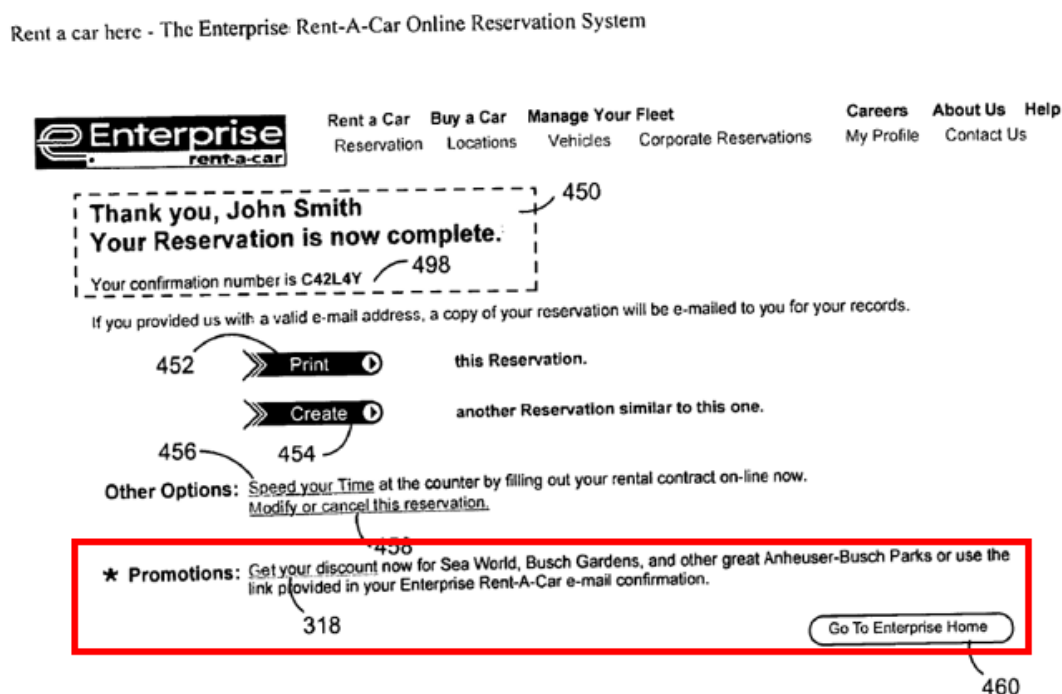


Fig. 53

POSAs would have understood links 318 of Fig. 53 to be deep-links given the common descriptor “promotional links 318” used by Belanger to describe these confirmation page links. Belanger, [0128] (describing links 318 that appear on

various webpages and are “selectable to initiate a reservation transaction according to the conditions of the promotion”), Fig. 37; Fox, ¶49. Additionally, given the value of deep-linking to yield “a large degree of user-friendliness in that the user is alleviated of the need to remember,” a POSA would have nevertheless been motivated to provide this same accustomed benefit of deep-linking to its customers for any outgoing links of booking sites having similar date and location fields. *See, e.g.,* Belanger, [0179], [00168-69], [0020]; Fox, ¶49. By providing outgoing deep-links that repurpose form data such as the date and location of a car rental, a user could be likewise advanced to a particular “stage” in the booking process of an amusement park, airline, hotel, etc. website. Fox, ¶49. For example, POSAs would have recognized that when a customer is renting a car in a specific geographic location, and/or on specific dates, a geographically local promotion with a date/time condition within the user’s rental period would have provided effective cross-promotion by repurposing user-entered trip parameters to prepopulate some of the same parameter values (e.g., repurposing car rental pick-up date/time and location in airline reservation travel dates and destination, hotel reservation check-in dates and location, local attraction tickets, etc.). *Id.*

Indeed the ’342 patent itself admits that:

A recent advancement in Digital Advertising has been “deep-linked” or “parameter-based” advertising. This form of advertising *today* applies to search and E-commerce sites where the user is searching for

a product or service, *like a travel site, and types in search parameters, such as where s/he is traveling from, where to, travel dates, how many people in the party, class of travel, etc. These search parameters are then passed* along with each ad click to the appropriate advertiser.

Ex-1001, 2:49-57 (emphasis added); *see also* TripAdvisor (Ex-1015), 1 (describing deep-links for travel sites that repurpose user-entered reservation data); Becker (Ex-1007), 2, 4 (describing search-transfer landing pages that re-execute the user's query ("sometimes with slight modification") at an advertiser's site); Fox, ¶51.

Moreover, it was known in the art that the reuse of a travel location parameter was particularly effective for increasing traffic conversion (sales), customer engagement, and customer experience. Fox, ¶52; TripAdvisor, 1 (announcing that "contextual-linking technology" providing travel location specific "deep-link[s]" "correlate[d] to search terms ... used by travelers" led to "response rates of over 8% for online travel advertising (a 32X improvement over the usual .25% response rate of most online advertisements).") Additionally, as the '342 patent admits, POSAs were aware that to make these deep-linked promotions work, the parameters entered by the user from the first site needed to be input into the appropriate variables on the second site. Ex-1001, 3:3-10. Accordingly, a POSA would have found it obvious to employ such deep-links in the "promotions" section of, for example, Belanger's Figure 53 to ensure that car rental customers would enjoy a similarly streamlined booking experience across other travel

websites (e.g., hotels, airlines, amusement parks, etc.) as an additional value in booking their car rental with a particular company (e.g., Enterprise Rent-a-Car). Belanger, [0168], [0179], [0206]; TripAdvisor, 1; Fox, ¶53.

In addition to providing value for customers, a POSA would have recognized that travel booking sites would benefit from the increased traffic of committed travelers who have booked a car rental, for linking to their hotel, airline, or attraction site. TripAdvisor, 1; Fox, ¶54. Likewise, Belanger's reservation website would also benefit from similar traffic and seek to facilitate business relationships with other travel sites for preferential ranking or placement of its own deep links on such booking pages. TripAdvisor, 1; Fox, ¶54. Indeed, Belanger teaches advertising its own deep-links on "other websites" (Belanger, [0168]) and a POSA would have recognized that the reciprocal exchange of traveler booking information would benefit all travel partner sites. TripAdvisor, 1, Ex-1001, 2:49-3:2; Fox, ¶54.

Regarding placement of promotional links 318, while Belanger's Figure 53 shows a single promotion link 318 after the bolded label "Promotions," POSAs would have recognized that the label "Promotions" may also refer to multiple promotion deep-links or a link to a page with multiple promotion links (e.g., as in Fig. 37, left column), and that this confirmation page would beneficially display multiple promotion links in a desired order or placement. Fox, ¶55; Belanger, Fig.

37 (showing promotions 318 including a link to “All Promotions”). POSAs thus would have been motivated to also implement the confirmation page in Figure 53 to include multiple promotion deep-links to different categories of travel services to provide more cross-promotional opportunities once a booking is confirmed. Fox, ¶55; TripAdvisor, 1.

1. Belanger’s Creation and Serving of Deep-Links

To serve the promotional deep-links, Belanger describes an exemplary “[a]utomated reservation transaction processor 150” including one or more “application servers” that interact with the “customer/promotional database.” Belanger, [0106], Fig. 6; Fox, ¶56. These application servers are in turn hosted on “web servers” to which customers’ browsers connect to access the reservation website. Belanger, [0106]; Fox, ¶56.

The application servers run a “servlet” that “performs the major decision-making tasks for the reservation booking process,” including generating a webpage by “interact[ing] with the databases to gather necessary information” to “determine[e] which page should be constructed” and “the particular data that should appear on the page.” Belanger, [0107-09]; Fox, ¶57. POSAs thus would have understood that with the promotional deep-links being part of the “data that should appear on the page,” the servlet generates the advertised promotions and

their deep-links for a webpage using the promotional settings in the promotional database. Belanger, [0019], [0107-08], [0168]; Fox, ¶57.

The provided promotional conditions (i.e., parameter values) used in the promotional deep-links on Belanger’s reservation website are associated with the field parameter names included in the deep-link manually (i.e., via human interaction). This manual process is facilitated through a user interface (*see, e.g.*, GUI 700 in Fig. 73) where an “administrator” (e.g., an employee of the reservation site or business partner (Belanger, [0212])) controls “a plurality of settings for the ... rules of a promotion.” *Id.*, [0081], [0209]. The promotion settings are stored in a “customer/promotional” database (*e.g.*, database 208 in Fig. 2) “for subsequent retrieval and processing” (*id.*, [0209]) and used to control the display of promotional deep-links shown on the reservation webpage (e.g., via marketing tile 317 where promotions 318 are displayed). *Id.*, [0128], [0209-10]. Belanger also describes data exchanges with other servers (e.g., business partners) implemented as web services to exchange data used for promotional linking operations, such as through well-known “Simple Object Access Protocol (SOAP)” and XML formatted messaging. *Id.*, [0024], [0110-13]; Fox, ¶58; Tidwell (Ex-1018), 10-12, 19 (describing web services using SOAP).

As can be appreciated, POSAs would have understood that manually creating the promotion settings for deep-links is time and resource intensive and

multiplies along with the scale of desired promotions, and not only impacts Belanger's offered promotions, but also the ability to implement outgoing links to travel partners. Fox, ¶59; *see* Ex-1001, 3:22-25. That is, this manual bottleneck would have likewise hampered Belanger's website operators' abilities to expand outgoing promotions to follow the industry trend towards reusing user-entered data across partner websites in the travel industry, as noted *supra* § VI.A. *See* Fox, ¶59; TripAdvisor, 1. Accordingly, POSAs would have looked to solutions for automating the programmatic association of web form parameters between websites to generate more and better deep-linked promotions that repurpose user-entered data, as discussed *infra* VI.C. Fox, ¶59.

B. Halevy⁴

Halevy teaches one such system for automatically generating deep-links, by querying a "form database." Halevy, [0009]. Halevy's form engine and database, like the '342 patent's universal variable "mapper," automatically maps fields on forms from different websites to a shared set of identifiers (i.e., "properties") determined from the form data, thereby mapping corresponding fields across various website forms to each other. *Id.*, [0037], Fig. 2; Fox, ¶61.

⁴ Halevy issued as U.S. Patent No. 8,037,068 and is assigned to Petitioner, Google LLC.

1. Halevy Automatically Maps Data Fields of Website Forms for Creating Deep-links

To create the form database, Halevy teaches a web crawling process (*see* Fig. 1 below) executed by a form engine 106 that “accesses a number of forms 121-123 which are hosted on servers located throughout network 110.” Halevy, [0035]. These forms are “obtained” from “a number of sources” and provide “access to data within underlying databases 131-133.” *Id.*, [0035], [0037], Fig. 1.

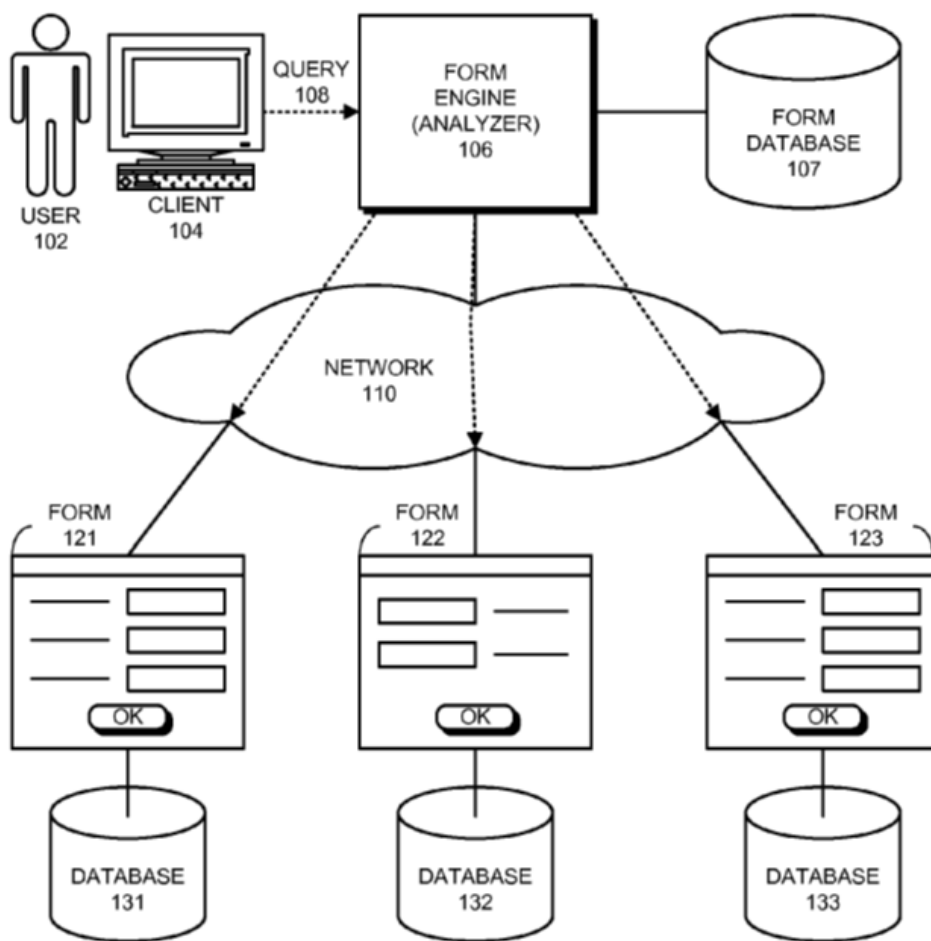


FIG. 1

Once forms are identified, as discussed in more detail *infra* § VI.D, Halevy’s crawling process “creates database entries for the identified forms” by “obtaining and storing metadata describing the identified forms into database entries” in form database 107. *Id.*, [0009], [0015]. The stored “metadata” includes data describing the “input fields” on the form, such as the field’s “parameter name” or field identifier. *Id.*, [0016], [0037], [0044], Fig. 2; Fox, ¶63.

2. Universal Variable Map (Form Database)

For each form entered in the form database, Halevy’s process computes a “class” for the website the form is found on (e.g., a “university” or “newspaper” website) and a “class” for the form itself. The form’s “class” indicates “the types of objects that can be obtained by querying the form,” such as “books, hotels, restaurants” and “houses, jobs, and cars for sale.” Halevy, [0039-44], [0057]; Fox, ¶64. Halevy refers to the “classes” as “SO-classes,” or “search ontology” classes. Halevy, [0038], Fig. 6; Fox, ¶64. The classes in turn have “SO-properties” or “search ontology” properties; these properties, which are also referred to as “output properties,” represent “the type of information [that is] obtain[ed]” by using the form. Halevy, [0038], [0048]; Fox, ¶64.

For example, if a form’s “SO-class” is identified as “car,” then the fields in the form are mapped to the “SO-properties” (i.e., normalized variables) for the “car” class such as the properties for “make” and “model” of the car. Halevy,

[0095], [0098-101]; Fox, ¶65. Likewise, POSAs would have understood that if the form's "SO-class" is identified as "hotel," then the fields in the form may be mapped to the "SO-properties" for the "hotel" class such as "arrival date," "departure date," and "location." Fox, ¶65. POSAs also would have understood that other classes of forms such as "amusement park," "car rental," "airline," etc., or more generally, "reservations" would similarly map relevant fields to that class's SO-properties. *Id.* Halevy also computes a "schema mapping," which is a "mapping between the fields in the form and the SO-properties." Halevy, [0049], [0062-67]; Fox, ¶65.

As shown in Figure 3 (annotated below), for each form, Halevy's form database stores the identified class of the form, data for each field (including the parameter name), and the mapping between the fields on the form and the corresponding properties of the form's class. Halevy, [0044-49] ("For each form in the source [i.e., website] the following information can be captured ... [f]ields in the form ... include[ing] its parameter name ... [t]he class of the form ... [s]chema mapping"), [0067], [0125]; Fox, ¶66.

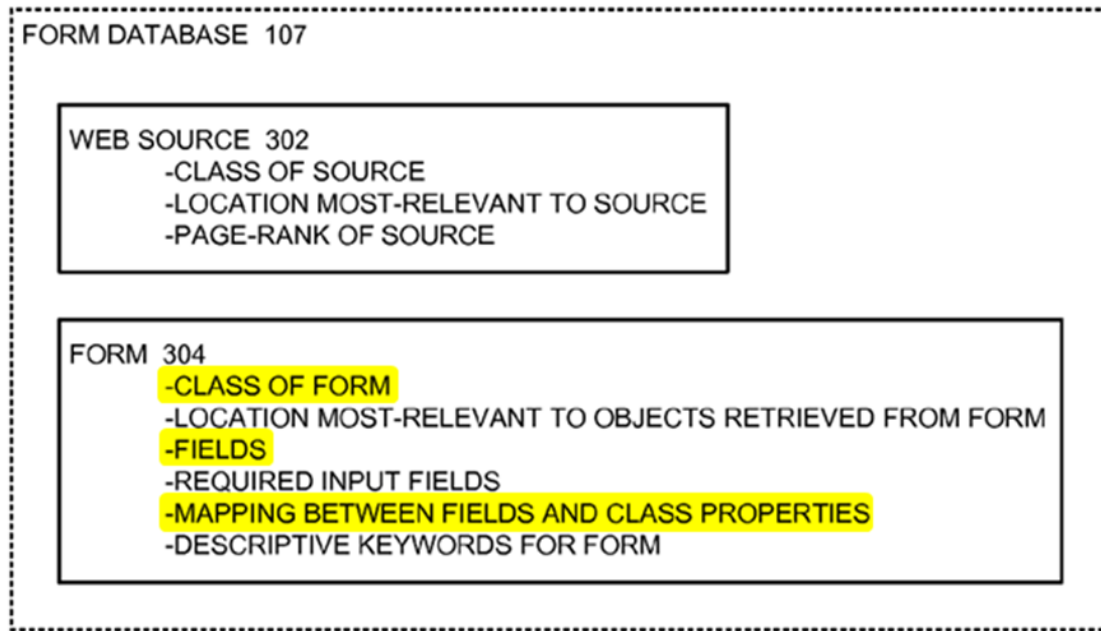


FIG. 3

The class and properties identified for the various forms are then used to “facilitate subsequent searches through content which is accessible through” the forms. Halevy, [0009], [0136-38], Fig. 2. This search process, explained in detail directly below, accepts a query string and via the form database’s hierarchy of classes and their properties, returns a list of deep-links to various identified forms in the database with the keywords of the query assigned to fields in the deep-link based on mapping the keywords and the form’s fields to the same SO-properties. *Id.*, [0018], [0136-38], [0125] (describing mapping query keywords to form fields using the SO-properties); Fox, ¶67. This use of the form database shows that Halevy maps corresponding forms fields from different websites to the same SO-property to provide search capabilities on multiple forms that return similar data,

despite the fields on each form having different parameter names. Fox, ¶68; *see also* Halevy, [0062-66] (describing mapping fields to SO-properties), [0125] (describing identifying forms by mapping query keywords to fields based on mapping the keywords to the same SO-properties).

3. Retrieving Deep-Links from the Form Database

Halevy teaches querying its form database to return a list of relevant “query-through links.” Halevy, [0136-38]; Fox, ¶69. Each of the query-through links automatically directs the user to either a results webpage or a partially filled form at the website identified in the link (Halevy, [0138], [0143]), with the keywords in the query associated with corresponding parameter names of the form’s fields. *Id.*, [0123-28] (describing generating deep-links with query keywords associated with corresponding field parameters based on the SO-properties determined for the keywords and fields); Fox, ¶69.

This query process is diagrammed in Halevy’s Fig. 6 (below) where the system “receives a query containing keywords (step 602)” and “predicts a search-ontology class for the query based on the keywords (step 604).” Halevy, [0136]. The system then “constructs a structured query” by “mapping keywords to [SO-] properties” of the predicted class (step 606). *Id.* For example, Halevy describes converting an example user query “Telephone David Ko 98102” to a structured query in the “Person” SO-class, with the “FirstName” SO-property assigned the

value “David,” the “LastName” SO-property assigned the value “Ko,” and the “ZipCode” SO-property assigned the value “98102.” *Id.*, [0112]; Fox, ¶70. Using this structured query, Halevy’s system “performs a lookup” in the form database “to identify forms of interest (step 608),” which are “rank[ed]” and then “present[ed] the user” as “query-through links” (step 612). Halevy, [0114], [0137-38]; Fox, ¶71. Each query-through link, when clicked by the user, “submit[s] a query to the associated form of interest (step 614)” by prepopulating the form’s fields with their associated keywords, as determined by the SO-property mapping. Halevy, [0136-38]; Fox, ¶71.

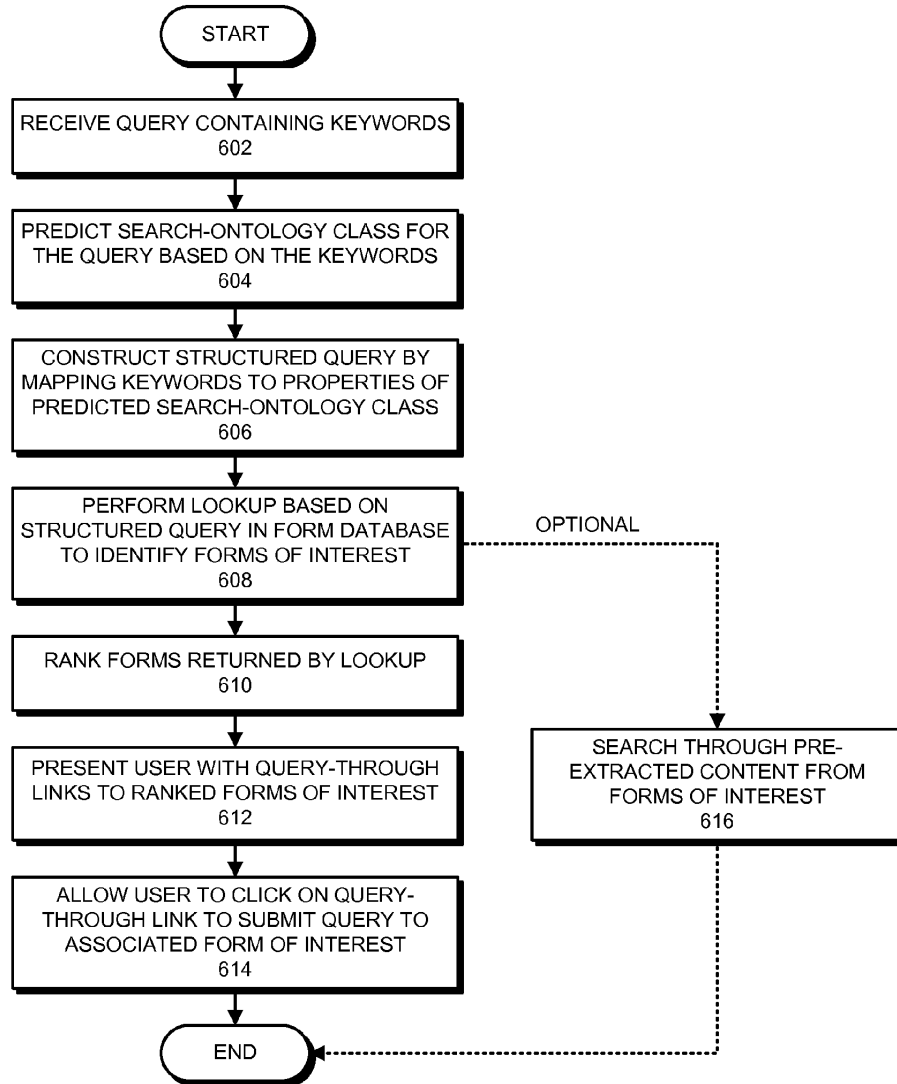


FIG. 6

Because Halevy’s query-through links (each to a “form of interest”) include an association of query keywords to corresponding field parameter names to automatically generate search results from the linked form using the keywords as values in the form’s fields, POSAs would have recognized the query-through links to be deep-links that repurpose the query keyword values. Halevy, [0123-28], [0136-38], [0143]; Fox, ¶72. Furthermore, because Halevy discloses that a single

structured query—with keywords of the query mapped to SO-properties—returns *multiple* query-through links—with keywords mapped to corresponding fields based on the SO-property mapping—, POSAs would have understood Halevy’s system to map corresponding form fields from different websites to the same SO-properties (i.e., obviously, because otherwise returning multiple deep-link results from a single query would not be possible). Fox, ¶72.

Since Halevy describes its search system functionality as an enhancement to conventional network-based search engines (Halevy, [0007-08]) and the system uses “any type” of “communication channel capable of coupling together computing nodes” (*id.*, [0034]), POSAs would have found it obvious for Halevy’s system to use SOAP (which is used by Belanger) as a communication channel to receive queries and return resultant deep-links. Fox, ¶73; Belanger, [0110], Tidwell, 10-12, 19. Likewise, POSAs would have found it obvious for Halevy’s system to use XML (which is used by Belanger) for data representations (e.g., of queries and resultant deep-links) that are communicated using SOAP. Fox, ¶73; Belanger, [0024], [0110-0120]; Tidwell, 2, 6, 11, 12, 15-38.

C. POSA’s Motivations to Modify Belanger in view of Halevy

As discussed *supra* § VI.A.1, Belanger teaches a reservation website implementation where an “application server” determines the content that is shown on the website, where this content includes promotional deep-links selected for

display by the servlet when generating a promotion for a webpage. Belanger, [0019], [0106-09], Fig. 53 (element 318); Fox, ¶74. Additionally, as discussed *supra* § VI.A, a POSA would have been motivated to implement Belanger’s reservation website so that the promotional deep-links displayed during or at the close of booking (e.g., Fig. 53) advertise related travel websites that are known to use similar user-entered parameters, and would benefit from repurposing these parameters. Fox, ¶74; TripAdvisor, 1. Belanger’s promotions rely on *manual* creation of deep-links (Belanger, [0207], [0209]), as noted *supra* § VI.A.1. Fox, ¶75. Rather than requiring a person (e.g., administrator or vendor employee) to manually enter promotional conditions for each vendor website deep-link promoted on Belanger’s website (*see* Belanger, [0209-12]), POSAs would have been motivated to look to solutions that generate deep-links with promotional conditions automatically, such as reusing parameters already entered by the user during the booking process to leverage at other travel websites. Fox; ¶75; TripAdvisor, 1. Halevy teaches one such system for automatically generating deep-links to other websites with the keywords from a query repurposed as parameter values in the deep-link. Fox, ¶75. A POSA would have been motivated to modify Belanger with Halevy’s teachings as detailed below.

Halevy’s system includes a “form engine” that “exist[s] on any computing device which is accessible through [a] network” using “any type of ...

communication channel capable of coupling together computing nodes.” Halevy, [0034]; Fox, ¶76. Belanger discloses using the well-known “Simple Object Access Protocol (SOAP)” to exchange data between computers (Belanger, [0110]), which a POSA would have found obvious to utilize as a “communication channel” to Halevy’s system. Halevy, [0007-08]; Fox, ¶76; Tidwell, 19 (describing using SOAP to submit a query to a search engine).

To automatically generate deep-links for promotions that repurpose user-entered travel data, as discussed above, a POSA would have been motivated to modify Belanger’s application server to retrieve deep-links for desired promotions automatically. Fox, ¶77. A POSA would have found it obvious to use Halevy’s system to automatically generate these deep-links because Halevy’s system returns one or more deep-links relevant to a query’s keywords, with some of the keywords mapped to corresponding parameter names in the deep-link. Fox, ¶77. A POSA would have recognized that such a web service data exchange is taught by Belanger for similar purposes (Belanger, [0110]), and leveraging this existing communication feature would have automated the sending of queries and retrieval of deep-links from Halevy’s system for presentation as related travel promotions (as already provided by Belanger, *e.g.*, at the close of booking via Figure 53) by submitting queries to Halevy’s “form engine” via SOAP-based communications. Fox, ¶77.

As discussed *supra* §VI.B.3, Halevy teaches allowing end-users to query the form database to retrieve deep-links. Halevy's system generates deep-links by determining keywords from the query to identify a "search class" for the query, "mapping the individual words in the query to properties of the chosen search class," and performing a look-up in the database with the class and property mapping. Halevy, [0089-92], [0112], [0137-38]; Fox, ¶78. Halevy describes using various types of "prediction" to perform its identification and mapping process. Halevy, [0090-111]; Fox, ¶78.

In combination with Belanger, where Belanger's application server generates deep-links for promotions by sending queries to Halevy's form engine, a POSA would have understood that the queries would have been constructed to yield a desired deep-link. Fox, ¶79. For example, should Belanger's application server display a promotion for hotels or even a specific hotel partner, POSAs would have found it obvious for the application server to construct a query for Halevy's system specifying the keyword "hotel" or the name of the branding partner, along with the user-entered values for date and location, to generate a list of deep-links—to hotel websites or to the specified brand website—that display search results using the user-entered date and location as parameter values. Likewise, a POSA would have found it obvious for the application server to construct a query using just the user-entered date and location values of the car

rental reservation of Belanger for a list of deep-links to any type of website that also uses date and location values to generate a results page. Fox, ¶79. Such an implementation would have been within a POSA's skill, as it would have involved basic programming techniques to communicate values entered by the user in the fields on Belanger's reservation website along with any pre-defined keywords (e.g., such as "hotel"), and to construct a query string so that Halevy's system would return one or more deep-links relevant to the query string. Fox, ¶79.

A POSA would have recognized that implementing Belanger's and Halevy's teachings discussed above, where Belanger's application server sends a query string to Halevy's system to retrieve deep-links, was an obvious way to leverage Halevy's already existing form database via Belanger's existing data exchange mechanism (i.e., SOAP) for using customer-entered parameters to automatically generate deep-links. Fox, ¶80; Tidwell, 10-12, 19 (describing using SOAP for submitting queries to a search engine). For example, when generating Belanger's Figure 53 UI (or another webpage where promotions are displayed) in view of Halevy, POSAs would have used Belanger's existing SOAP/XML messaging to send Halevy's system query strings, using relevant data entered by the customer such as date and location along with any other optional keywords, and to search Halevy's form database for one or more relevant deep-links for the promotions displayed on Belanger's webpages. Fox, ¶80.

To retrieve deep-links for presentation, a POSA would have modified Belanger's application server, which already runs the "servlet" responsible for selecting promotions for display, as discussed *supra* § VI.A.1, to query Halevy's system (using SOAP) with the user-entered data and any other keywords during the booking process to repurpose the user-entered data on Belanger's reservation website in the deep-links. Such would enable a more robust and automatic presentation of promotion deep-links (e.g., Fig. 53) as was known to be desirable in the travel industry to yield better results. Fox, ¶81, TripAdvisor, 1.

For example, Belanger's Figure 53 UI is a "confirmation" page. Belanger, [0063]. Thus, a POSA would have understood that at the point in the reservation process where Figure 53 (below) is displayed, the user has already entered data relevant to the reservation on previous pages, e.g., on the "home" page shown in Figure 37. Belanger, [0056], [0122], Fox, ¶82.

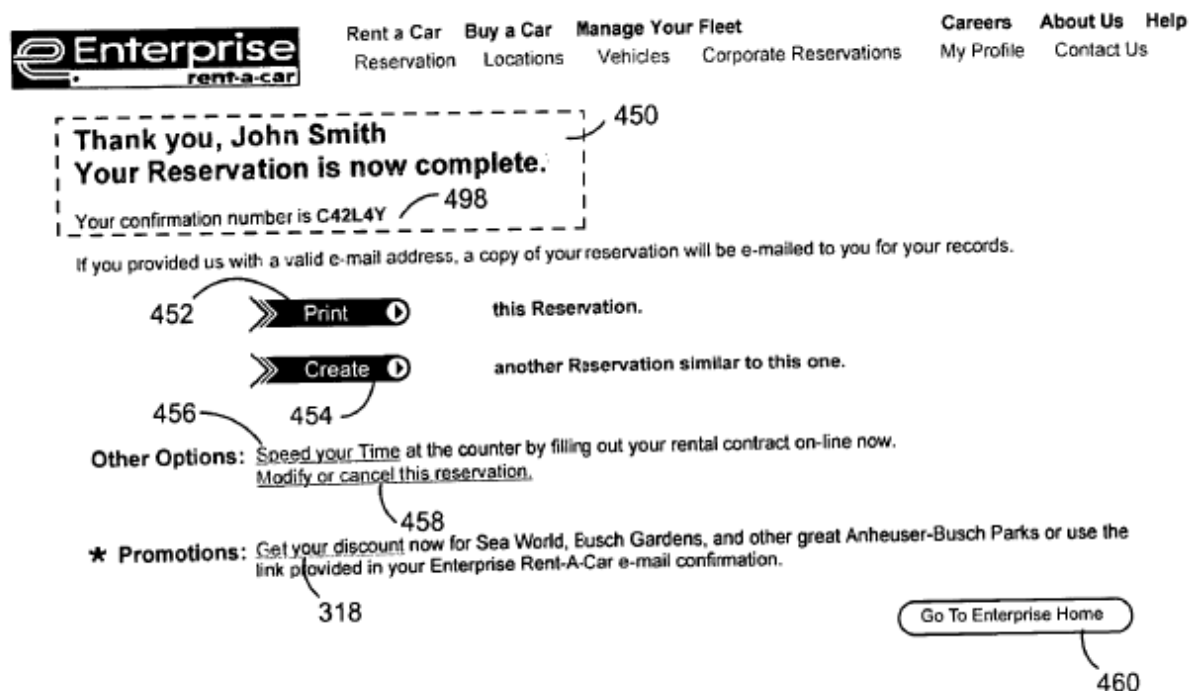


Fig. 53

Figure 53 in Belanger teaches displaying promotions (promotional link 318) upon reaching confirmation. As discussed *supra* § VI.A, a POSA would have been motivated to implement Figure 53 so that this promotional link is a deep-link to a website offering reservations for travel-related products/services such as hotels, restaurants, attractions, or airlines, where the deep-link includes parameter values corresponding to values entered by the user during the reservation process. Fox, ¶83; TripAdvisor, 1. Further, as discussed in this section above, a POSA would have leveraged Halevy's teachings to automate the process of serving deep-links to avoid the well-recognized limitations and drawbacks of manual coding. A POSA

would have recognized that the presentation of the promotions on, *e.g.*, Fig. 53, is an interface design consideration that may vary based on the website's presentation style and the desired promotion to be displayed, and that the query submitted to Halevy's system via SOAP/XML would be adjusted accordingly to return a desired deep-link. Fox, ¶83.

For example, Fig. 53 may include a list of different types of promotions, in which case the query for each promotion would be restricted to only return deep-links to specific products or services. Fox, ¶84. In this case, Belanger's application server would submit to Halevy's system a query string specific to each promotion, as discussed above, that returns one or more deep-links to forms in specific "class[es]" of interest (*e.g.*, airline booking, hotel booking, restaurant booking, etc.), where the date and location values submitted by the user on Belanger's reservation website are populated as the date and location values in each deep-link. Halevy, [0123-28], [0136-38]; Fox, ¶84. Furthermore, Halevy teaches "ranking" search results based on relevance of the pages that the links point to and "proximity to the location mentioned in the query and associated with the web source"; thus, POSAs would have been motivated to implement Belanger's webpage to only show a certain number of the top-ranked promotions using the deep-links, or would have selected a single deep-link to promote, to avoid visually cluttering the page. Halevy, [0145]; Fox, ¶85.

As another example, Fig. 53 could be implemented to show separate links to promotions for different categories, *e.g.*, airline booking, hotel booking, and restaurant booking. In this case, each link would take the user to lists of promotions in that category. Each list would be generated by submitting a relevant query string to Halevy’s system, and then ranking and limiting the resultant deep-links in the same manner discussed above, except that separate queries would be submitted for each category. Fox, ¶86.

D. Claim-by-Claim Analysis

In the discussion below, “Belanger/Halevy” refer to the combination of Belanger in view of Halevy described *supra* § VI.C, and “Belanger” or “Halevy” refer to the references’ respective disclosures. Fox, ¶87.

1. Claim 1

a. [1PRE] “A method, comprising:”

Belanger teaches a “*method* of processing a reservation transaction.” Belanger, [0014]. As discussed *supra* § VI.C, an example of performing Belanger’s method—when Belanger is implemented to query Halevy’s form database—includes displaying promotions on Belanger’s webpages with deep-links that reuse the values entered by the user on Belanger’s website (*e.g.*, via the webpage in Fig. 37). As detailed *infra* §§ VI.D.1.b-VI.D.1.i, this method *comprises* the remaining limitations of claim 1. Fox, ¶89.

- b. **[1A] “obtaining or inferring computer-readable field identifiers from sites or applications on the Internet, the obtaining performed by a spidering module running on a computer having at least one processor and non-transitory computer memory;”**

Belanger/Halevy, as discussed *supra* § VI.C, discloses this limitation via Halevy’s form crawling process discussed *supra* § VI.B.1. Specifically, Halevy teaches crawling “network 110” (e.g., *the Internet*) via a “form engine” that “crawl[s] through web sites to identify web-based forms.” Halevy, [0034-35]. The form engine can “exist on any computing device which is accessible through network 110,” and the computing device “stor[es] instructions ... executed by a computer.” *Id.*, [0034], claim 9. POSAs would have thus recognized the form engine to be “*a spidering module running on a computer having at least one processor and non-transitory computer memory.*” Fox, ¶90.

As Halevy’s Figure 1 (provided below) shows, the crawling process “*obtain[s]*” form data from “*sites or applications on the Internet*” because it “accesses a number of forms 121-123 which are hosted on servers located throughout network 110 ... to access to data within underlying databases 131-133.” Halevy, [0035], Fig. 1. During this crawling process, Halevy’s “form engine 106 first obtains web-based forms of interest, and then creates corresponding database entries for the identified forms in an associated form database 107.” *Id.*, [0035], [0037], Fig. 1.

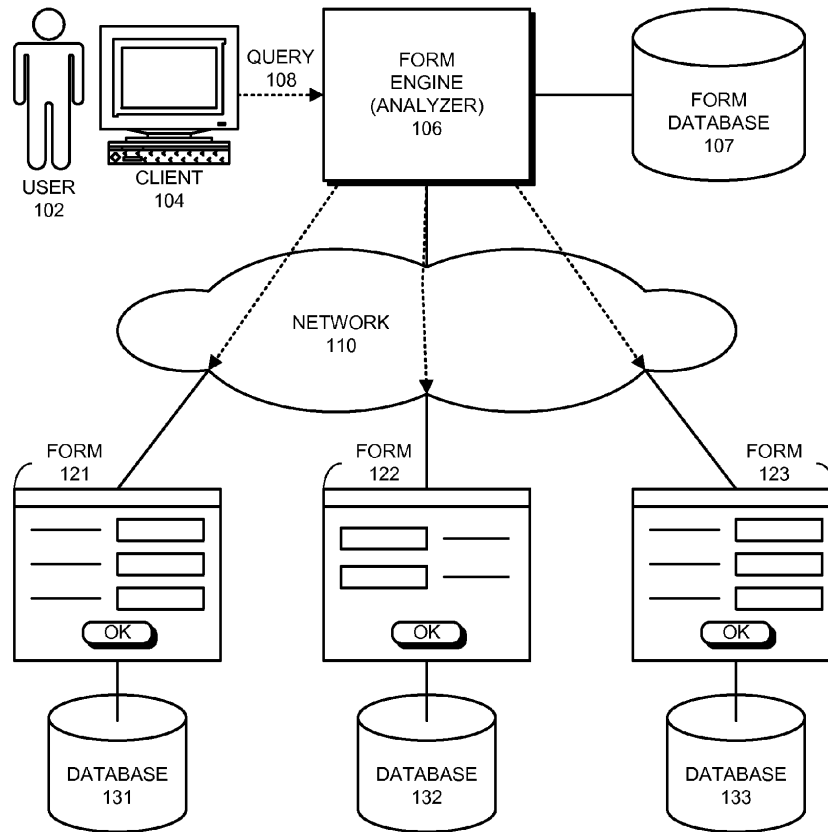


FIG. 1

Halevy’s crawling process includes “*obtaining ... computer-readable field identifiers*” from the various forms because for each form, the process captures “each field” including “its *parameter name (used to invoke some server side-script)*, the human-readable or visible text for that field, field types (text, select box, radio button, or checkbox), and exemplary values for the field (e.g., the options in a select box).” *Id.*, [0043-44] (emphasis added).

Halevy discloses that forms are written in “HTML,” which a POSA would have understood was a standard language for writing webpages. *Id.*, [0142]; Fox, ¶93; Belanger, [0012]; Jha (Ex-1005), [0063], [0098]. POSAs would have

understood that in HTML, the “parameter names” for fields are the same as what the ’342 patent describes as the “names of programming variables and identifiers.” (Ex-1001, 13:32-36); Fox, ¶93. Thus, the “parameter name” for each field is a “*computer-readable field identifier*.” Fox, ¶93.

- c. **[1B] “mapping the computer-readable field identifiers to one another using normalized variables stored in a normalized variable data database, the normalized variable data database representing a universal variable map of searchable classes and attributes of products or services on the Internet, the mapping linking computer-readable field identifiers of the sites or applications on the Internet;”**
 - i. **“mapping the computer-readable field identifiers to one another using normalized variables stored in a normalized variable data database, ...”**

As discussed *supra* § VI.C, in Belanger/Halevy, Halevy’s “form database” is accessed via its “form engine” by Belanger’s “application server” via a web service protocol (SOAP/XML) during the booking process. Belanger, [0110]; Fox, ¶94. As discussed *supra* §§ VI.B.1 and VI.D.1.b, Halevy’s form engine executes a crawling process that categorizes webpages and forms into a “schema” (i.e., a model/structure) consisting of a “search ontology” (SO) of “classes and properties.” Halevy, [0038]; *see* Uschold (Ex-1013), p.1 (defining and explaining “ontology” and “database schema”). The “SO-classes” are identified for each form and used to map the fields in the form (i.e., “*computer-readable field identifiers*”)

to one another using “SO-properties,” as shown below in annotated Figure 3.

Halevy, [0039-40], [0045], [0049], [0062], [0067]; Fox, ¶94.

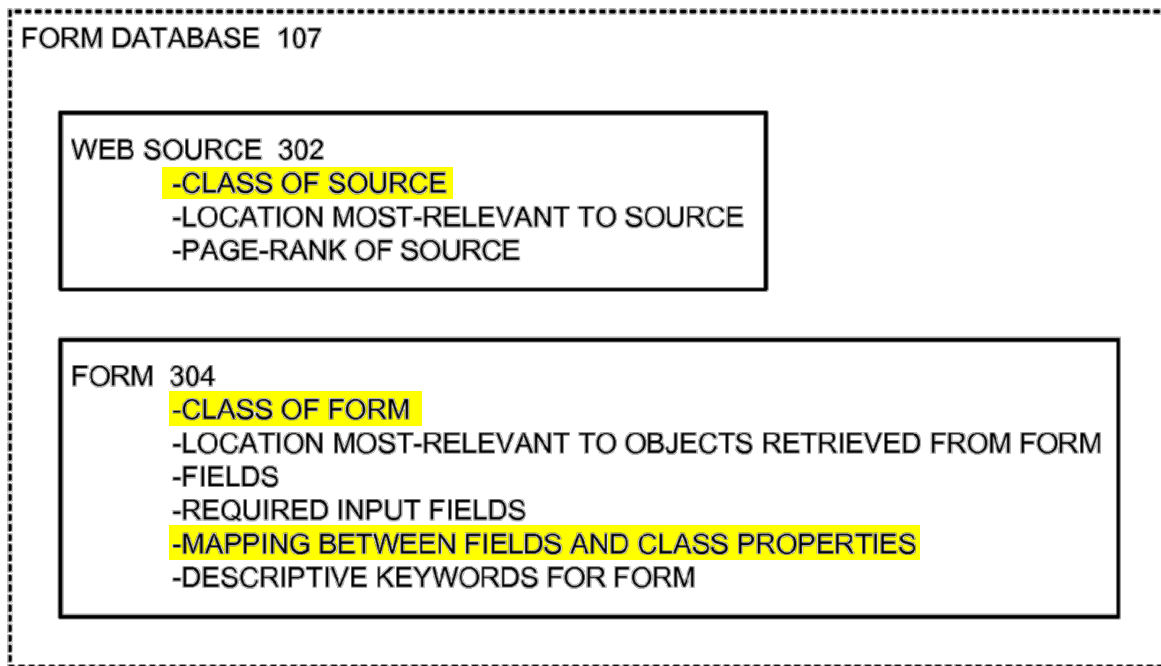


FIG. 3

The SO-properties are associated with SO-classes and are derived from the “output properties,” described as “the type of information ... obtain[ed]” from querying a form. Halevy, [0048], [0062], [0067]. As discussed *supra* § VI.B.3, multiple different forms on different websites can map to the same SO-classes, and likewise, fields from different forms can map to the same SO-properties. Halevy, [0035], [0037], [0067], [0095], Fig. 2; Fox, ¶95. For example, for forms that identify in the SO-class “car,” the fields (i.e., “*computer-readable field identifiers*”) are *mapped* to their corresponding SO-properties (i.e., “*normalized variables*”), such as “make” and “model” associated with the “car” SO-class.

Halevy, [0095], [0098-101]; Fox, ¶95. Since Halevy’s crawling process maps the fields of various web forms to a set of SO-properties associated with the form’s identified SO-class, POSAs would have recognized the SO-properties to be “*normalized variables*” that are used to “*map[] the computer-readable field identifiers to one another.*” Fox, ¶96. Further, since the mapping of fields to SO-properties is stored in the form database (as shown in Figure 3), POSAs would have recognized that the SO-properties (i.e., “*normalized variables*”) are “*stored in*” the form database, thus making the form database “*a normalized variable data database.*” Fox, ¶96.

- ii. **“... the normalized variable data database representing a universal variable map of searchable classes and attributes of products or services on the Internet, the mapping linking computer-readable field identifiers of the sites or applications on the Internet;”**

As discussed *supra* § VI.D.1.d.i, POSAs would have recognized Halevy’s form database to be a “*normalized variable data database*” that stores a mapping of SO -properties (i.e., “*normalized variables*”) associated with SO-classes to fields (i.e., “*computer-readable field identifiers*”) found on various website forms (i.e., “*sites or applications on the Internet*”) analyzed by the crawling process. Halevy, [0043-49], [0067], Fox, ¶97.

Halevy’s disclosure of the form database in which corresponding fields from different forms on different websites map to the same SO-properties based on the

SO-class identified for the form would have been recognized by a POSA as what the '342 patent refers to as a *universal variable map*. See Ex-1001, 17:48-52 (“Variations of linking data variables and fields may include a source form, a destination form, **a single source to many destination form**, many sources to one destination form” (emphasis added)); Fox, ¶98. Thus, Halevy’s form database is a “*universal variable map*” that “*link[s] computer-readable field identifiers of sites or applications on the Internet*” because form fields (including parameter names) from multiple different websites are linked to each other in the form database by being mapped to the same SO-properties associated with an SO-class. *Id.*

Belanger/Halevy further teaches that the form database “*represent[s] a universal variable map of searchable classes and attributes of products or services on the Internet*” because as noted *supra* §§ VI.B.3 and VI.C, Belanger’s “application server” (Belanger, [0106-07]) reads data entered by the user in a form on Belanger’s website and uses that data to query Halevy’s form database to retrieve a list of deep-links, each to a “form of interest.” Halevy, Abstract, [0137-38]; Fox, ¶99. As discussed *supra* § VI.C, the query submitted to Halevy’s form database (including user-entered data from Belanger’s website as query keywords) *maps* the keywords to “properties of the chosen search[-ontology] class” identified for the query. Halevy, [0089], [0136]; Fox, ¶99. POSAs would have thus recognized that Halevy’s SO-classes are “*classes ... of products or services on the*

Internet” because they are determined for each form based on the “types of objects that can be obtained by querying the form” (e.g., books, hotels, restaurants, houses, jobs, cars, etc.). Fox, ¶100; Halevy, [0045]. POSAs also would have recognized that Halevy’s SO-properties are “*attributes of products or services on the Internet*” because they “specify the type of information” obtained from querying the form such as SO-properties of “make” and “model” for the SO-class of “cars.” Fox, ¶100; Halevy, [0095], [0098-101].

POSAs would have recognized that this same process is applicable to “applications” because it was known in the art (as acknowledged by the ’342 patent) that “fields or variables on the publisher site or mobile app” would have been obtained and processed in the same manner using known communication protocols like SOAP. Fox, ¶101; Ex-1001, 1:53-58, 3:3-3:21; *see also* Tidwell, 10-12 (describing using SOAP to provide web services to web-based applications including from “an Application Service Provider’s vast server farm to a PDA”).

- d. **[1C] “automatically programmatically generating a deep-linked ad utilizing the universal variable map, the deep-linked ad linking at least one of a set of computer-readable field identifiers used by a first site or application on the Internet with at least one of a set of computer-readable field identifiers used by a second site or application on the Internet, wherein the first site and the second site are in same vertical or**

different verticals for which the spidering module is configured to search on the Internet;”

- i. “automatically programmatically generating a deep-linked ad utilizing the universal variable map, ...”**

As discussed *supra* § VI.C, promotions displayed on Belanger’s webpages (e.g., in Fig. 53 or another page) are *deep-linked ads*, and Belanger/Halevy discloses “*automatically programmatically generating*” these ads, as explained below. Fox, ¶102.

In Belanger/Halevy, Belanger’s “application server” determines the content displayed on its webpages, including displaying promotions having deep-links *generated* by querying Halevy’s system with values in the query mapped to corresponding parameter names in each deep-link. *Supra* § VI.C; Fox, ¶103.

POSAs would have thus recognized Belanger’s promotions with Halevy-generated deep-links to be *deep-linked ads*. Fox, ¶103. POSAs would have understood that these ads are *automatically programmatically generated* by Belanger’s application server because the “servlet” running on the application server “interacts with the databases to gather necessary information for ... the particular data that should appear on the page” (Belanger, [0108]), which in combination with Halevy, would have included querying Halevy’s system for deep-links. Fox, ¶103. Thus, Belanger/Halevy discloses “*automatically programmatically generating a deep-linked ad.*”

As also discussed *supra* § VI.C, Belanger’s application server queries Halevy’s form engine via a web service protocol (SOAP/XML) during the booking process. Belanger, [0024], [0110]; Fox, ¶104. Halevy’s form engine then accesses its form database. Halevy, [0034]. As discussed *supra* §§ VI.B.2 and VI.D.1.c, the form database meets the claimed “*universal variable map*,” and as discussed below, Belanger/Halevy generates the deep-linked ads *utilizing* the “form database.” Fox, ¶104. Specifically, in Belanger/Halevy, Belanger’s “application server” *utilizes* the form database because Belanger’s “servlet” on the “application server” (Belanger, [0106-07]) communicates reservation data (e.g., date and location) as keywords in a query using SOAP/XML to Halevy’s form database to perform a “lookup” to retrieve one or more deep-links, each pointing to a “form of interest” (Halevy, Abstract, [0137-38]). *Supra* § VI.C; Fox, ¶104. Thus, POSAs would have recognized that Belanger’s server in the combination *utilizes* Halevy’s “form database” in *generating* deep-linked ads. Fox, ¶104.

- ii. **“... the deep-linked ad linking at least one of a set of computer-readable field identifiers used by a first site or application on the Internet with at least one of a set of computer-readable field identifiers used by a second site or application on the Internet, wherein the first site and the second site are in same vertical or different verticals for which the spidering module is configured to search on the Internet;”**

As discussed *supra* §§ VI.B.1 and VI.D.1.b, Halevy’s crawling process for creating the form database “obtains web-based forms to be searched ... from a number of sources,” including “crawl[ing] through web sites to identify web-based forms, ... receiv[ing] manually provided web-based forms, ... find[ing] web-based forms through methods other than crawling.” Halevy, [0009]. POSAs would have recognized that Belanger’s own website (*i.e.*, “*a first site or application*”) would have been included in Halevy’s form database because Belanger’s website has a number of forms, where a POSA would have understood that in the software running Belanger’s website each form is associated with “*a set of computer-readable field identifiers used by a first site*”; thus, the website is one of “a number of sources” used to “identify web-based forms.” Halevy, [0035]; Belanger, [0122], Fig. 37; Fox, ¶105. Likewise, to the extent Belanger’s form webpage (e.g., Fig. 37) was not previously crawled by Halevy’s system, a POSA would have found it obvious (prior to utilizing Halevy as a web service) to provide Belanger’s form webpages to Halevy’s system (*see* Halevy, [0035]) to provide Belanger’s reservation website with the ability to submit queries for deep-links for its own promotions. Fox, ¶106.

An advertised promotion generated using Belanger/Halevy for presentation during or at the end of the booking process contains a deep-link to another vendor’s website (e.g., an amusement park, hotel, or airline website) and as such,

maps fields from Belanger’s reservation web form (e.g., Fig. 37) to corresponding fields in a form on a vendor’s website. Fox, ¶107. For instance, in the exemplary use of Belanger/Halevy discussed *supra* § VI.C, the promotion served on Belanger’s webpages (e.g., promotional deep-link 318 in Fig. 53), is a *deep-linked ad* that, as discussed *supra* § VI.D.1.d.i, includes a deep-link retrieved by querying the form database with keywords including user-entered data (e.g., date and location values from the fields on Belanger’s reservation web form), which returns a deep-link to a “form of interest” with the keywords mapped (via the SO-properties identified for the keywords and form fields) to corresponding parameter names of the form at the deep-link. Fox, ¶107; Halevy, [0136-38].

POSAs would have thus recognized Belanger’s promotions with Halevy’s deep-links to each be a “*deep-linked ad linking one or more of a set of computer-readable field identifiers used by a first site or application with one or more of a set of computer-readable field identifiers used by a second site or application*” because the deep-link in the promotion includes the date and location values entered by the user in the fields on Belanger’s reservation web form (e.g., Fig. 37) and via the deep-link, the user would have been directed to a vendor website with corresponding fields (e.g., date and location) in the vendor’s form set to those values. Halevy, [0136-38], Fox, ¶108.

POSAs would have understood that in Belanger/Halevy *supra* § VI.C, the query submitted to Halevy’s system to retrieve deep-links generated for Belanger’s promotions can include keywords specific to the type of promotion being displayed. Fox, ¶109. For example, a query that includes a keyword specifying “hotel” promotions would be recognized by POSAs to return deep-links from Halevy in the SO-class for “hotel.” *Id.* In this case, POSAs would have understood that deep-links retrieved for hotel promotions would be in a different SO-class than Belanger’s car reservation website and thus would have recognized that “*the first site and the second site are in ... different verticals for which the spidering module is configured to search on the Internet.*” Fox, ¶109.

e. [1D] “placing the deep-linked ad on a page of the first site or application on the Internet;”

Belanger’s “website” discloses a *first site or application*. Belanger, Abstract, [0020]; Fox, ¶110.

As explained *supra* § VI.D.1.d.i (limitation [1C]), in Belanger/Halevy, Belanger’s “servlet” on the “application server” generates the *deep-linked ad* as part of deciding “data that should appear on the page,” which includes the “promotion[s]” with deep-links. Belanger, [0107-08], [0128], Fig. 53 (showing an example webpage with promotional deep-link 318); Fox, ¶110. As discussed *supra* § VI.C, POSAs would have understood that Belanger/Halevy would have displayed one or more deep-linked ads on Belanger’s webpages resulting from

Belanger's application server generating a promotion that includes a deep-link queried from Halevy's form database. Thus, Belanger/Halevy discloses "*placing the deep-linked ad on a page of the first site or application.*" Fox, ¶111.

f. [1E] "and in response to user interaction with the deep-linked ad on the page of the first site or application displayed on a user device:"

The claimed *deep-linked ad* is the "promotion" shown on Belanger's webpages or application (e.g., Fig. 53 or another webpage). *See supra* § VI.D.1.d.i (limitation [1C]). Belanger discloses that a "customer[]" or "*user*" of its website may "select" deep-links in the promotions, *i.e., interact with* them. Belanger, Abstract, [0019]; Fox, ¶113. This interaction happens *with the deep-linked ad on the page of the first site or application displayed on a user device*, since it happens on a webpage generated by Belanger's reservation website, which is the claimed *first site*. *See supra* § VI.D.1.e (limitation [1D]). POSAs familiar with web browsers would have understood that ads are displayed on *a user device* because Belanger discloses that users access its website through "customer computers" or "user computer[s]." Belanger, [0106], claim 1; Fox, ¶113.

As explained *infra* §§ VI.D.1.g-VI.D.1.i, the steps that happen *in response to* the user selecting a deep-link in Belanger's promotion meet the remaining limitations of claim 1. Fox, ¶114.

g. [1F] "extracting, from the user interaction with the deep-linked ad on the page of the first site or

application, values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application on the Internet;”

As discussed *supra* §§ VI.D.1.e and VI.C, placing an ad on Belanger’s website (e.g., Fig. 53 or another page) would have resulted in one or more promotions, each having a deep-link to a vendor’s website (generated by querying Halevy’s form database) with user-entered values (e.g., the date and location) entered on Belanger’s website (e.g., the Fig. 37 webpage) mapped to corresponding parameter names in each deep-link. Fox, ¶115. These date and location values are “*values corresponding to the at least one of the set of computer-readable field identifiers*” because they are the values entered into the fields in the Fig. 37 webpage (i.e., associated with each field’s parameter name), which are the “*set of computer-readable field identifiers used by the first site or application.*” See *supra* § VI.D.1.d.ii (limitation [1C]).

POSAs would have understood that Halevy’s deep-links include Belanger’s date and location values (as would have been used by Belanger’s application server to query Halevy’s form database), and that these values would have been *extracted from* Belanger’s website, for two alternative reasons. First, POSAs would have recognized that these values would have been *extracted from* Belanger’s website *in response* to selecting the deep-linked ad because Halevy teaches that for deep-links to websites that use JavaScript, selecting the deep-link directs a server to construct

an “invocation string” that “submit[s] the form automatically on behalf of the user” by concatenating the deep-link’s parameters “and their mapped words or internal value” to the invocation string and passing the completed string back to the browser to invoke the form. Fox, ¶116; Halevy, [0127-28].

Alternatively, under Patent Owner’s apparent construction in litigation, the browser loading the deep-link in the browser to invoke the form to pass the parameter values from the deep-link to the form constitutes *extracting* the values *from the website in response to the user selecting the deep-link ad*. Fox, ¶116; Ex-1022, 32 (infringement contention citing similar functionality); Ex-1021, 33 (same); Ex-1023, 24 (same). The Board need not determine whether Patent Owner’s apparent construction is correct, as the limitation is met regardless.

h. [1G] “automatically directing the user device to display a page of the second site or application; and”

As discussed *supra* §§ VI.C and VI.D.1.f, the user’s browser in Belanger/Halevy is taken to the vendor website, *i.e.*, the *page of the second site or application, in response to the user interacting with the deep-linked ad* by clicking a promotional deep-link. Fox, ¶117. POSAs would have understood that taking the user to the vendor website indicated by the URL in the deep-link in response to the user clicking the promotion constitutes *automatically directing the user device to display* the vendor website or application. As explained *supra* § VI.D.1.f for Limitation [1E], the user accesses Belanger’s reservation website via a computer.

Furthermore, POSAs would have understood that the user uses a “browser” application on the computer to access Belanger’s website, and thus would have recognized that a standard feature of a browser is that when a user clicks the link, the browser automatically directs the user to the website at that link without the need for further user intervention. Fox, ¶118; Jha, [0168]; Becker, 1.

- i. **[1H] “passing the values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application to the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application such that the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application are determined using the universal variable map and such that one or more fields on the page of the second set or application are explicitly or implicitly pre-populated with the values extracted from the user interaction with the deep-linked ad on the page of the first site or application without necessitating any pre-existing relationship between the pages.”**

As discussed above, POSAs would have understood Halevy to teach that a web form at a website can be automatically invoked using a “query-through link” including an “invocation string” containing field names (i.e., *computer-readable field identifiers*) set to certain values. *Supra* § VI.B.3, VI.D.1.g; Halevy, [0022-23], [0123]; Fox, ¶119. In the case of the ad served on Belanger’s webpages (e.g., Fig. 53) including a vendor deep-link with corresponding field names (i.e., *computer-readable field identifiers*) set to the date and location values (i.e., *values*

corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application), POSAs would have understood that the process of creating an invocation string that invokes the form in the deep-link includes *passing* the date and location values that had previously been entered by the user in the fields on Belanger’s website (e.g., Fig. 37) via the deep-link to corresponding fields on a form at the vendor’s website. Fox, ¶120.

In Belanger/Halevy, the invocation string constructed for the deep-links in Belanger’s promotions includes values provided on the Figure 37 webpage (e.g., the date and location values), and each deep-link is used to invoke the form on the vendor website with the values prepopulated in their corresponding fields to return a results page. Halevy, [0138], [0143]; Fox, ¶121; *supra* § VI.C. In view of Halevy’s above-discussed teachings regarding how form invocation works to display a results page by prepopulating a form’s fields with the values in the deep-link (*see* Halevy, [0138], [0143]), POSAs would have understood this process to involve “*passing the values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application*” (e.g., date and location values of the corresponding fields from Belanger’s reservation website) “*to the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application*” (e.g., date and location fields on the form at the vendor’s website) in order to display the results

from automatically searching the form at the vendor's website. Because the user in this case does not see the form, POSAs would have recognized that "*one or more fields on the page of the second set or application*" (e.g., the form invoked on the vendor's website) are "*implicitly pre-populated with the values extracted from the user interaction with the deep-linked ad on the page of the first site or application*" (e.g., as discussed *supra* § VI.D.1.g) in order to return the results of querying the form. Fox, ¶121.

Halevy also teaches that fields on the form at the vendor's site (e.g., the vendor's web form) can be "*explicitly*" prepopulated since the invocation string can also direct the user to "*a partially filled form and therefore expedite the query answering process (or correct the inputs to the form, if necessary)*." Halevy, [0143]. In this case, POSAs would have understood the Date and Location values in the deep-link to be *explicitly* prepopulated in corresponding fields of the partially filled form displayed to the user. Fox, ¶123.

As discussed *supra* §§ VI.D.1.c (limitation [1B]) and VI.D.1.d (limitation [1C]), Belanger/Halevy teaches that deep-linked ads *link* the sets of computer-readable field identifiers used by the first and second sites via the form database, which is a *universal variable map*. POSAs thus would have understood Belanger/Halevy to disclose that "*the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application are*

determined using the universal variable map.” Fox, ¶124. Further, since Halevy teaches that its form database is created via crawling and categorizing webpages and forms from various unrelated websites found on the Internet (Halevy, [0039-49], [0057-58], [0095]), POSAs also would have understood the form database’s creation, and thus the linking between fields on various forms to occur “*without necessitating any pre-existing relationship between the pages.*” Fox, ¶124.

2. Claim 2: “The method according to claim 1, further comprising: applying, by the spidering module via programming code, a set of enhancing rules to enhance results of the obtaining of the computer-readable field identifiers.”

Belanger/Halevy, as discussed *supra* § VI.C, discloses this limitation since Halevy, discussed *supra* §§ VI.B.1 and VI.D.1.b, teaches a crawling process (i.e., *the spidering module*) executed by a form engine that analyzes each form to determine information about the form and the form’s fields to create a searchable hierarchy of SO-classes and SO-properties in the form database. Halevy, [0037-38].

POSAs would have understood Halevy’s crawling process to use “*a set of enhancing rules to enhance the results of the obtaining*” of form fields (i.e., *computer-readable field identifiers*) because for each form, the crawling process determines “[t]he class of the form” (i.e., “the types of objects that can be obtained by querying the form” such as “books, hotels, restaurants” also referred to as SO-

class), the geographic locations relevant to the form, any “required fields,” “[o]utput properties” for the fields (e.g., SO-properties), and any “additional keywords that describe the form.” Halevy, [0045-49]; Fox, ¶126. This information for each form and field, is stored in the form database as shown in Figure 3, annotated below:

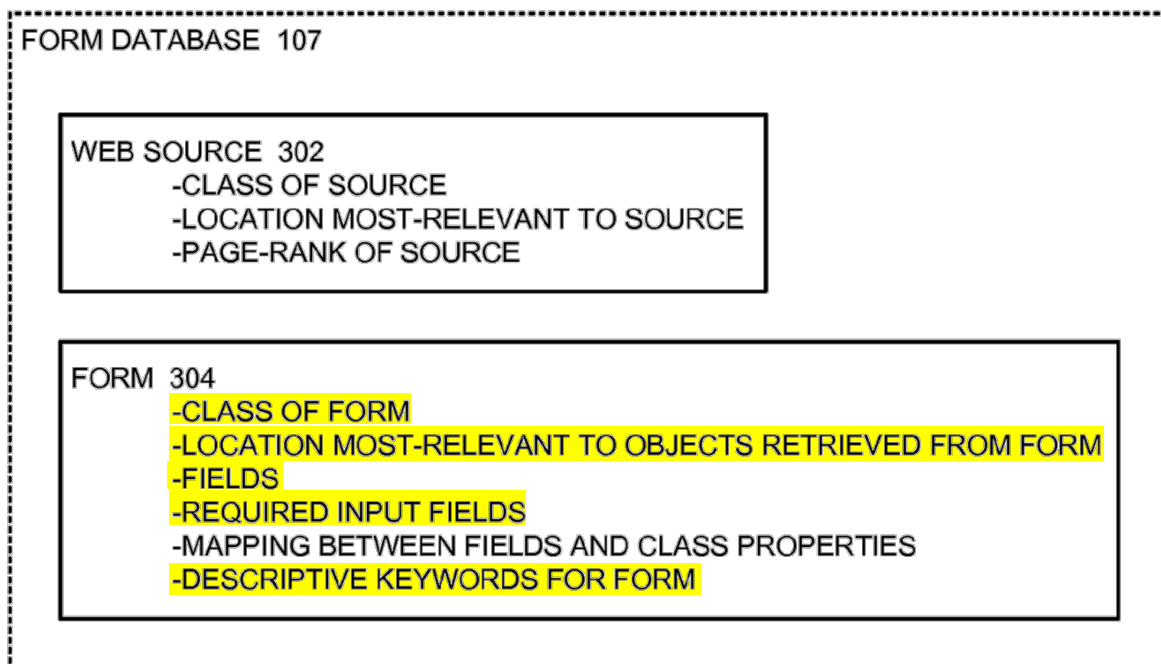


FIG. 3

POSAs would have understood that determining this information for each form “*enhance[s] the results of the obtaining*” because this information is used to “facilitate subsequent searching operations through content which is accessible through the web-based forms.” Halevy, [0036], [0037]; Fox, ¶127.

POSAs would have recognized that this analysis performed by the crawling process is “*appl[ied] via programming code*” because it is performed by Halevy’s

form engine that “generally exists on any computer device.” Halevy, [0034-35], [0037]; Fox, ¶127.

3. Claim 3: “The method according to claim 1, wherein the obtaining includes gathering information pushed from the sites or applications on the Internet.”

As discussed *supra* §§ VI.D.1.b, Belanger/Halevy adds forms to its form database via Halevy’s crawling process that *obtains* webpages to analyze web forms across the Internet. Halevy, [0034-35]; Fox, ¶128. Halevy teaches several methods for obtaining the websites that are crawled including “receiv[ing] manually provided web-based forms.” Halevy, [0009], [0035], [0130]. POSAs would have recognized “manually provided” webforms received by Halevy’s system to be *information gathered* via a “*push[] from the sites or applications on the Internet.*” Fox, ¶128.

4. Claim 4: “The method according to claim 1, wherein the obtaining includes running a programmatic query on a site or application, receiving a response page from the site or application responsive to the programmatic query, and parsing the response page to obtain a set of computer-readable field identifiers from the site or application.”

As discussed *supra* §§ VI.B.1 and VI.D.1.b, Belanger/Halevy adds forms to the form database via Halevy’s crawling process, executed by a form engine, that *obtains* webpages and web forms from across the Internet. Halevy, [0034-35]; Fox, ¶129. POSAs would have understood that crawling involves sending a query in the form of a URL to a webserver to obtain a webpage response from that webserver,

and then parsing the response webpage. Fox, ¶129. In addition to *obtaining* a webpage’s form, Halevy teaches that the form engine “also retrieve[s] pages from a given form by submitting queries to it” (i.e., *runs a programmatic query* to “receive a response page ... responsive to the programmatic query”). Halevy, [0132]. These queries allow the form engine to “extract[] the content” from the form by “extracting the content from associated output fields.” Halevy, [0011-14], [0133-35], Fig. 5. POSAs would have understood this querying by Halevy’s form engine to be a *programmatic query* because the form engine, as part of the crawling process, “obtain[s] constants associated with a given field of the given form from a drop-down menu associated with the given field,” then uses the constants “as inputs to the given field or as inputs to similar fields in other forms during subsequent query operations.” Halevy, [0014], [0133-35], Figs. 4, 5, Fox, ¶129.

Halevy also discloses that its crawling process *parses* the retrieved pages *to obtain a set of computer-readable field identifiers* (i.e., fields) because the crawling process’s querying of a form “produce[s] corresponding outputs,” which are then analyzed for “output properties” or “the SO-properties ... likely to” be found in the “answers” to the query. Halevy, [0017], [0048], [0133]. POSAs would have recognized that Halevy’s process of identifying “outputs” as SO-properties is *obtaining a set of computer-readable field identifiers* because, as discussed *supra* §

VI.B.3, the SO-properties are assigned to corresponding fields in the form so that the form can later be retrieved with the fields prepopulated based on the SO-properties. *Id.*, [0049]; Fox, ¶130.

Because the SO-properties act as mapping variables, POSAs would have recognized them to be “*a set of computer-readable field identifiers*” for the form, based on its SO-class in the form database. Fox, ¶131.

5. Claim 5: “The method according to claim 1, wherein the mapping is performed when the computer-readable field identifiers are obtained from the sites or applications on the Internet, when the deep-linked ad is automatically programmatically generated, or when the deep-linked ad is about to be served in real-time.”

As discussed *supra* §§ VI.B.2 and VI.D.1.c, Halevy’s form database is a “*universal variable map*” consisting of entries created by the form engine crawling forms such that the entry for each form includes the parameter names for the fields in the form (i.e., *computer-readable field identifier*), an identified class for the form, and a *mapping* of the forms fields to a set of properties in the identified class. Halevy, [0009], [0044-45], [0049], [0130-31], Fig. 3; Fox, ¶132. Because the form engine creates the database entries for each form during the crawling process (Halevy, [0037]), POSAs would have understood the *mapping* to occur “*when the computer-readable field identifiers*” or field parameter names “*are obtained from the sites or applications*” by Halevy’s crawler. Fox, ¶133. POSAs also would have

understood that “*when the deep-linked ad is automatically programmatically generated,*” Belanger/Halevy is making use of the mapping. *Id.*

6. Claim 6: “The method according to claim 1, wherein the spidering module parses forms from the sites or applications on the Internet and form data associated with the forms, identifies any product associated with the forms, and obtains product attributes from the form data for indexing.”

As discussed *supra* §§ VI.D.1.b, Halevy adds forms to the form database via a crawling process (i.e., “*the spidering module*”) executed by a form engine that *obtains* webpages to analyze web forms *from sites or application on the Internet*. Halevy, [0034-35]; Fox, ¶134. During the crawling process, each *form* is *parsed* to capture the “fields in the form” including “its parameter name (used to invoke some server-side script), the human-readable or visible text for that field, field types (text, select box, radio button, or checkbox), and exemplary values for the field (e.g., the options in a select box).” Halevy, [0043-50]. The *form data* is also *parsed* in that Halevy’s crawler also “retrieve[s] pages from a given form by submitting queries to it (step 408).” *Id.*, [0132-35], Figs. 4, 5. These pages are then stored “into a web index (or into some other type of database) along with other web pages to facilitate searches through the retrieved pages.” *Id.*, [0132], *see also* [0014]. POSAs would have recognized that the indexing of a results pages from querying a form used to “*identify any product associated with the forms*” because Halevy uses these results to identify the class of the form or “types of objects that

can be obtained by querying the form. For example, books, hotels, restaurants.” *Id.*, [0045]; Fox, ¶135.

POSAs would have recognized that the indexing of a results page from querying a form is “*obtain[ing] product attributes from the form data for indexing*” because Halevy teaches that this extracted form data is used to determine “output properties” or “the type of information we will obtain from answers to a successful form submission” (also used as SO-properties) or in other words, the “product attributes” for the identified form class (i.e., “types of objects that can be obtained by querying the form”). Halevy, [0045], [0048]; Fox, ¶136; *see also* Ex-1001, 14:28-31 (describing “identifying a product associated” with a form and then “obtaining product attributes from form data”). Further, POSAs familiar with search engines would have understood that indexing is an obvious part of the operation of search engines. Fox, ¶136.

7. Claim 7: “The method according to claim 1, wherein the first application and the second application are in same vertical or different verticals for which the spidering module is configured to search on the Internet.”

As discussed *supra* § VI.D.1.d.ii, Belanger/Halevy discloses that the deep-links generated by Halevy’s system for specific types of promotions (e.g., “hotels”) on Belanger’s reservation website are such that *the first site* (e.g., Belanger’s reservation website) *and the second site* (e.g., a hotel vendor’s website) *are in ...*

different verticals for which the spidering module is configured to search on the Internet. Fox, ¶137.

As discussed *supra* § VI.D.1.b, POSAs would have recognized that Halevy's crawling process executed by the form engine (i.e., "*the spidering module*") would have used the same process to crawl "applications" because it was known in the art (as acknowledged by the '342 patent) that "fields or variables on the publisher site or mobile app" would have been obtained and processed in the same manner using known communication protocols (e.g., SOAP). Fox, ¶138; Ex-1001, 1:53-58, 3:3-21; *see also* Tidwell, 10-12 (describing using SOAP to provide web services to web-based applications including "an Application Service Provider's vast server farm to a PDA"). Thus, POSAs would have likewise recognized that in Belanger/Halevy, the deep-link is such that "*the first application and the second application are in ... different verticals for which the spidering module is configured to search on the Internet.*" Fox, ¶139.

8. Claims 8-21

Independent claim 8 recites in limitation [8PRE] a "*system comprising at least one processor and at least one non-transitory computer readable medium storing instructions translatable by the at least one processor to perform*" the same method recited in claim 1. *Supra* § VI.D.1. This method is performed in Belanger via "a reservation booking website hosted by automated reservation transaction

processor 150” (i.e., “*at least one processor*”). Belanger, [0106]. Belanger’s *processor* includes “an application server” that interacts with various databases based on interaction with the customer’s computer via a web server. *Id.* POSAs would have recognized the databases to be stored in “*non-transitory computer memory*” since the databases “store[] the profiles of any registered customers, the profiles of any corporate accounts, and data relating to any rental promotions being offered by the company.” *Id.*; Fox, ¶140. Belanger/Halevy thus teaches an application server that has *at least one processor* and *non-transitory computer memory*, which a POSA would have understood is a *system*. Fox, ¶140; Belanger, [0106-07]. Belanger also discloses that its application server executes “software” to facilitate its website. Belanger, [0107-08], Fig. 7. Thus, POSAs would have recognized the application server’s memory to “*stor[e] instructions translatable by the at least one processor to perform*” the steps in limitations [8A]-[8H] for the same reasons discussed *supra* §§ VI.D.1.b–VI.D.1.i. Fox, ¶141.

Independent claim 15 recites in limitation [15PRE] a “*computer program product*” identical to the *system* in claim 8. The software on Belanger’s application server that implements Belanger’s website is a *computer program product* (Belanger, [0107]), and the remainder of claim 15 is met by Belanger/Halevy for the same reasons discussed above for claim 8. Fox, ¶142.

Claims 8 and 15 are otherwise identical to claim 1 (except to exclude the “*spidering module*”) and are met by Belanger/Halevy for the same reasons summarized in the following table and explained in the sections indicated. Fox, ¶143.

Claim 8 Element	Claim 15 Element	Corresponding Claim 1 Element & Section
[8A]	[15A]	[1A] – VI.D.1.b
[8B]	[15B]	[1B] – VI.D.1.c
[8C]	[15C]	[1C] – VI.D.1.d
[8D]	[15D]	[1D] – VI.D.1.e
[8E]	[15E]	[1E] – VI.D.1.f
[8F]	[15F]	[1F] – VI.D.1.g
[8G]	[15G]	[1G] – VI.D.1.h
[8H]	[15H]	[1H] – VI.D.1.i

The dependent claims from claims 8 and 15 recite identical limitations to the corresponding claim 1 dependents listed below (except to exclude the *spidering module*), and are met for the reasons summarized in the following table and explained in the sections indicated. Fox, ¶144.

Claim 8 Dependent	Claim 15 Dependent	Corresponding Claim 1 Dependent & Section
9	16	2 – VI.D.2
10	17	3 – VI.D.3
11	18	4 – VI.D.4
12	19	5 – VI.D.5
13	20	6 – VI.D.6
14	21	7 – VI.D.7

VII. GROUND 2: CLAIMS 1-21 ARE OBVIOUS OVER BELANGER AND APA IN VIEW OF HALEVY

As discussed *supra* § VI.D, Belanger in view of Halevy (Ground 1) renders claims 1-21 obvious.

To the extent that explicit disclosure is required regarding travel websites utilizing deep-link advertising, as evidence of the knowledge of the art, the Applicant Admitted Prior Art (“APA”) states that at the time of the ’342 patent, such sites were known, as were applications doing the same. Ex-1001, 2:49-55, 3:2-21. Specifically, Applicants admitted that deep-linked-based advertising was known and in use for advertising on “search and E-commerce sites where the user is searching for a product or service,” like the “travel sites” described in Belanger. *Id.*, 2:49-55; Fox, ¶146. With these sites, a user entered search parameters (e.g., “where s/he is traveling from, where to, travel dates, how many people in the party, class of travel, etc.”) and the parameters were “passed along with each ad click to the appropriate advertiser” to “automatically activate searches from their home page,” thereby deep-linking the user “into their booking process.” Ex-1001, 2:51-63.

These ads worked by mapping the “fields or variables on the publisher site or mobile app” to the “corresponding fields or variables on the advertiser site or mobile apps.” *Id.*, 3:7-12. This mapping was done via manual coding “by developing programmatic integrations between variables and fields” on the

publisher's and advertiser's websites. *Id.*, 3:12-16. Manual coding was necessary because each website was known to use "its own field descriptors and variable names." *Id.*, 3:16-21.

POSAs would have recognized the partner and reservation websites described in Belanger to be examples of publisher websites and applications described in the APA, having fields that require manual coding to map their search parameters to the deep-links provided by Belanger so that their search parameter values would have been passed along in the promotional deep-links on Belanger's reservation website. Fox, ¶149.

POSAs would have known that automating this mapping process would have been more efficient, as the APA states that mapping via manual programming was "a time and resource intensive activity that multiplies as the size of the network grows, limiting the size, scale and market adoption for deep-linked ads." Ex-1001, 3:22-25. POSAs would have thus looked to Halevy to automate this manual mapping process, as Halevy's form database, discussed supra §§ VI.B and VI.C, maps the forms and fields on various websites to a hierarchy of SO-classes and SO-properties that can be queried with keywords to retrieve deep-links that map the keywords to parameter names of the form at each deep-link to invoke a search on the linked website's form. Fox, ¶150; Halevy, [0038], [0136-38].

VIII. NO BASIS EXISTS FOR DISCRETIONARY DENIAL

A. Section 314(a)

“[T]he PTAB will not deny institution based on *Fintiv* if,” as here, “there is compelling evidence of unpatentability.” Director’s Interim Procedure for Discretionary Denials, 5 (June 21, 2022) (“Interim Procedure”) (discussing *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 (Mar. 20, 2020) (precedential)). That should conclude the *Fintiv* analysis.

If considered, the *Fintiv* factors also weigh against denial.

1. Stay Potential

No stay has been requested. Thus, without “specific evidence” of how the court will rule, this factor is at worst neutral. *Sand Revolution II, LLC, v. Cont’l Intermodal Group–Trucking*, IPR2019-01393, Paper 24, at 7 (June 16, 2020) (informative).

2. Trial Timing

No trial date has been set. Trial would be expected well after a final written decision regardless of the outcome of Petitioner’s pending motion to transfer from Delaware to N.D. Cal. *See* Ex-1010, 2. Median time to trial in Delaware is 32.9 months. *See* Ex-1011, 14. Median time to trial in N.D. Cal. is 48.9 months. *See* Ex-1011, 66.

3. Litigation Involvement

District-court activity has been minimal, with no rulings on Petitioner’s Motion to Dismiss or Motion to transfer. *See* Ex-1010. There has been no exchange of contentions, claim construction activity, or decisions on the merits. *Samsung Elecs. v. Netlist, Inc.*, IPR2022-00063, Paper 13, at 8 (May 5, 2022) (granting institution where no “claim construction and merits determinations” had occurred).

4. Issue Overlap

Given its early stage, the validity issues in the litigation are unknown. However, Petitioner challenges all claims, whereas Metarail’s complaint only identifies one asserted claim. Ex-1012, ¶55. This weighs against denial. *Markforged Inc. v. Continuous Composites*, IPR2022-00679, Paper 7, at 32-33 (Oct. 25, 2022).

5. Litigation Defendants

That Petitioner is the defendant in the district-court proceeding “is far from an unusual circumstance.” *Sand Revolution II*, at 12-13. This factor is neutral.

6. Other Circumstances

The Petition’s “strong showing on the merits” weighs against denial. Interim Procedure, 3-5.

B. Section 325(d)

No basis exists for denial under Section 325(d). While Stephens (U.S. Pub. 2007/0239494) is cited in the '342 patent, was applied by the Examiner during prosecution of U.S. Patent No. 9,633,378 (the parent of the '342 patent), and lists Belanger as “related” and incorporated by reference (Ex-1016, [0001]), Stephens is not in the same family as Belanger and does not include the same disclosure. The portions of Stephens applied by the Examiner include [0007-08], [0053], [0056-62], [0135], and Fig. 2. *See* Ex-1017, 206-19, 283-93, 325-35. These portions describe deep-links, and more specifically, division-specific deep-links for a customer manually created by an administrator. These portions, however, do not describe Belanger’s website as the claimed *first site*, the fields in Belanger’s website being used to repurpose user-entered data as was known in the art, nor Belanger’s website hosting promotional deep-links to other travel websites as suggested by Belanger, or by querying Halevy’s form database via web service, as applied above. Thus, the application of Belanger here is not cumulative to what the Examiner applied from Stephens during prosecution of the '342 patent.

Moreover, Halevy was not cited or applied during prosecution of the '342 and '378 patents, nor were any references cumulative to it considered. Accordingly, the combination of Belanger/Halevy applied here is not cumulative to the prosecution record. Thus, this Petition provides new art and arguments not

previously considered and no basis exists for discretionary denial. *Advanced Bionics, LLC v. MED-EL Elektromedizinische Geräte GmbH*, IPR2019-01469, Paper 6 (PTAB Feb. 13, 2020) (precedential).

IX. CONCLUSION

The Board should institute review and cancel claims 1-21.

Dated: October 2, 2024

Respectfully submitted,
Google LLC

/Scott A. McKeown/

Scott A. McKeown, Reg. No. 42,866
WOLF, GREENFIELD & SACKS, P.C.

X. CLAIM LISTING

The following claim listing assigns element labels (e.g., [1A], [1B], etc.) to certain claims for clarity.

Claim 1
[1PRE] A method, comprising:
[1A] obtaining or inferring computer-readable field identifiers from sites or applications on the Internet, the obtaining performed by a spidering module running on a computer having at least one processor and non-transitory computer memory;
[1B] mapping the computer-readable field identifiers to one another using normalized variables stored in a normalized variable data database, the normalized variable data database representing a universal variable map of searchable classes and attributes of products or services on the Internet, the mapping linking computer-readable field identifiers of the sites or applications on the Internet;
[1C] automatically programmatically generating a deep-linked ad utilizing the universal variable map, the deep-linked ad linking at least one of a set of computer-readable field identifiers used by a first site or application on the Internet with at least one of a set of computer-readable field identifiers used by a second site or application on the Internet, wherein the first site and the second site are in same vertical or different verticals for which the spidering module is configured to search on the Internet;
[1D] placing the deep-linked ad on a page of the first site or application on the Internet;
[1E] and in response to user interaction with the deep-linked ad on the page of the first site or application displayed on a user device:
[1F] extracting, from the user interaction with the deep-linked ad on the page of the first site or application, values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application on the Internet;
[1G] automatically directing the user device to display a page of the second site or application; and
[1H] passing the values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application to the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application such that the correspondingly linked at least one of the set of computer-readable field identifiers used by the

second site or application are determined using the universal variable map and such that one or more fields on the page of the second set or application are explicitly or implicitly pre-populated with the values extracted from the user interaction with the deep-linked ad on the page of the first site or application without necessitating any pre-existing relationship between the pages.

Claim 2

The method according to claim 1, further comprising: applying, by the spidering module via programming code, a set of enhancing rules to enhance results of the obtaining of the computer-readable field identifiers.

Claim 3

The method according to claim 1, wherein the obtaining includes gathering information pushed from the sites or applications on the Internet.

Claim 4

The method according to claim 1, wherein the obtaining includes running a programmatic query on a site or application, receiving a response page from the site or application responsive to the programmatic query, and parsing the response page to obtain a set of computer-readable field identifiers from the site or application.

Claim 5

The method according to claim 1, wherein the mapping is performed when the computer-readable field identifiers are obtained from the sites or applications on the Internet, when the deep-linked ad is automatically programmatically generated, or when the deep-linked ad is about to be served in real-time.

Claim 6

The method according to claim 1, wherein the spidering module parses forms from the sites or applications on the Internet and form data associated with the forms, identifies any product associated with the forms, and obtains product attributes from the form data for indexing.

Claim 7

The method according to claim 1, wherein the first application and the second application are in same vertical or different verticals for which the spidering module is configured to search on the Internet.

Claim 8

[8PRE] A system, comprising: at least one processor; and at least one non-transitory computer readable medium storing instructions translatable by the at least one processor to perform:

[8A] obtaining or inferring computer-readable field identifiers from sites or applications on the Internet;

[8B] mapping the computer-readable field identifiers to one another using normalized variables stored in a normalized variable data database, the normalized variable data database representing a universal variable map of searchable classes and attributes of products or services on the Internet, the mapping linking computer-readable field identifiers of the sites or applications on the Internet;
[8C] automatically programmatically generating a deep-linked ad utilizing the universal variable map, the deep-linked ad linking at least one of a set of computer-readable field identifiers used by a first site or application on the Internet with at least one of a set of computer-readable field identifiers used by a second site or application on the Internet, wherein the first site and the second site are in same vertical or different verticals for which the spidering module is configured to search on the Internet;
[8D] placing the deep-linked ad on a page of the first site or application on the Internet;
[8E] and in response to user interaction with the deep-linked ad on the page of the first site or application displayed on a user device:
[8F] extracting, from the user interaction with the deep-linked ad on the page of the first site or application, values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application on the Internet;
[8G] automatically directing the user device to display a page of the second site or application; and
[8H] passing the values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application to the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application such that the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application are determined using the universal variable map and such that one or more fields on the page of the second set or application are explicitly or implicitly pre-populated with the values extracted from the user interaction with the deep-linked ad on the page of the first site or application without necessitating any pre-existing relationship between the pages.
Claim 9
The system of claim 8, wherein the instructions are further translatable by the at least one processor to apply a set of enhancing rules to enhance results of the obtaining of the computer-readable field identifiers.
Claim 10

The system of claim 8, wherein the obtaining includes gathering information pushed from the sites or applications on the Internet.
Claim 11
The system of claim 8, wherein the obtaining includes running a programmatic query on a site or application, receiving a response page from the site or application responsive to the programmatic query, and parsing the response page to obtain a set of computer-readable field identifiers from the site or application.
Claim 12
The system of claim 8, wherein the mapping is performed when the computer-readable field identifiers are obtained from the sites or applications on the Internet, when the deep-linked ad is automatically programmatically generated, or when the deep-linked ad is about to be served in real-time.
Claim 13
The system of claim 8, wherein the instructions are further translatable by the at least one processor to parse forms from the sites or applications on the Internet and form data associated with the forms, identify any product associated with the forms, and obtain product attributes from the form data for indexing.
Claim 14
The system of claim 8, wherein the first application and the second application are in same vertical or different verticals for which the system is configured to search on the Internet.
Claim 15
[15PRE] A computer program product comprising at least one non-transitory computer readable medium storing instructions translatable by at least one processor to perform:
[15A] obtaining or inferring computer-readable field identifiers from sites or applications on the Internet;
[15B] mapping the computer-readable field identifiers to one another using normalized variables stored in a normalized variable data database, the normalized variable data database representing a universal variable map of searchable classes and attributes of products or services on the Internet, the mapping linking computer-readable field identifiers of the sites or applications on the Internet;
[15C] automatically programmatically generating a deep-linked ad utilizing the universal variable map, the deep-linked ad linking at least one of a set of computer-readable field identifiers used by a first site or application on the Internet with at least one of a set of computer-readable field identifiers used by a second site or application on the Internet, wherein the first site and the second

site are in same vertical or different verticals for which the spidering module is configured to search on the Internet;
[15D] placing the deep-linked ad on a page of the first site or application on the Internet;
[15E] and in response to user interaction with the deep-linked ad on the page of the first site or application displayed on a user device;
[15F] extracting, from the user interaction with the deep-linked ad on the page of the first site or application, values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application on the Internet;
[15G] automatically directing the user device to display a page of the second site or application; and
[15H] passing the values corresponding to the at least one of the set of computer-readable field identifiers used by the first site or application to the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application such that the correspondingly linked at least one of the set of computer-readable field identifiers used by the second site or application are determined using the universal variable map and such that one or more fields on the page of the second set or application are explicitly or implicitly pre-populated with the values extracted from the user interaction with the deep-linked ad on the page of the first site or application without necessitating any pre-existing relationship between the pages.
Claim 16
The computer program product of claim 15, wherein the instructions are further translatable by the at least one processor to apply a set of enhancing rules to enhance results of the obtaining of the computer-readable field identifiers.
Claim 17
The computer program product of claim 15, wherein the obtaining includes gathering information pushed from the sites or applications on the Internet.
Claim 18
The computer program product of claim 15, wherein the obtaining includes running a programmatic query on a site or application, receiving a response page from the site or application responsive to the programmatic query, and parsing the response page to obtain a set of computer-readable field identifiers from the site or application.
Claim 19
The computer program product of claim 15, wherein the mapping is performed when the computer-readable field identifiers are obtained from the sites or applications on the Internet, when the deep-linked ad is automatically

programmatically generated, or when the deep-linked ad is about to be served in real-time.

Claim 20

The computer program product of claim 15, wherein the instructions are further translatable by the at least one processor to parse forms from the sites or applications on the Internet and form data associated with the forms, identify any product associated with the forms, and obtain product attributes from the form data for indexing.

Claim 21

The computer program product of claim 15, wherein the first application and the second application are in same vertical or different verticals for which the system is configured to search on the Internet.

Claim 25

The computer program product of claim 23, wherein the deep-linked ad is generated or refined based on contextual information provided by a user's implicit action on the page of the first site or application.

Claim 26

The computer program product of claim 23, wherein the user interaction with the deep-linked ad comprises the user explicitly or implicitly selecting the deep-linked ad.

CERTIFICATE OF SERVICE UNDER 37 C.F.R. § 42.6 (E)(4)

I certify that on October 2, 2024, I will cause a copy of the foregoing document, including any exhibits or appendices filed therewith, to be served via Overnight FedEx at the following correspondence address of record for the patent:

Metarail, Inc.
1901 Central Drive
#716
Bedford, TX 76021

Date: October 2, 2024

/MacAulay Rush/
MacAulay Rush
Paralegal
WOLF, GREENFIELD & SACKS, P.C

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24, the undersigned certifies that the foregoing Petition for *Inter Partes* Review contains 13,064 words excluding a table of contents, a table of authorities, Mandatory Notices under § 42.8, a certificate of service or word count, or appendix of exhibits or claim listing. Petitioner has relied on the word count feature of the word processing system used to create this paper in making this certification.

Date: October 2, 2024

/MacAulay Rush/
MacAulay Rush
Paralegal
WOLF, GREENFIELD & SACKS, P.C