Exhibit 6M

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Apple Inc., Google Inc., and Motorola Mobility LLC

Petitioners,

v.

Arendi S.A.R.L.

Patent Owner.

Case No. IPR2014-00206

Patent No. 7,496,854

PATENT OWNER ARENDI S.A.R.L.'S PRELIMINARY RESPONSE UNDER 35 U.S.C. § 313 and 37 C.F.R. § 42.107

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Cases

Statutes

EXHIBIT LIST

Arendi Exhibit Number	Description
2001	American Heritage College dictionary 3rd edition 1997 definition of the term "dictionary."
2002	American Heritage College dictionary 3rd edition 1997 definition of the term "designate."

I. INTRODUCTION

Patent Owner Arendi S.A.R.L. ("Arendi" or "Patent Owner") respectfully requests that the Board decline to initiate *inter partes* review of claims 19-35, 57-85, 96, and 99 of U.S. Patent No. 7,496,854 (the "'854 Patent") because Petitioners Apple Inc., Google Inc., and Motorola Mobility LLC ("Petitioners") have failed to show that they have a reasonable likelihood of prevailing with respect to any of the challenged claims. 35 U.S.C. § 314.

The Petitioners have submitted proposed grounds for challenge based on anticipation or obviousness. However, for each proposed ground, Petitioners' allegations fail to state a ground for invalidation under the referenced section of U.S. patent law, and/or at least one claim element is missing from the relied-upon reference or combination of references. Thus, the Petitioners have failed to meet their initial burden to show that each element was known in the prior art.

This Preliminary Response will assist the Board in identifying elements in the independent claims which are not shown in the references cited. For each dependent claim, the Petitioners rely on the presentation relative to its independent claim. Given that each dependent claim incorporates all of the elements of its independent claim, these omissions carry through to the dependent claims.

Therefore, the dependent claims should likewise not be subjected to an *inter partes* review for Petitioners' lack of a likelihood of prevailing.

II. OVERVIEW OF THE '854 PATENT

The '854 Patent is directed, among other things, to computer-implemented processes for automating a user's interaction between a first application, such as a word processing application or spreadsheet application, on the one hand, and a second application, such as contact management application having a database, on the other hand. In the '854 Patent, Exhibit 1001, Figs. 1 and 2 are flow charts showing for these interactions a number of scenarios, which are described from col. 4, lines 25-39. Further details of the interactions are provided in discussion thereafter of the other figures of the '854 Patent, and the discussion includes references back to relevant portions of the flow charts in Figs. 1 and 2. Fig. 1 is reproduced below.

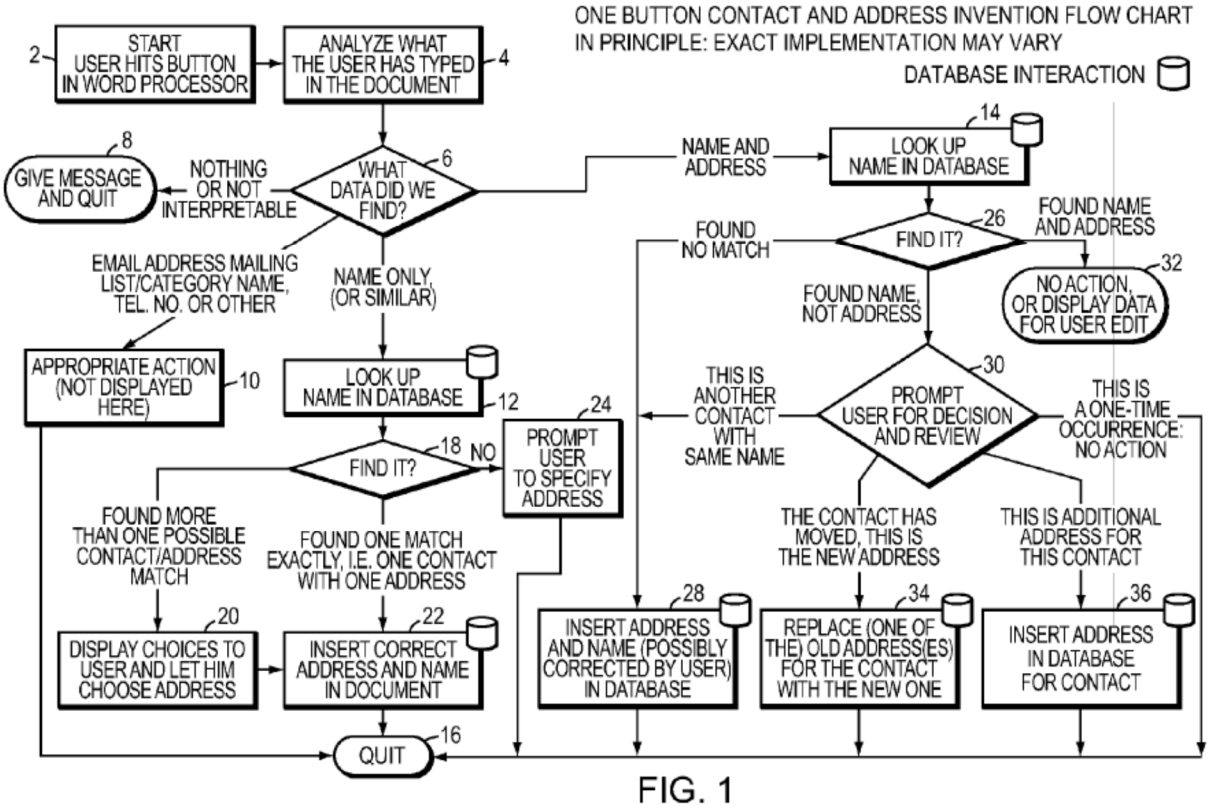


FIG. 1

In various scenarios, text in a document in the first application is analyzed (in step 4 of Fig. 1) to identify contact information. Exhibit 1001, col. 4, lines 25-39. The analysis takes place without user designation of a specific part of the document to be subject to the analyzing. *Id*.

Once contact information has been identified, a number of different scenarios can follow, depending on the circumstances. In one scenario, if the identified contact information includes a name, a search is initiated in the database associated with the second application for the name. *Id*., Fig. 1, steps 6, 12, and 14. If the contact information identified in the document included only a name, and if only a single entry is found in the database for the name and the entry

includes a single address, then the address is inserted into the document. *Id.*, Fig. 1, steps 6, 12, 18, and 22; Fig. 4; col. 5, line 61 – col. 6, line 5, which is reproduced below, shows the document displayed in Microsoft Word after the address has been inserted.

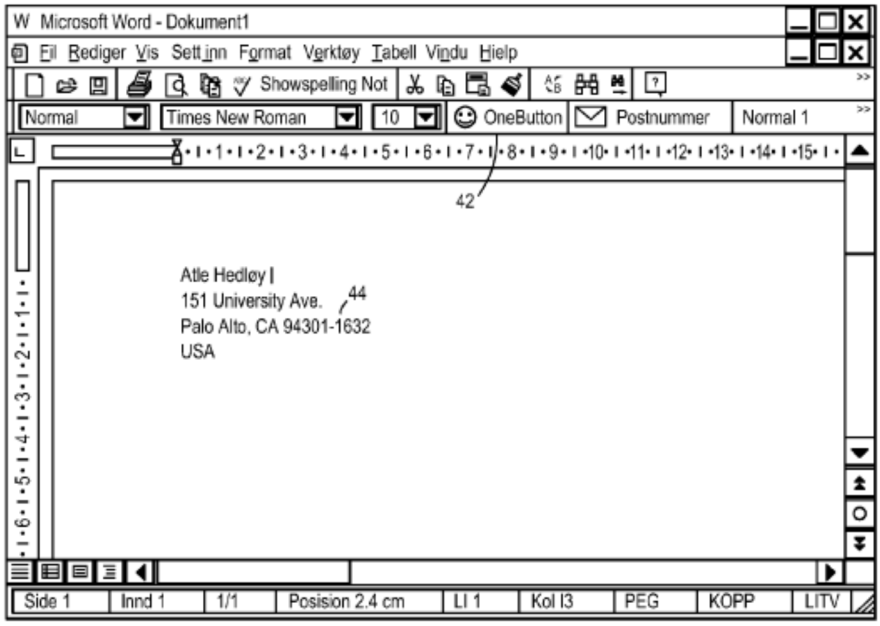


FIG. 4

Shown in Fig. 4 is the One Button 42, which, when pressed, launches the processes just recited, including analyzing the document to identify contact information, the searching in the database, and inserting of the address. *Id.*, Fig. 2, step 1; col. 4, lines 19-24; col. 5, line 61- col. 6, line 5.

On the other hand, if multiple addresses are found in searching the database for the identified name, these found addresses are displayed, and the user is

presented with a choice of which of the addresses to insert. *Id.*, Fig. 1, steps 18, 20, and 22; Fig. 10; col. 7, line 27 to col. 8, line 7.

In another scenario, when the user clicks on the "One Button" while viewing a document that includes a name and an address, the document is analyzed as before (per Fig. 1, step 4) to identify the name and the address. Next, the database is searched for the identified name (per Fig. 1, step 14). If the name happens to be in the contact database but the address in the contact database for that name differs from the address typed by the user into the document (per Fig. 1, step 26), then the user is prompted to make a choice (per Fig. 1, step 30). The user is presented with screen shown in Fig. 9, which is reproduced below.

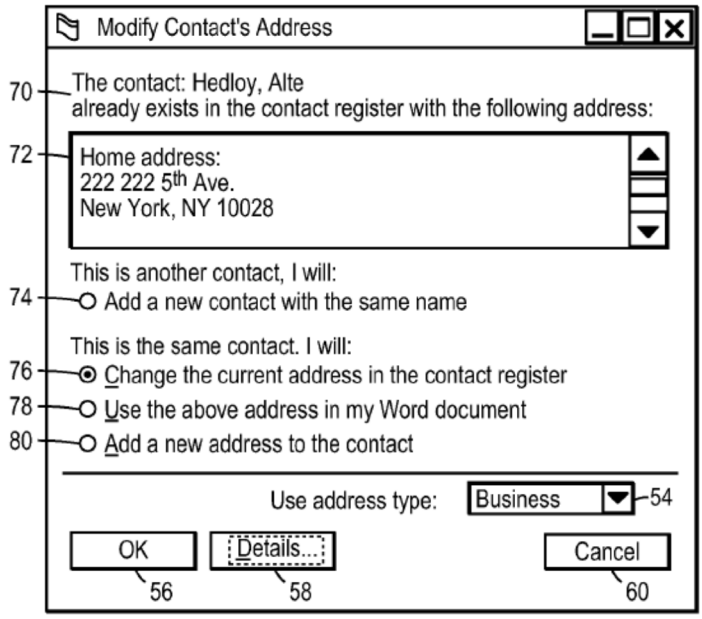


FIG. 9

Fig. 9 represents a screen presented to the user in which the user is given a series choices that can be made in this specific context. *Id.*, col. 6, line 63 – col. 7,

line 23. The screen reproduces the name that is both in the document and in the contact database, and it also displays the address that is in the contact database for that name. Below this information, the screen offers a total of four choices in two categories. As shown in Fig. 9 and explained in the '854 Patent, the user is enabled to select one of the four choices. *Id*. The first category is that "This is another contact," and the choice under this category is to "Add a new contact with the same name". The second category is that "This is the same contact", and the user is given three other choices for the contact: (a) "Change the current address in the contact register"; (b) "Use the above address [reproduced from the contact database] in my Word document"; and (c) "Add a new address to the contact".

These same four choices are also illustrated in connection with item 30 of Fig. 1 of the '854 Patent, which shows logical flow followed in described embodiments of the invention. Item 30 is labeled "PROMPT USER FOR DECISION AND REVIEW", and there are four outcomes shown from this item: (1) "THIS ANOTHER CONTACT WITH THE SAME NAME"; (2) "THE CONTACT HAS MOVED, THIS IS THE NEW ADDRESS"; (3) "THIS IS A ONE-TIME OCCURRENCE: NO ACTION"; and (4) "THIS IS ADDITIONAL ADDRESS FOR THIS CONTACT". These choices are described in the '854 Patent, col. 4, line 64 – col. 5, line 8.

It can be seen that the first of the four choices is to add a new contact, and two of the remaining choices are specific ways of updating an existing contact. (Another choice offered is to do neither of these and simply use the address in the Word document as typed.) Consequently, the screen of Fig. 9 presents to the user a choice, among other things, between competing alternatives of storing a new contact or updating an existing contact.

III. CLAIM CONSTRUCTION

In an *inter partes* review, the Patent Trial and Appeal Board gives patent claims their "broadest reasonable interpretation in light of the specification of the patent." 37 C.F.R. § 42.100(b) *Phillips v. AWH Corp.*, 415 F.3d 1303, 1316 (Fed. Cir. 2005) (*en banc*). The prosecution history is also relevant to identify the correct construction of claim terms. *Phillips v. AWH Corp.*, 415 F.3d at 1317. Extrinsic evidence may also be relevant to establish the meaning of terms, but such evidence is only relevant to the extent it is consistent with the specification and file history. *Id.*, 1319.

Patent Owner Arendi proposes construction of certain claim terms below pursuant to the specification standard. The proposed claim constructions are offered for the sole purpose of this proceeding and thus do not necessarily reflect appropriate claim constructions to be used in litigation and other proceedings wherein a different claim construction standard applies.

A. "Associated"

The term "associated" should be construed in accordance with its ordinary and customary meaning and the clear usage of the term within the intrinsic evidence as "a pre-existing connection or relationship". The '854 patent refers to the searching of a database for additional contact information (e.g. physical and e-mail addresses, phone numbers) that is "related" to text, identified in a Word document (e.g. a name), that is in a shared entry in the database. See for example the Abstract, col. 3 lines 63-66, col. 5 line 66, col. 6 line 2, col.4 lines 43-45, 57-58.

B. "Second information associated with the first information from a second application program"

The phrase "second information associated with the first information from a second application program" should be construed in accordance with its ordinary and customary meaning, to indicate that the second application program contains both the "first information" and the "second information" and associates the "second information" with the "first information" such that, by virtue of that association, a "first information" may be used to locate the "second information" within the second application program. This is consistent with the description of the term "information management source" as used in the specification of the '854 patent.

C. "User Designation"

The term "user designation" should be construed in accordance with its ordinary and customary meaning as meaning an action, by a user, to "indicate or specify; point out." See, e.g., the definition of "designate" from the Heritage College dictionary 3rd edition 1997 in Patentee's Exhibit 2002.

D. "Application Program"

The term "application program" should be construed in accordance with its ordinary and customary meaning and the clear usage of the term within the intrinsic evidence as a "an independently executable computer program designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." The patentee has used the term "application program" to refer to word processors, spreadsheet applications and contact managers within the specification such as Microsoft Word, Microsoft EXCEL and Microsoft Outlook. See col. 8 lines 30-33 and 57-67. See also Fig. 1-3 that show Microsoft Word and Example 7 beginning at Col. 8 line 55 entitled "Spreadsheet Application" that discloses using Microsoft EXCEL.

IV. OVERVIEW OF THE PRIOR ART

A. Overview of LiveDoc

LiveDoc concerns structure detection within a document where a "structure" represents meaningful bits of syntactically- regular information. LiveDoc allows a user to perform a function based upon an identified structure. To accomplish this goal, LiveDoc constructs "a means of passing text from a user's document for matching against a collection of recognizers." Exhibit 1005 at page 53. Thus, LiveDoc operates outside of any application program and outside of the document under the control of the application program.

The LiveDoc architecture is shown in Fig. 3 at page 56 where the LiveDoc manager communicates with an external application (i.e., a text editor) using API callbacks. See Exhibit 1005 page 57, left column.

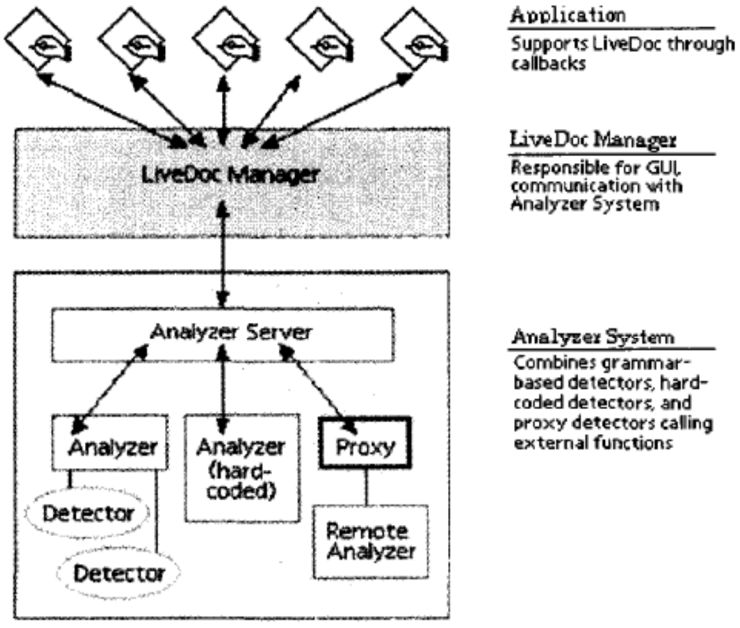


Figure 3: The high-level LiveDoc architecture

The LiveDoc application receives only the text from the text editor (application in Fig. 3) and analyzes the text independently of the

actual document in the text editor using a set of detectors under the control of an analyzer server.

In order for the discovered structures to be visible to a user, the user must enter "LiveDoc mode" by pressing the function key causing the LiveDoc Manager to update "the display to present the highlight information over the discovered structures." *Id.* at page 56. The user can then use the mouse to move over a highlighted item and press the mouse button that causes the LiveDoc Manager to present a menu of functions associated with the highlighted item.

LiveDoc knows where these structures appear in the text passed to it- an e-mail address might appear in characters 150 through 162 of the window's contents –but it has no idea where in the window those characters physically appear, and, thus, where the highlights should appear: this is information held by the application, not by LiveDoc. Hence, LiveDoc must ask the application for the information about the structures it has found via a callback. Once this information is available, the highlights and their associated mouse-sensitive regions can be constructed.

The overlaid highlights are independent and separate from the text editor and the document. Fig. 2, reproduced below, shows some of the actions that LiveDoc allows for a recognized structure.

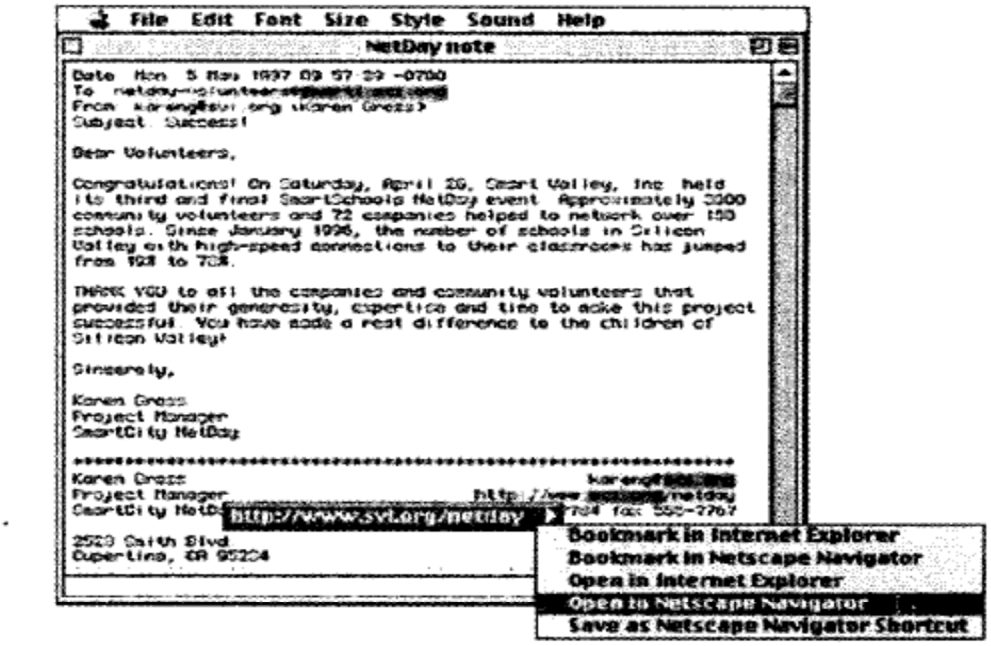


Figure 2: A sample interaction with LiveDoc. Note the highlighting of the discovered structures, the menu of actions available or the selected structure, and the nested highlighting of nested structures.

Each of the functions shown involves using the recognized text with an external application.

> "Our initial implementation of LiveDoc as LiveSimpleText assumed that actions would be handled by external applications, such as a Web browser presenting the page pointed to by a URL:" Id. at 57.

B. Overview of Drop Zones

Drop Zones expands on LiveDoc wherein a user that has entered LiveDoc mode may be presented with an interface that interprets the meaning of the identified and selected structure and presents recommended appropriate actions.

For example, a user may select a structure by moving over the highlighted structure and selecting the structure. Drop Zones recognizes the structure (e.g., as a name) and highlights any of the assistant function that can operate on the identified structure (e.g. a name can be used with an e-mail assistant). A set of possible actions are then presented to the user for using the identified structure. "Drop Zones goes beyond LiveDoc in allowing the user to select some subset of those terms and drag them as a group" to be operated on by the assistant. Drop Zones at 62.

C. Overview of Domini [6,085,206]

Domini is directed to a combined spell checking and grammar checking module that operates within a word processing application. See, e.g., Ex. 1006 Abstract, col. 3 lines 1-12, col. 5 lines 1-8. Fig. 3 reproduced below shows a dialog box for the spell checking and grammar checking module after a user has selected the "spelling and grammar" command within the application program (e.g. word processing application).

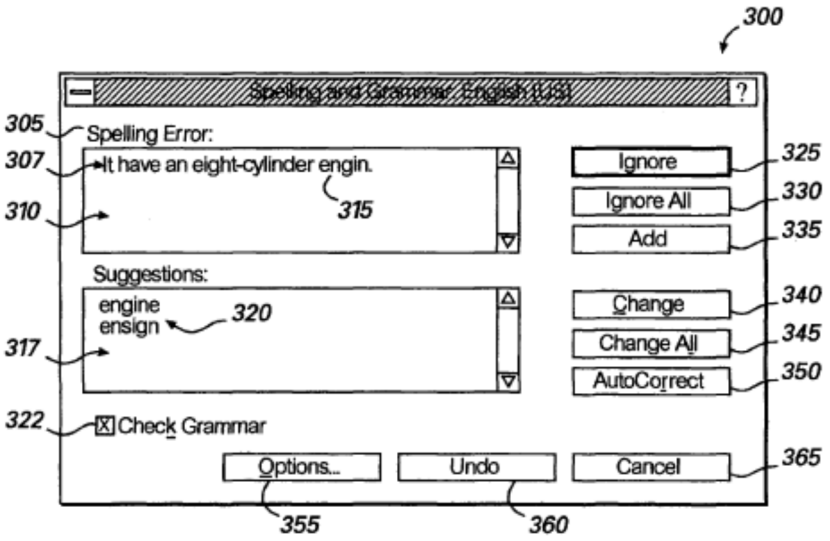


FIG.3

Upon selection of the spelling and grammar command within the word processing application, Domini extracts a sentence from the word processing document and calls the spell checking program module. *Id.* at col. 16 line 56-66. The spell checking program module selects a word from the sentence (i.e. a first word) and verifies whether the word appears in the one or more dictionaries of the spell checking program module. Col. 17 lines 19-42.

If the word does not appear within any dictionary, an error is generated indicating that the word is potentially misspelled, and the word processing application accesses a string buffer that contains one or more suggestions for the potentially misspelled word. The suggestions are displayed within a combined spelling and grammar checking dialog box. *Id.* at col. 18 line 4-20. If the user agrees that the word is misspelled, the user may then select between the one or more suggestions to replace the word within the document. *Id.* at col. 18 lines 21-26. Of course, the user may determine that the word is not misspelled, or that none of the presented options is correct (i.e., that the spell checker has guessed incorrectly about what the user intended to type).

D. Overview of Miller [5,946,657]

U.S. patent 5,946,657 to Miller et al (herein, "Miller") Petitioners Exhibit 1007) was before the USPTO in the prosecution of the '854 patent and is listed on the face of the '854 patent under "References Cited."

Miller issued on August 31, 1999 from an application filed February 1, 1996, and therefore qualifies as a reference under 35 U.S.C. §102(e), at best.

Miller discloses systems and methods for "detecting structures in data and performing actions on detected structures" (claim 1).

More particularly, Miller teaches a computer program (165; Fig. 2) that works outside of a document, such as a word processor document (210; Fig. 2).

Miller's program (165) includes an "analyzer server" (220) that "receives data having recognizable patterns from a document 210" (Abstract; also col. 3, lines 57-58).

After receiving "data having recognizable patterns from a document 210," Miller uses "pattern analysis units, such as a parser and grammars or a fast string search function and dictionaries" to find "recognizable structures" (col. 3, lines 57-64). Then, "Upon detection of a structure, analyzer server 220 links actions associated with the responsible pattern to the detected structure, using conventional pointers."

After structures are detected, an "application program interface" (230) within Miller's program (165; Fig. 2) subsequently "communicates with application 167 to obtain information on the identified structures so that user interface 240 can successfully present and enable selection of the actions" (col. 4, lines 2-5).

Miller's user interface (240) also makes "the presentation regions mouse-sensitive, i.e. aware when a mouse event such as a mouse-down operation is performed while the cursor is over the region" (col. 5, lines 35-37).

However, Miller's highlighting of the "detected structures" does not occur without user intervention. Miller describes this highlighting function in connection with Fig. 8 and Fig. 9.

Fig. 8, reproduced below, is a block diagram showing the flow of how Miller recognizes patterns and performs actions. More particularly, after initially detecting patterns and linking actions to the detected structures (flow chart boxes 810-830), Miller loops to determine whether the content has changed (box 840) or whether the region has changed (box 850), and if so loops back to earlier points in the flow.

Otherwise, Miller "continues to block 860" (col. 6, lines 3-4) where it determines whether the user has requested that the structures be displayed. As explained by Miller: "As illustrated by block 860, method 800 loops between blocks 840 and 860 until a request for display of identified structures is received 860" (col. 6, lines 4-6).

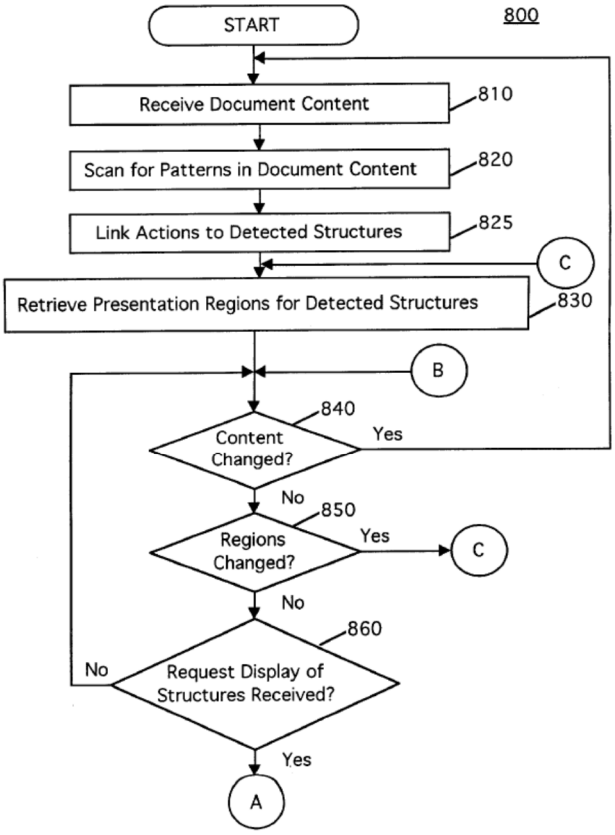


FIG. 8

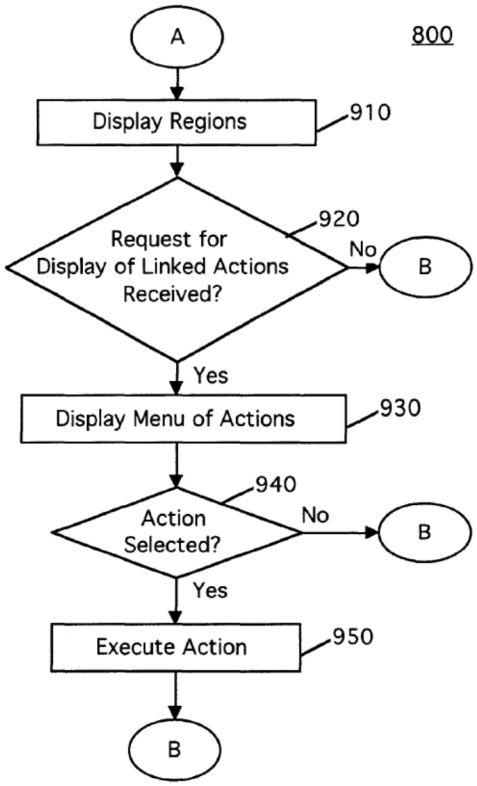


FIG. 9

Further, in order to know whether a given item of "recognized structure" might be somehow useful – e.g., via one or more of the "actions" linked to the detected structures - the user must take an action (another user intervention) by, for example, performing a "mouse-down operation over a structure" [col. 5, lines 38-39]. This requirement is also illustrated by box 920 of Miller's Fig. 9 above ("Request for Display of Linked Actions Received"), in which a request may be a "selection mechanism, such as a mouse-down operation over a detected structure, which causes the candidate actions linked to the structure to be displayed 930" [col. 6, lines 17-21]. Miller's Fig. 9 is copied above for ease of reference.

As illustrated by Miller's Fig. 8 and Fig. 9, Miller requires user intervention (e.g., "request for display of structures – Fig. 8 box 860 and request for display of linked actions" – Fig. 9, box 920) in order to highlight (i.e., mark) display a detected "structure," and indicate to the to the user that an action is available for the detected "structure."

If a user selects an available action, the available actions relate to use of the information in the detected "structure." As explained by Miller, "Upon selection of a candidate action, user interface 240 transmits the selected structure and the selected action to action processor 250. Action processor 250 retrieves the sequence of operations that constitute the selected action, and performs the sequence using the selected structure as the object of the selected action." (col. 4, lines 52-57).

The available actions disclosed by Miller do not encompass using the detected structure in association with some other information (e.g., information that might be called "second" information) in another application.

Rather, available actions disclosed by Miller are limited to using the information in the detected structure itself. See, for example, Miller's Fig. 7 and its related text, in which the only "actions" available for a detected phone number are to call the number or add the number to a phone book—neither of which requires second information. As explained by Miller: "In this example, pop-up

menu 710 displays the candidate actions linked to the selected telephone number grammar 410, including dialing the number and putting the number into an electronic telephone book. Upon selection of the action for putting the number in an electronic telephone book, user interface 240 transmits the corresponding telephone number and selected action to action processor 250. Action processor 250 locates and opens the electronic telephone book, places the telephone number in the appropriate field and allows the user to input any additional information into the file" (Miller at column 5, lines 40-50).

In summary, given data that includes "recognizable structures," Miller is capable of detecting those structures. However, highlighting such structures [Fig. 8], and indicating to the user that there are actions available for those structures, requires additional user intervention [Fig. 9]. Finally, Miller does not disclose any "second information."

E. Overview of Luciw [5,644,735]

Luciw describes logical processes, usable by a pen-based computer system that functions as a personal organizer, to provide "implicit or explicit assistance" for "user supportive information functions". Luciw, Exhibit 1003, col. 4, lines 14-18 (pen-based computer system); col. 2, lines 16-19 (implicit or explicit assistance).

The pen-based computer system has a database that can be queried. Id., col. 8, lines 31-34. Luciw describes "implicit" assistance, wherein a user has used a smart field to enter a word used for look up in the database or has otherwise similarly triggered a database lookup, and "explicit" assistance, wherein the user explicitly invokes assistance from the device as by using pen 38 of Fig. 2. See *Id.* at col. 8, lines 11-62.

The logical processes used by the Luciw device for providing implicit and explicit assistance are shown in Fig. 3 of Luciw. Id., col. 8, lines 2-6. A review of Fig. 3 shows that the database is queried in step 106 if it is determined in step 104 that there is an implicit assist. On the other hand, if in step 104 it is determined that there is not an implicit assist, and if further it is determined that there is an explicit assist, there is no database query, because the only database query indicated is in step 106, exclusively where there is an implicit assist.

As an example of implicit assist, Luciw provides Figs. 4b, 4c, 5, 6a and 6b, which describe use of a "smart field". Id., col 10, line 23 et seq. (beginning discussion of smart fields in connection with Fig. 4b). According to Luciw, "[a] smart field is considered to be a predefined region on screen 52 of computer system 10 shown in FIG. 2, or a predefined region within a window which appears on screen 52". *Id.* col. 8, lines 16-19. Fig. 4b is reproduced below.

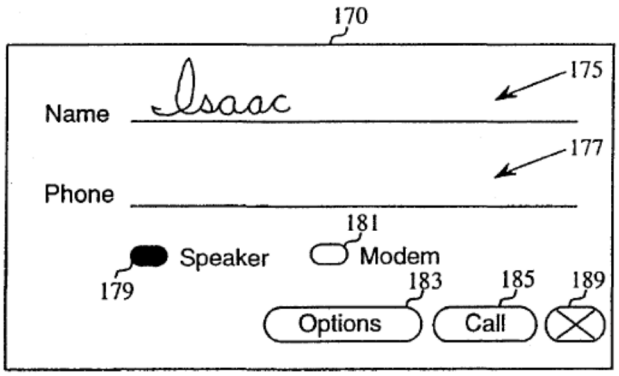


Figure 4b

According to Luciw, Fig. 4b "shows a phone slip window 170 with a smart name field 175 which has for example been evoked by either highlighting the verb 'call' or by simply writing the word on the display surface either before or after establishment of window 170." *Id.*, col 10, lines 24-28. Operation of the phone slip window is explained in the lines thereafter in Luciw:

> Once the particular window 170 is presented to the user, the name ISAAC can be handwritten into the particular smart field 175. The assistance process recognizes the handwritten name "Isaac," and either continues operation as suggested at step 106 in FIG. 3 directly, or concurrently displays the recognized name in formal font form, as suggested in FIG. 4c, in the same position of the smart field, where formerly the handwritten name "Isaac" had been established. As will readily be recognized, window 170 in FIG. 4b may contain several smart fields, in this case for example definable for either the "name" field 175 or a "phone" field shown at step 177.

Id., col. 10, lines 27-39.

Because the user of the Luciw device uses the smart field to specify the field for which a database search is desired—a name in the name field 175 or a phone number in the phone field 177—the Luciw device can then use the entered item to search in the database for an item that has the same value for a corresponding attribute. *Id.*, col. 10, line 51 to col. 12, line 11.

F. Overview of Nielsen [5,963,964]

U.S. patent 5,963,964 to Nielsen (herein, "Nielsen;" Petitioners' Exhibit 1009) issued on October 5, 1999 from an application filed April 5, 1996, and therefore qualifies as a reference under 35 U.S.C. §102(e), at best.

Nielsen discloses "methods, systems, and computer program products" by which a visitor to a web site may create and update, in the visitor's web browser, a bookmark for that website. The creation of a bookmark as disclosed by Nielsen takes place in response to an action by the user of the browse within the browser itself ("In order to save the current web page as a new bookmark, the user selects the 'Add Bookmarks' command from the command section 402." – col. 1, lines 56-59). The updating of a bookmark, as described by Nielsen, is performed automatically by the browser (see the flowchart in Fig. 17 and the related text beginning at col. 7, line 37).

Nielsen does not teach that creation or updating of such a bookmark occurs in response to the visitor merely selecting a link (URL) to the web page.

V. SINCE THE PRIOR ART DOES NOT ANTICIPATE OR RENDER ANY CLAIM OBVIOUS, NO *INTER PARTES* REVIEW SHOULD BE INITIATED

A. Overview of Reasons for Denying *Inter Partes* Review

The references cited in the present Petition fail to anticipate any claim under 35 U.S.C. §102(a) or 35 U.S.C. §102(e) as alleged, and fail to render any claim obvious under 35 U.S.C. §103(a) as alleged. The Petitioners' Ground 1 relies on a combination of two articles, LiveDoc and Drop Zones, as a basis for arguing anticipation. This combination is improper, since anticipation requires that all claim elements be within a single reference. Therefore, the Petitioners fail to establish anticipation of any of the independent claims and the dependent claims based on this combination. {**Response IV.B**}

Many of the claims cited by the Petitioners for review in the **petition require "marking" "first information" "without user intervention**."

In Grounds 1, 2 and 3, the Petitioners cite "LiveDoc/Drop Zones" for this limitation. However, the Petitioners are incorrect because LiveDoc and Drop Zones expressly require user intervention (holding down a function key) in order to mark detected structures ("Holding down a function key places the document in "LiveDoc mode" and presents the highlighted structures; releasing the function key returns the document to normal"). {**Response IV.C**}

In Grounds 5 and 6, the Petitioners cite Miller for this limitation. However, the Petitioners are incorrect because Miller expressly requires user intervention (a "request for display of structures" – Fig. 8 box 860) in order to mark detected structures. {**Response IV.M**}

Some claims are more specific about the marking by requiring that **the marking is to alert the user that the first information can – after the marking - be utilized in a second application program.**

In Ground 4, the Petitioners cite Domini for this limitation. However, the Petitioners are incorrect because Domini does not mark first information to alert the user that the first information can be utilized in a second application program. Rather, Domini marks text to indicate that the text has already been used in a search of a dictionary, and does not teach that the text (first information) may yet be used in a second application program. {**Response IV.J; Response IV.K**}

In Ground 7, the Petitioners cite Luciw for this limitation. However, the Petitioners are incorrect because Luciw does not mark first information to alert the user that the first information can be utilized in a second application program. Rather, Luciw merely recognizes hand-written user input (e.g., a name) and converts it to, and displays it in, a computer font. Luciw's display of that user input "as a formal font" does not alert the user that the first information can – after

the marking - be utilized in a second application program, as required by the claims. {**Response IV.N; Response IV.O; Response IV.P**}

All of the claims cited by the Petitioners for review in the petition require a step of or means **for "responding to a user selection by performing an operation related to a second information, the second information associated with all or part of the first information from the second application program**." The Patent Owner notes that this limitation requires not only a "second application program," but also requires "second information associated with all or part of the first information from the second application program" and an "operation related to a second information," which operation is performed in response "to a user selection." The Patent Owner also notes that the term "associated" requires an association that exits prior to the "operation" performed in response "to a user selection" (i.e., "associated" requires a pre-existing association).

In Grounds 1, 2, and 3, the Petitioners cite "LiveDoc/Drop Zones" for this limitation. However, the Petitioners are incorrect because LiveDoc and Drop Zones fail to disclose performing an operation related to a second information and/or fail to disclose such a performance in response to a user selection {**Response IV.D; Response IV.E; Response IV.F; Response IV.G**}

In Ground 4, the Petitioners cite Domini for this limitation. However, the Petitioners are incorrect because Domini is a spell checker that operates within a

first application program, and thus fails to teach a second application program, and consequently fails to teach a second information associated with the first information from a second application program {**Response IV.H; Response IV.I**}

In Grounds 5 and 6, the Petitioners cite Miller for this limitation. However, the Petitioners are incorrect because Miller fails to teach second information related to the first information from the second application. Rather, Miller offers a user the option of adding information (e.g., first information) to, e.g., a phone book or address book, but that does not require or suggest that the address book or telephone book already have some information (second information) that is related to the information being added {**Response IV.L}**

In Ground 7, the Petitioners cite Luciw for this limitation. However, The Petitioners are incorrect because Luciw fails to disclose a second application program, and fails to disclose second information from a second application program. Luciw is directed to a pen-based computer system that includes a notepad application as "an application program running under the operating system…" However, Luciw does not disclose any interaction between the notepad application program and a second application program. {**Response IV.N**}

B. Because Petitioners rely on a combination of two articles, LiveDoc and Drop Zones, as a basis for arguing anticipation, Petitioners fail to establish anticipation of any of the independent claims and the dependent claims under Ground 1.

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987).

In many of its proffered rejections, Petitioners' improperly treat LiveDoc and Drop Zones as a single reference. For example, in Ground 1, the Petitioners argue that LiveDoc and Drop Zones, reproduced together in Petitioners' Exhibit 1005 and called by Petitioner "LiveDoc/Drop Zones", anticipate all of the independent claims, and several of the dependent claims at issue in the present Petition [IPR2014-00206], under 35 U.S.C. §102(a).

However, LiveDoc and Drop Zones are not a single reference, as required for a rejection based on anticipation. These are two different articles included among dozens of different articles in the SIGCHI Bulletin. (Ex. 1005, p. 1) LiveDoc is an article published beginning at page 51 of the SIGCHI Bulletin, Volume 30, No. 2 entitled "From Documents to Objects: An Overview of LiveDoc." In contrast, Drop Zones is an article published beginning at page 59 of that SIGCHI Bulletin entitled "Drop Zones: An Extension to LiveDoc." In an

Information Disclosure Statement pursuant to 37 C.F.R. 1.98, each would be separately listed.

Indeed, LiveDoc and Drop Zones are listed as separate articles in the '854 Patent itself as the next-to-last prior art reference and the prior art reference immediately preceding the next-to-last reference, in the second column of page 3.

Because LiveDoc and Drop Zones are separate documents, their combination cannot be a basis for anticipation, and therefore Ground 1 fails to establish anticipation of independent claims 19, 25 and 31.

C. Because LiveDoc requires the user to press and hold a function key in order to see identified structures, LiveDoc fails to disclose or suggest the limitation of "marking without user intervention the first information", and therefore Ground 1 fails to establish anticipation, and Ground 2 fails to establish a *prima facie* case for obviousness.

Independent claims 19, 25 and 31 require "marking" the "first information" "without user intervention" "to alert the user that the first information can be utilized in a second application program." When the claims dependent on these independent claims are considered, then all of claims 19-35 include this limitation. The LiveDoc and Drop Zones references relied upon by the Petitioner fail to disclose or suggest this claim limitation.

The Petitioners allege that LiveDoc "runs in the background" and "highlights information in the document that can be used to perform a related

action." (Petition, page 16). The Petitioners thus equate highlighting information in LiveDoc with marking information in the claims.

The Petitioners specifically allege that "the first information is marked without user intervention to alert the user that the first information can be utilized in a second application program" (Petition at page 17).

The Petitioners are incorrect because LiveDoc does not highlight (mark) the first information without user intervention.

In support of their allegation, the Petitioners selectively quote LiveDoc in a way that makes LiveDoc appear to say the exact opposite of what LiveDoc actually says. The passage quoted by the Petitioners is copied below. The double-underlined text was omitted by the Petitioners in the Petition:

> "In LiveDoc, the structure detection process is run in the background on the visible document's text, whenever that document is presented or updated. The results of LiveDoc's analysis are then presented by visually highlighting the discovered structures with a patch of color around the structure. Holding down a function key places the document in "LiveDoc mode" and presents the highlighted structures; releasing the function key returns the document to normal. Pointing at a highlight and pressing the mouse button then displays the menu of actions that can be applied to the structure, as shown in Fig 2."

As such, the omission of the underlined sentence changes the meaning of the paragraph, as compared to the Petitioners' presentation of the paragraph. Whereas

the Petitioners allege that the paragraph teaches that "the first information is marked without user intervention," that allegation is based on the Petitioners' misleading presentation of LiveDoc.

In reality, as revealed by the full quote above, LiveDoc requires user intervention (i.e., holding down a function key) after the structures have been detected to show the user the structures.

For at least this reason, LiveDoc fails to teach all of the limitations of the claims, and the Petitioners have failed to establish a case for anticipation, and fails to establish a *prima facie* case for obviousness based on LiveDoc.

D. Because LiveDoc fails to disclose performing an operation related to second information, the second information associated with the first information from the second application program, Ground 2 fails to establish a prima facie case for obviousness for claims 23, 29, 35, 61, 77 and 83.

The Patent Owner notes that, as part of Ground 2, the Petitioners alleged that some dependent claims are obvious over LiveDoc and Drop Zones. Dependent claims 23, 29, 35, 61, 77 and 83 require "wherein the additional data is entered by a user." Each of those claims depends from a claim that requires that "the operation performed" (in response to a user selection) "is entering additional data into a database."

For these claims, the Petitioners present two allegations.

(1) <u>Bookmarks</u>

The Petitioners allege that LiveDoc teaches "adding a bookmark ('additional data') in a web browser based on a URL in a document. (See, e.g., Fig. 2 of LiveDoc)" and then allege that "It was well known that the names of bookmarks can be edited based on a user's personal preference for names of bookmarks." [Petition at page 27]. Here, the Patent Owner notes that the "bookmark" is the "first information" (i.e., the URL found in a document).

However, the Petitioners neglect to mention that the "operation" in the claims must be "related to a second information, the second information associated with the first information from the second application program."

The actions alleged by Petitioners fail to meet this limitation.

The editing of a name of a bookmark by a user "based on a user's personal preference for names of bookmarks" is not "related to a second information, the second information associated with the first information from the second application program" because the revised name preferred by the user comes from the user, not a second application program.

Furthermore, the Petitioners' "bookmark" example fails to meet this claim limitation because, in adding or editing a bookmark (first information) there is no preexisting association (i.e., an association that existed between fist information and second information) before the "selection" to which the alleged operations

must be "in response." Indeed, the Petitioners even fail to identify a "second information."

(2) Adding data to a database

The Petitioners also argue that "It would also have been obvious to one of ordinary skill in the art that the user would enter additional information into an address book when a detected name is added to the address book as disclosed in Drop Zones." [Petition at page 27].

However, the Petitioners again neglect to mention that the "operation" in the claims must be "related to a second information, the second information associated with the first information from the second application program."

Adding information to an address book does not require or suggest that the address book already have some information (second information) that is associated with the first information, as required by the claims.

E. Because LiveDoc fails to disclose performing an operation ("identifying") related to second information, the second information associated with the first information from the second application program, Ground 2 fails to establish a prima facie case for obviousness for claims 70 and 71.

Dependent claims 70 and 71 each depend from claim 57, and require that (at least part of) the identifying of "first information" "occurs after the user selection."

Significantly, these limitations require a temporal relationship – specifically that (at least some of) the "identifying" first information occurs "after the user

selection." Stated alternatively, the "user selection" occurs before (at least some of) the "identifying" first information.

The Petitioners fail to account for this temporal relationship.

The Petitioners' entire argument regarding claims 70 and 71 is as follows:

> "Claims 70 and 71 are directed to the identifying (or at least part of the identifying) occurring after the user selection. These claims are obvious in view of LiveDoc/Drop Zones. (Menascé Decl. ¶ 119.) In LiveDoc, the detection process runs in the background, but structures are not highlighted until a function key is held down. It would have been obvious that the detection process could also be run in response to holding the function key. *(See, e.g.,* LiveDoc at 55 ("Holding down a function key places the document in `LiveDoc Mode' and presents the highlighted structures."); Menascé Decl. ¶ 119.) Although such implementation would be slower than running the process in the background and thus not as desirable, it would still have been obvious from a technical perspective, as the authors were aware of such implementation but chose to implement the more desirable one. *(See, e.g.,* LiveDoc at 56 ("LiveDoc works quietly in the background and displays the results of its analysis on demand, rather than performing the analysis on demand.")) (Menascé Dec1.1 119.)"

The Petitioners' argument focuses on the fact that LiveDoc "runs in the background" and that the "structures are not highlighted until a function key is held down." The Petitioners then allege (without any support, other than the summary

conclusions of their expert) that "would have been obvious that the detection process could also be run in response to holding the function key." The Petitioners then offer an opinion as to the desirability of running LiveDoc "in the background."

However, the Petitioners' argument fails to mention "user selection," and fails to address the temporal limitations (i.e., that at least some of the "identifying" of first information occurs "after the user selection.") of claims 70 and 71.

In summary, nothing in the Petitioners' argument support their conclusion that LiveDoc or Drop Zones discloses or suggests that (at least some of) the "identifying" occurs "after the user selection."

For at least these reasons, the Petitioners Ground 2 fails to establish a *prima facie* case of obviousness for claims 61, 70, 71, 77 and 85.

F. Because LiveDoc fails to disclose the cause-and effect relationship between responding to a user selection and initializing the second application program, and also fails to disclose the cause-and effect relationship between responding to a user selection and displaying the second information, Ground 1 fails to establish anticipation for claims 85, 96 and 99, and fails to establish a *prima facie* case of obviousness for claim 64.

Independent claims 85, 96 and 99, as well as dependent claim 64, require "responding to a user selection by performing an operation related to a second information, the second information associated with all or part of the first information from the second application program." More particularly, these claims further require that the act of (or means for) responding include:

initializing the second application program;

searching, using the second application program, for the second information associated with the first information; and if said second information exists,

retrieving and displaying the second information.

The underlining is by the Patent Owner, for emphasis.

The acts of "initializing the second application program" and "displaying the second information" both occur in as part of "responding to a user selection." In other words, there is a cause-and-effect relationship between the "user selection" and the "initializing" and "displaying." No such cause-and-effect is disclosed in LiveDoc or Drop Zones, as discussed below.

For at least this reason, the LiveDoc and Drop Zones references relied upon by the Petitioner fail to disclose or suggest at least the "initializing" and "displaying" in response to a "user selection" as required by claims 64, 86,96 and 99.

(1) LiveDoc and Drop Zones fail to disclose the cause-and-effect relationship of "initializing the second application program" in response to a user selection, and for this additional reason Ground 1 fails to establish anticipation of claims 85, 96 and 99, and Ground 2 fails to establish a *prima facie* case of obviousness for claim 64.

The Petitioners fail even to argue that the step of "initializing the second application program" occurs in response to the "user selection," and for at least

that additional reason the Petitioners' Ground 1 fails to establish anticipation of claims 85, 96 and 99 by LiveDoc or Drop Zones.

For the step of "initializing the second application" as part of "responding to the user selection," the Petitioners do not present any new analysis. Rather, the Petitioners simply state "*See* claim 64b" (Petition at page 25). The element of claim 64, which the Petitioners refer to as "64b," requires only "initializing the second application program," and when read in conjunction with claims 63 and 57 from which claim 64 depends, it can be understood that "initializing the second application" in claim 64 is similar to the "initializing" requirement of claims 85, 96 and 99.

However, the Petitioners fail to account for the requirement that the "initializing" occurs in response to the user selection; in other words, the Petitioners fail to account for the cause-and-effect relationship between the user selection and the step of "initializing the second application program." Rather, the Petitioners argument states, in its entirety: "The second application program must necessarily be initialized in order to run."

Because the Petitioners fail even to argue that the step of "initializing the second application program" occurs in response to the "user selection," and for at least that additional reason the Petitioners' Ground 1 fails to establish anticipation

of claims 85, 96 and 99 by LiveDoc or Drop Zones, and Ground 2 fails to establish a *prima facie* case of obviousness for claim 64.

(2) LiveDoc and Drop Zones fail to disclose the cause-and-effect relationship of "displaying the second information" in response to a user selection, and for this additional reason Ground 1 fails to establish anticipation of claims 85, 96 and 99, and Ground 2 fails to establish a *prima facie* case of obviousness for claim 64.

The Petitioners fail even to argue that the step of "displaying the second information" occurs at all, and for at least that additional reason the Petitioners' Ground 1 fails to establish anticipation of claims 85, 96 and 99 by LiveDoc or Drop Zones.

For the step of "displaying the second information," the Petitioners do not present any new analysis. Rather, the Petitioners simply state "*See* claim 64d" (Petition at page 25). The element of claim 64, which the Petitioners refer to as "64d" requires only "retrieving the second information," and does not require an act of "displaying the second information."

The Petition fails even to allege that LiveDoc or Drop Zones discloses the "displaying the second information" requirement, and makes no attempt to argue that it occurs in response to the "user selection." In fact, the Petitioners fail even to argue that the step of "displaying the second information" occurs at all. For at least these additional reasons, the Petitioners' Ground 1 fails to establish

anticipation of claims 85, 96 and 99 by LiveDoc or Drop Zones, and Ground 2 fails to establish a *prima facie* case of obviousness for claim 64.

G. Because LiveDoc and Drop Zones in view of Nielsen fails to disclose performing, in response to a user selection, an action related to second information from the second application program, and fails to resolve the shortcomings of LiveDoc and Drop Zones, Ground 3 fails to establish a *prima facie* case for obviousness.

The Petitioners alleged that some dependent claims are obvious under 35 U.S.C. 103(a) over LiveDoc and Drop Zones in view of Nielsen [Exhibit 1009]. Petition, page 29. The Petitioners cite Nielsen as disclosing "creating bookmarks in Netscape Navigator that include URLs and a corresponding user-entered name for the bookmark (Fig. 12 at 1210; 2:9-31)." *Id.* The Petitioners allege that creating bookmarks "enters the bookmark data" into a "database," and imply that entering bookmarks into a database is "performing an operation." *Id.*

However, the Petitioners neglect to mention that the "operation" in the claims must (i) be a response to "a user selection" and must (ii) be "related to a second information, the second information associated with the first information from the second application program."

The creation or update of a bookmark by a user is not an action in response to a "user selection." Rather, the creation of a bookmark is an optional action that, if it occurs at all, takes place in the browser after the user has selected the URL from the document in the first application (see, e.g., Fig. 2 of LiveDoc, where the

URL has been highlighted). Similarly, as explained in Nielsen, the updating of a bookmark is performed automatically by the browser (see, e.g., Nielsen Fig. 17 and text at col. 7, line 37 – col. 8, line 22).

In addition, the URL, as illustrated in Fig. 2 of LiveDoc, is not "second information" since "second information" must be "from the second application program," and the URL comes from the document being analyzed and not from a "second application program." In short, creating a bookmark is not an operation "related to a second information, the second information associated with the first information from the second application program."

The Petitioners further cite Nielsen as disclosing that "the name of the bookmark added to the database can be changed." *Id*., p. 30. The Petitioners make no argument as to how this observation applies to the claims (see the first full paragraph on page 30 of the Petition, in which Petitioners merely allege that Nielsen discloses this feature). In any event, because the new name comes directly from the mind of the user performing the editing, editing a bookmark name does not involve "second information associated with the first information from the second application program."

Furthermore, the Petitioners fail to allege that the Nielsen reference addresses or resolves any of the shortcomings of LiveDoc as discussed above.

For at least these reasons, Ground 3 fails to establish a prima facie case for obviousness based on LiveDoc and Drop Zones in view of Nielsen

H. Because Domini fails to disclose second information associated with the first information from a second application program, Ground 4 fails to establish anticipation.

Independent claims19, 25, 31, 57, 73 and 79 require "performing an operation related to a second information, the second information "associated with the first information from the second application program." Similarly, method claim 85, 96 and 99 require, among other things, a step of "performing an operation related to a second information, the second information associated with all or part of the first information from the second application program." When the claims dependent on these independent claims are considered, then all of claims 19-35, 57-85, 96 and 99 include this limitation. The Domini reference relied upon by the Petitioner fails to disclose or suggest this claim limitation.

Because spell checker modules, such as those described in Domini, present words based upon a guess of what the user intended to type, rather than information with a pre-existing relationship (or ", Domini fails to teach a second information associated with the first information from a second application program, and therefore the Petitioners have failed to establish a case for anticipation based on Domini.

Spell checker modules, such as those of Domini, assess whether a typed word in document matches an entry in a dictionary. If there is no matching word in the dictionary, then spell checkers assume that the word in misspelled, and suggest a list of other words that might, or might not, be what the user intended.

For example, a user may type a set of characters such as "reid" and the spell checker will mark the characters because the specified set of characters in the specified order do not appear within the spell checker's dictionary. The spell checker then makes a guess at the intent of the user, by displaying possible choices, such as "red, read, reader etc." The search of the database is not a search for information that is "related to" the entered information. These guesses are not "related information" as there is no pre-existing "relationship" between the typed characters and the retrieved words within the dictionary.

The spell checker will allow a user to select from the entries, and upon the selection of an entry a relationship is established. This "relationship" occurs too late to meet the limitation of the selected word being "associated with the first information" since the spell checker's operation is based upon a guess and not based upon any pre-existing relationship between the typed information and the suggested replacement terms.

In fact, there may be no relationship at all between the words that the spell checker finds and the intent of the user. As in the example presented, the user may

have intended to type "reid" e.g., as a name or a word in a different language, or may have intended to type the word "ride", which was not presented to the user. Thus, in this case, no relationship exists between the typed characters and the items that are identified in the search of the dictionary.

Spell checkers never search for related information, wherein there is an already-established relationship between the entered text (first information) and the related information (second information) found within a database. Consequently, Ground 4 fails to establish anticipation of independent claims 19, 57 and 85.

I. Because Domini fails to disclose a second application program, Ground 4 fails to establish anticipation.

Independent claims 19, 25, 31, 57, 73 and 79 require performing an operation related to a second information, the second information associated with the first information from the second application program." Similarly, method claim 85, 96 and 99 require, among other things, a step of "performing an operation related to a second information, the second information associated with all or part of the first information from the second application program." When the claims dependent on these independent claims are considered, then all of claims 19-35, 57-85, 96 and 99 include this limitation. The Domini reference relied upon by the Petitioner fails to disclose or suggest this claim limitation.

Because the Domini spell checker is a module operating inside of the first application, the reference fails to disclose a second application and fails to disclose

second information from a second application, and therefore the petitioners fail to establish a anticipation based on Domini.

Embodiments of the '854 patent include the retrieval of information (e.g., a name or address) from one application program (e.g. Microsoft Outlook) while a user works simultaneously in another application program (e.g., Microsoft Word). See col. 2 lines 14-23. The independent claims all require "a first application program" and "a second application program." As would be understood by one of ordinary skill in the art, an "application program" as required by the independent claims is directed to an independently executable program, as opposed to a utility module, such as a spell checker, that provides a feature to another program. Examples of applications are a word processing program, such as Microsoft Word, a spread sheet program, such as Microsoft Excel, and a contact database program, such as Microsoft Outlook.

A person of ordinary skill in the art would also understand that an "application" will utilize utilities, such as spell-checker module, for example.

The '854 patent solves the problem of communicating information between two independently executable application programs. The '854 contemplates the interaction between executable application programs wherein a user may operate within a first application program without the need for exiting and initiating a second application program to find information that is related to information

entered into a document within the first application program. Examples of application programs as specified within the specification of the '854 patent encompass only separately executable computer programs: a word processing program, such as Microsoft Word, a spread sheet program, such as Microsoft Excel, and a contact database program, such as Microsoft Outlook. See col. 1, lines 39-42 and col. 9 lines 64 – col. 10, line 10.

The Domini reference, in contrast, discloses working with suggested spelling and grammar corrections while a user works within a single program—i.e. a word processor. See, e.g., Abstract, col. 3 lines 1-12 etc. Because Domini fails to teach a second application program, Domini fails to teach or suggest the functionality of the present patent as embodied in independent claims, each of which requires a first application program and a second application program. As a result, Domini fails to anticipate or make obvious the independent claims listed above.

Also, as a practical matter, the dictionary that is searched in Domini is not a "second application program" because a dictionary is merely a listing of properly-spelled words, and therefore not a "program" in any sense. See, e.g., the definition of "dictionary" from the Heritage College dictionary 3rd edition 1997 in Patentee's Exhibit 2001.

Additionally, since Domini fails to disclose "a second application" as would be understood by one of ordinary skill in the art, Domini also fails to disclose

"second information" "associated with the first information from a second application program." Thus, Domini fails to teach the "associated information" limitation of the independent claims in addition to failing to teach a second application program. Consequently, Ground 4 fails to establish anticipation of independent claims19, 25, 31, 57, 73 and 79.

J. Because Domini fails to disclose marking the first information to alert the user that the first information can be utilized in a second application program, for this additional reason, Ground 4 fails to establish anticipation.

Independent claims 19, 25 and 31 require "marking" the "first information" "without user intervention" "to alert the user that the first information can be utilized in a second application program." When the claims dependent on these independent claims are considered, then all of claims 19-35 include this limitation. The Domini reference relied upon by the Petitioner fails to disclose or suggest this claim limitation.

Rather, because Domini is a spell checker, the only use (operation) of the words of the text (first information) in Domini occurs before the word is marked in Domini's "red, bold typeface" or flagged with a blinking cursor (as described by Domini at column 11, lines 46-49). Thus the marking in Domini is an indication that the word has already been used, and not that he marked word "can be utilized in a second application."

More specifically, each and every word of the text in Domini is used as input to a search of a dictionary. Indeed, this order of events (search first, then mark) is critical to Domini, because such a search must occur before a word can be identified as possibly misspelled, and therefore before such a word can be marked. Without such a prior search, it would be impossible for Domini to know whether a word is possibly misspelled.

Consequently, by the time Domini marks a word (first information), that word has already been used, and the marking does not indicate that the word "can be utilized in a second application program."

For at least this reason, Ground 4 fails to establish anticipation of independent claims 19, 25 and 31.

K. Because Domini's actions subsequent to marking fail to correspond to the requirements of the claims, for this additional reason, Ground 4 fails to establish anticipation.

Domini allows a user to take actions after a word has been marked (i.e., in Domini, an unrecognized word is displayed in red, bold typeface), but those actions do not teach the limitations of the claims.

(1) Adding a Word to a Dictionary is not an Operation Relating to Second Information

Domini allows a user to add an unrecognized word to a dictionary. However, such an action is not an operation relating to a second information, since

it is the first information (i.e., the unrecognized word) that is added to the dictionary.

Domini fails to teach or suggest that adding information to a spell-checker's dictionary requires that the dictionary already have some information (second information) that is related to the information being added. Nor does Domini teach that the user adds any such association between the word added and another word.

Consequently, the act of adding an unrecognized word to a dictionary is not an operation relating to second information. For at least that reason, the act of adding an unrecognized word to a dictionary fails to teach the limitations of the claims.

(2) Replacing a Word in a Document is not an Operation Relating to Second Information, which Second Information as Associated with the First Information from a Second Application Program

Domini allows a user to replace an unrecognized word in a document with another term. However, as described above, Domini presents potential replacement words based upon a guess of what the user intended to type. If one of the guesses is correct, then the user may select the correct guess and replace the unrecognized word with the correct guess. Prior to the user's selection, however, there is no association between the unrecognized term and the replacement term, and a such the "second information" (replacement term) is not "associated" with the "first information" (potentially misspelled term), since the term "associated" requires

that the "association" exists before the potential replacement terms and presented to the user, and therefore before the user makes the "selection." In other words, there is no pre-existing association the unrecognized term and the replacement term, as required by the ordinary meaning of the word "association."

Consequently, the act of replacing a word in a document as described by Domini is not an operation relating to second information, which second information is associated with the first information from a second application program, at least because there is no pre-exiting association.

For at least the foregoing additional reasons, Ground 4 fails to establish anticipation.

L. Because Miller fails to disclose second information related to the first information from the second application, Ground 5 fails to establish anticipation by Miller, and Ground 6 fails to make a *prima facie* case for obviousness.

Independent claims19, 25, 31, 57, 73 and 79 require performing an operation related to a second information, the second information associated with the first information from the second application program." Similarly, method claim 85, 96 and 99 require, among other things, a step of "performing an operation related to a second information, the second information associated with all or part of the first information from the second application program." When the claims dependent on these independent claims are considered, then all of claims 19-35, 57-85, 96 and 99

include this limitation. The Miller reference relied upon by the Petitioners fails to disclose or suggest this claim limitation.

As noted by the Petitioners in the present Petition, Miller teaches that a user may add some information from a detected structure into, e.g., a phone book or address book.

However, the Petitioners then allege – and without providing any support for the allegation – that the information added joins "other address information in the address book, such as a name. . . ." [Petition at page 41].

Miller fails to teach or suggest that information from a "detected structure" is correlated or related to any "second information" already in, e.g., an address book. Rather, Miller merely offers the option to put the information into a designated storage [e.g., from Miller's Fig. 6: "Put in electronic telephone book"]. Adding information to an address book or an electronic telephone book does not require or suggest that the address book or telephone book already have some information (second information) that is related to the information being added.

Indeed, Miller itself recognizes that the user may need to input additional information at column 5, lines 38-50. The Petitioners recognize this feature at page 41 by citing Miller's Fig. 7 and the text at column 5, lines 47-50 – "Action processor 250 locates and opens the electronic telephone book, places the telephone number in the appropriate field and allows the user to input any

additional information into the file" (underlining added here by the Patent Owner, for emphasis). In any case, any additional information added by the user does not have a pre-existing association with first information, and is not, as required by the claims, "from" the second application.

In their final observation in the claim chart for element 19d, the Petitioners provide some observations about linking "actions to the detected structure" and retrieving "the sequence of operations that constitute the selection action." [Petition at page 41] The Petitioners do not correlate these observations to any argument. However, if Petitioners intend to allege that these observations teach an "operation related to a second information," the Petitioners are incorrect.

The Patent Owner respectfully observes that the claims require that the "second information" is "associated with the first information from the second program. The Petitioners observations fail to correlate any "candidate action" or "sequence of operations" to a second application program, or even identify them as an alternative interpretation of "second information". . Indeed, the Petitioners do not identify the source of the "candidate action" or "sequence of operations."

Because Miller does not teach second information related to the first information from the second application, Miller cannot and does not teach "an operation related to a second information, the second information associated with the first information from the second application program" and as such Miller does

not and cannot teach "responding to a user selection by performing an operation related to a second information, the second information associated with the first information from the second application program" as required by the claims.

For at least these reasons, the Petitioners' Ground 5 fails to establish anticipation by Miller, and Petitioners' Ground 6 fails to establish a *prima facie* case for obviousness in view of Miller.

M. Because Miller fails to disclose marking first information "without user intervention", for this additional reason, Ground 5 fails to establish anticipation by Miller, and Ground 6 fails to establish a *prima facie* case for obviousness.

Independent claims 19, 25 and 31 require "marking" the "first information" "without user intervention." When the claims dependent on these independent claims are considered, then all of claims 19-35 include this limitation. The Miller reference relied upon by the Petitioner fails to disclose or suggest this claim limitation.

The Miller reference relied upon by the Petitioners fails to disclose or suggest this claim limitation.

Rather, as illustrated by Miller's Fig. 8 and Fig. 9, Miller requires user intervention (e.g., "request for display of structures" – Fig. 8 box 860) in order to simply highlight (mark) a detected "structure."

More particularly, in Miller, text from a document (210) is provided to a program (165). The program 165 includes, among other things, an "application

program interface" (230) and a "user interface" (240). Miller explains the function of the "application program interface" (230) and a "user interface" (240) as follows: "After identifying structures and linking actions, application program interface 230 communicates with application 167 to obtain information on the identified structures so that user interface 240 can successfully present and enable selection of the actions." [Miller, column 4, lines 1-11].

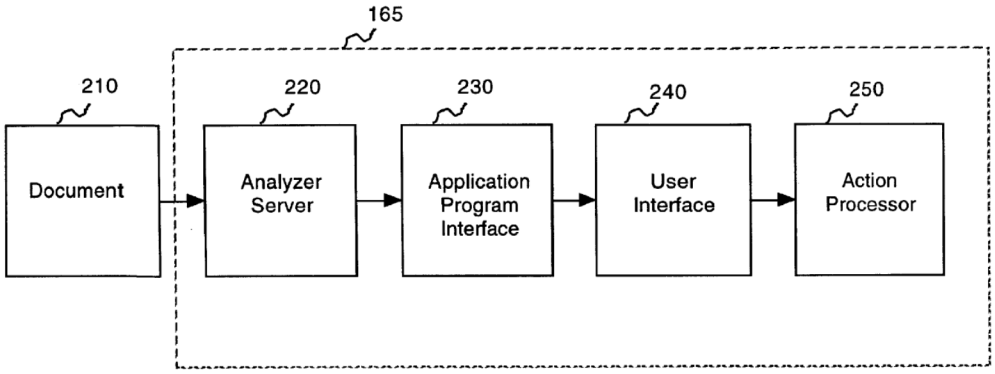


FIG. 2

More particularly, Miller teaches that the interface 240 is to "highlights the detected structures." *Id.* However, Miller's highlighting of the "detected structures" does not occur without user intervention. Miller describes this highlighting function in connection with Fig. 8.

Fig. 8, reproduced below, is a block diagram showing the flow of how Miller recognizes patterns and performs actions. More particularly, after initially detecting patterns and linking actions to the detected structures (flow chart boxes

810-830), Miller loops to determine whether the content has changed (box 840) or whether the region has changed (box 850), and if so loops back to earlier points in the flow.

Otherwise, Miller "continues to block 860" (col. 6, lines 3-4) where it determines whether the user has requested that the structures be displayed. As explained by Miller: "As illustrated by block 860, method 800 loops between blocks 840 and 860 until a request for display of identified structures is received 860" (col. 6, lines 4-6).

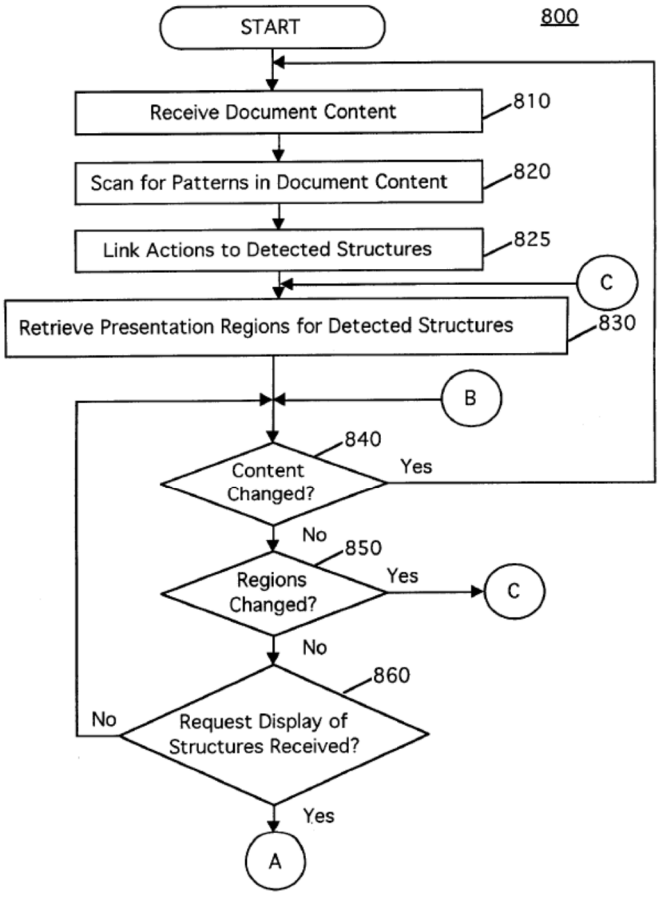


FIG. 8

As illustrated by Miller's Fig. 8, Miller requires user intervention (e.g., "request for display of structures" – box 860) in order to simply highlight (mark) a detected "structure."

For at least these reasons, the Petitioners' Ground 5 fails to establish anticipation by Miller, and Petitioners' Ground 6 fails to establish a *prima facie* case for obviousness in view of Miller.

N. Because Luciw fails to disclose a second application program, and fails to disclose second information from a second application program, Ground 7 fails to establish anticipation.

Independent claims19, 25, 31, 57, 73 and 79 require performing an operation related to a second information, the second information associated with the first information from the second application program." Similarly, method claim 85, 96 and 99 require, among other things, a step of "performing an operation related to a second information, the second information associated with all or part of the first information from the second application program." When the claims dependent on these independent claims are considered, then all of claims 19-35, 57-85, 96 and 99 include this limitation. The Luciw reference relied upon by the Petitioner fails to disclose or suggest this claim limitation.

Because Luciw discloses only a single application for personal information management, Luciw fails to disclose a second application, and for at least that reason the Petitioners fail to make a case for anticipation based on Luciw.

Luciw is directed to a pen-based computer system that provides assistance to the user based on a user's entry within a notepad application. Luciw identifies the notepad application as "an application program running under the operating system…" See col. 6 lines 49-51. Luciw also mentions that the pen based computer system may include other "applications." See col. 6 lines 55-59. However, Luciw does not describe operation of these "other" applications and does

not disclose any interaction between the notepad application program and a second application program. Luciw therefore fails to teach a "second application program," and by extension therefore fails to teach "marking without user intervention the first information to alert the user that the first information can be utilized in a second application program" and fails to teach "responding to a user selection by performing an operation related to a second information" since only the operation of notepad is discussed in Luciw.

Luciw simply describes a personal information manager i.e. the notepad program, which allows one to store and retrieve contact information from a computer database. Luciw only references the term "database" generically and does not refer to the database as an "application program." Luciw refers to the "computer database" as a "frame database system" where a "frame" identifies the formatting of database entries. See Fig. 5 where a frame of type <person> would include attributes such as name, birthday, telephone number etc. See also col. 10 lines 51-60. Thus, Luciw merely discloses the interaction between a single "application program" (notepad) and an associated computer database and fails to teach a "second application program" as required by the claims.

The Petitioners allege that the second application program is the database; however the Luciw disclosure does not refer to the database as a database program

or as an application program but merely references that the database exists and has a format.

The '854 patent and the associated claims are not directed to operation of a single application program and an associated database as disclosed in Luciw, but rather to obtaining associated information from a second application program based upon first information that is entered into a document of a first application program. One of ordinary skill in the art would understand that an "application program" is a self-contained and executable computer program for performing a specific function. Because petitioners fail to show a second application program in Luciw, the petitioners have failed to establish a case for anticipation based upon Luciw for at least this reason.

O. Because Luciw fails to disclose marking first information to alert the user that the first information can be utilized in a second application, for this additional reason, Ground 7 fails to establish anticipation.

Independent claims 19, 25 and 31 require "marking" the "first information" "without user intervention" "to alert the user that the first information can be utilized in a second application program." When the claims dependent on these independent claims are considered, then all of claims 19-35 include this limitation. For at least that reason, the Petitioners fail to make a case for anticipation based on Luciw.

The Petitioners allege that, "once a user enters a first name, the system without user intervention recognizes the name and displays it as a formal font (i.e., marks first information) to alert the user that the name can be used for an implicit assist action – for example, to locate additional information in a database . . . ." (Petition, page 51).

This is incorrect because Luciw's system does not display the name in a "formal font" so as to alert the user that the name can be used in a second application program. Indeed, Luciw's display of a name "as a formal font" has nothing to do with whether that name can be utilized in a second application, as required by the claims.

Rather, Luciw's transformation of a handwritten name to a "formal font" (e.g., see Fig. 4b and Fig. 4c) occurs merely because Luciw's device has "recognized the handwritten name" (column 10, lines 29-30). Indeed, any action taken by Luciw with respect to that handwritten name is presented as an alternative action, distinct from the display of the name in a formal font, as described by Luciw at column 10, lines 29-35 (emphasis added by the Patent Owner):

> "The assistance process recognizes the handwritten name "Isaac," and **either** continues operation as suggested at step 106 in FIG. 3 directly, **or** concurrently displays the recognized name in formal font

form, as suggested in FIG. *4c,* in the same position of the smart field, where formerly the handwritten name "Isaac" had been established."

Thus, the transformation of the handwritten name and the display of the name in a "formal font" does not indicate to or alert a user that the name can be used in a second application program, and for at least this reason, Ground 7 fails to establish anticipation.

P. Because Luciw fails to disclose "marking" or "identifying" first information without user intervention, for this additional reason, Ground 7 fails to establish anticipation.

All claims cited by the Petitioners for review in the petition require "marking" or "identifying" the "first information" "without user intervention."

The Petitioners allege that this limitation is disclosed by Luciw's "implicit" assistance – for all claims, *see* Petitioners' allegation for element "[19c]" at page 51 of the Petition.

The Petitioners are incorrect because, as Luciw explains at col. 4, lines 7-41 Luciw's "implicit" assistance inherently requires user action to indicate the "first information" by either (a) entering the information in a certain way (i.e., into a "smart field;" "A particular field is considered smart, because of the specialized capabilities of the smart field to respond with particularized effectiveness and intelligence to user needs, indications, or events registered, for example, by pen 38, within the bounds of the particular smart field " *id.*), or (b) indicating that the

information is for implicit assistance by performing a "predefined allowable event" ("Certain kinds of events on screen 52, for example, such as the writing of a particular indication or word on screen 52 outside of a particular smart field may trigger an implicit assist. In general, implicit assist can be triggered by the happening of any of a number of predefined allowable events." *id.*).

However, by typing into a "smart field" or performing a "predefined allowable event" the user tells the system exactly what the first information is, including its type (e.g., telephone number or name by typing into a particular field). Consequently, there is no need for the system to identify or mark at all, and indeed, Luciw teaches no such identification or marking as required by the claim, "without user intervention."

Consequently, Luciw does not disclose "marking" or "identifying" first information without user intervention, and for at least this additional reason, Ground 7 fails to establish anticipation.

VI. CONCLUSION

For the foregoing reasons, Petitioners have failed to establish a reasonable likelihood of prevailing as to the claims of the '854 Patent as described above, and *inter partes* review of claims 19-35, 57-85, 96, and 99 of U.S. Patent No. 7,496,854 should be denied.

Dated: March 12, 2014

Respectfully submitted,

/Robert M. Asher, #30,445/

Robert M. Asher
Registration No. 30,445
Bruce D. Sunstein
Registration No. 27,234
Thomas J. Tuytschaevers
Registration No. 42,190
Sunstein Kann Murphy & Timbers LLP
125 Summer Street
Boston, MA 02110
Tel: (617) 443-9292
Fax: (617) 443-0004

03324/00503 2038588.1

CERTIFICATE OF SERVICE

It is certified that on March 12, 2014, copies of the Patent Owner Arendi S.A.R.L.'s Preliminary Response Under 35 U.S.C. § 313 and 37 C.F.R. § 42.107 has been served on Petitioners as provided in 37 C.F.R. § 42.6(e) via electronic mail transmission addressed to the persons at the following addresses:

LEAD COUNSEL FOR PETITIONER APPLE DAVID L. FEHRMAN dfehrman@mofo.com Registration No. 28,600 MORRISON & FOERSTER LLP 707 Wilshire Blvd., Suite 6000 Los Angeles, CA 90017-3543 Tel: (213) 892-5200 Fax: (213) 892-5454	BACK-UP COUNSEL FOR PETITIONER APPLE MEHRAN ARJOMAND marjomand@mofo.com Registration No. 48,231 MORRISON & FOERSTER LLP 707 Wilshire Blvd., Suite 6000 Los Angeles, CA 90017-3543 Tel: (213) 892-5200 Fax: (213) 892-5454
LEAD COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC. MATTHEW A. SMITH smith@turnerboyd.com Registration No. 49,003 TURNER BOYD LLP 2570 W. El Camino Real, Suite 380 Mountain View, CA 94040 Tel: (650) 265-6109 Fax: (650) 521-5931	BACK-UP COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC. ZHUANJIA GU gu@turnerboyd.com Registration No. 51,758 TURNER BOYD LLP 2570 W. El Camino Real, Suite 380 Mountain View, CA 94040 Tel: (650) 265-6109 Fax: (650) 521-5931 and kent@turnerboyd.com docketing@turnerboyd.com

Date: March 12, 2014

/Robert M. Asher, #30,445/

Robert M. Asher
Registration No. 30,445
Sunstein Kann Murphy & Timbers LLP
125 Summer Street
Boston, MA 02110
Tel: (617) 443-9292
Fax: (617) 443-0004

Exhibit 6N

Trials@uspto.gov
Tel: 571-272-7822

Paper 9
Entered: June 11, 2014

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC., GOOGLE INC., and MOTOROLA MOBILITY LLC
Petitioner

v.

ARENDI S.A.R.L.
Patent Owner

Case IPR2014-00206
Patent 7,496,854 B2

Before SALLY C. MEDLEY, HOWARD B. BLANKENSHIP, and TREVOR M. JEFFERSON, *Administrative Patent Judges.*

JEFFERSON, *Administrative Patent Judge*.

DECISION
Institution of *Inter Partes* Review
37 C.F.R. § 42.108

I. INTRODUCTION

A. Background

Apple Inc., Google Inc., and Motorola Mobility LLC (collectively, "Petitioner") filed a Petition (Paper 3, "Pet.") to institute an *inter partes* review of claims 19-35, 57-85, 96, and 99 of U.S. Patent No. 7,496,854 B2 (Ex. 1001, "the '854 patent"). Pet 1; *see* 35 U.S.C. § 311. Arendi S.A.R.L. ("Patent Owner") filed a Preliminary Response (Paper 7, "Prelim. Resp.").

The standard for instituting an *inter partes* review is set forth in 35 U.S.C. § 314(a), which provides as follows:

> THRESHOLD.—The Director may not authorize an inter partes review to be instituted unless the Director determines that the information presented in the petition filed under section 311 and any response filed under section 313 shows that there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.

Pursuant to 35 U.S.C. § 314, we authorize an *inter partes* review to be instituted as to claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85, and 96 of the '854 patent.

B. Related Matters

Patent Owner sued Petitioner for infringement of the '854 patent in *Arendi S.A.R.L. v. Apple Inc*., No. 1:12-cv-01596-LPS (D. Del.); *Arendi S.A.R.L. v. Motorola Mobility LLC*, Case No. 1:12-cv-01601-LPS (D. Del.). Pet. 1.

Petitioner also filed a petition for *inter partes* review of claims 1-18, 36-56, 86-95, 97, 98, 100, and 101 of the '854 patent in IPR2014-00207. *Id.* at 3-4.

C. References Relied Upon

Petitioner relies upon the following prior art references:

Reference(s)	Exhibit
James R. Miller & Thomas Bonura, *From Documents to Objects: An Overview of LiveDoc*, SIGCHI BULLETIN, Vol. 30, No. 2, April 1998, pp. 53-58 ("LiveDoc") and James R. Miller & Thomas Bonura, *Drop Zones: An Extension to LiveDoc*, SIGCHI BULLETIN, Vol. 30, No. 2, April 1998, pp. 59-63 ("Drop Zones") (collectively, "LiveDoc/DropZones")	Ex. 1005
U.S. Patent No. 6,085,206 ("Domini")	Ex. 1006
U.S. Patent No. 5,946,647 ("Miller")	Ex. 1007
U.S. Patent No. 5,644,735 ("Luciw")	Ex. 1008
U.S. Patent No. 5,963,964 ("Nielsen")	Ex. 1009

D. The Asserted Grounds

Petitioner contends that the challenged claims are unpatentable based on the following grounds (Pet. 13-14):

Reference[s]	Basis	Claim(s) Challenged
LiveDoc/Drop Zones[1]	§ 102(a)	19-22, 24-28, 30-34, 57-60, 62-69, 72-76, 78-82, 84, 85, 96, and 99
LiveDoc/Drop Zones	§ 103(a)	23, 29, 35, 61, 64-71, 77, and 83

[1] The parties disagree on whether LiveDoc/Drop Zones can be considered a single reference for purposes of anticipation. Pet. 15; Prelim. Resp. 27-28. Because we are not persuaded that the published material, even if considered as a single reference, discloses all limitations of any of the claims that are challenged under § 102 for reasons discussed below, we do not reach the "single reference" issue.

Reference[s]	Basis	Claim(s) Challenged
LiveDoc/Drop Zones and Nielsen	§ 103(a)	22-24, 28-30, 34, 35, 60-62, 76-78, and 82-84
Domini	§ 102(e)	19, 20, 22-26, 28-32, 34, 35, 57, 58, 60-74, 76-80, 82-85, 96, and 99
Miller	§ 102(e)	19, 21-25, 27-31, 33-35, 57, 59-63, 72, 73, 75-79, and 81-84
Miller	§ 103(a)	20, 26, 32, 58, 64-69, 74, 80, 85, 96, and 99
Luciw	§ 102(e)	19-35, 57-71, 73-85, 96, and 99

E. The '854 Patent

The '854 patent, titled "Method, System and Computer Readable Medium for Addressing Handling From a Computer Program," relates to computer implemented processes for providing a computer program, such as a word processing program or spreadsheet program, that is coupled to an information management source, such as a database program or contact management program. Ex. 1001, 1:19-50.

Figures 3 and 4 of the '854 patent are reproduced below.

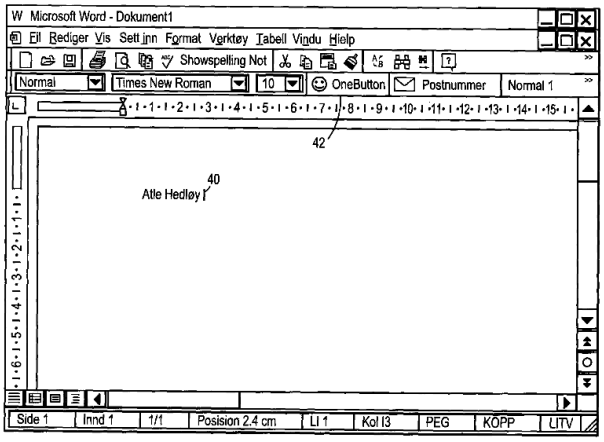


FIG. 3

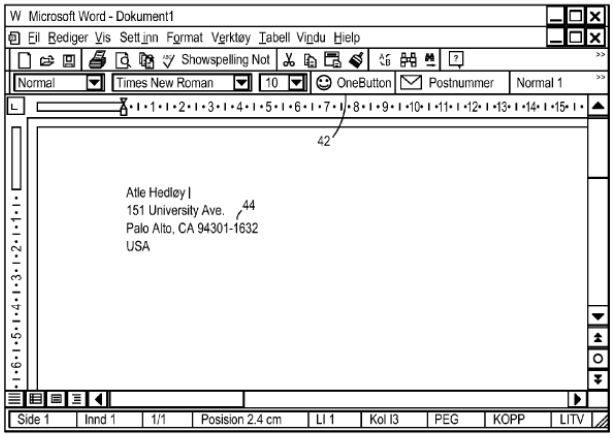


FIG. 4

Figure 3 illustrates the inputting of a name to be searched into a document. Figure 4 illustrates a retrieved address that is inserted into a document. Ex. 1001, 2:51-57. The user types a name into the document. When the user clicks on OneButton 42, the claimed process is launched, retrieving name 40 from the document, searching a database for name 40, and inserting the retrieved address associated with name 40 into the document, as shown in Figure 4. Ex. 1001, 5:60-6:5.

Figure 2 of the ’845 patent, a flow chart illustrating a method for address handling within a computer program, is reproduced below.

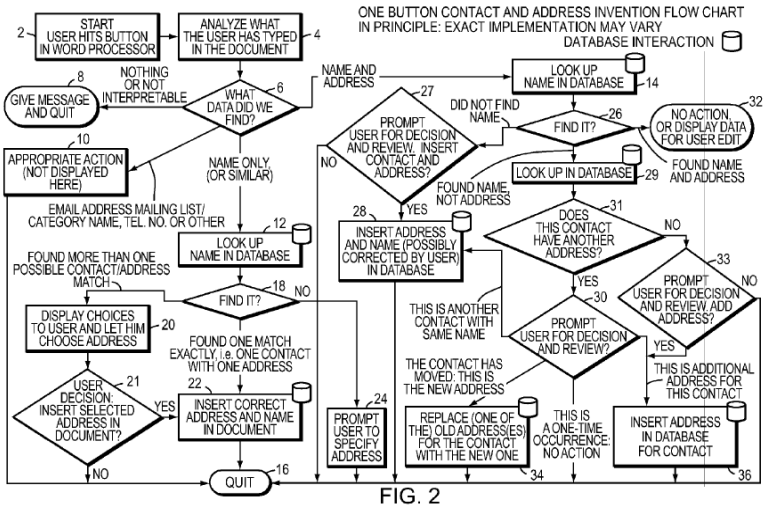


FIG. 2

Figure 2 depicts a flow chart of the address handling process initiated by the user clicking on OneButton 42 of Figure 4. At step 4, text typed by the user in a document is analyzed for contact information. At step 6, if the identified contact information includes a name, a search occurs in the database at step 12. When the database finds a name with more than one possible matching address, the user is prompted for a decision, and that selected information is added to the document at step 22. Ex. 1001, 5:10-22, 6:4-5.

F. Illustrative Claim

Independent claim 19, reproduced below, is illustrative of the claimed subject matter:

> 19. A method for information handling within a document created by a first application program comprising the steps of:
>
> entering a first information in the first application program;
>
> marking without user intervention the first information to alert the user that the first information can be utilized in a second application program; and
>
> responding to a user selection by performing an operation related to a second information, the second information associated with the first information from the second application program.

II. ANALYSIS

A. Claim Construction

We determine the meaning of the claims as the first step of our analysis. The Board interprets claims using the broadest reasonable construction. *See* 37 C.F.R. § 100(b); Office Patent Trial Practice Guide, 77 Fed. Reg. 48756, 48766 (Aug. 14, 2012). Claim terms, generally, are given their ordinary and customary meaning, as would be understood by one of ordinary skill in the art in the context of the entire disclosure. *See In re Translogic Tech., Inc.*, 504 F.3d 1249, 1257 (Fed. Cir. 2007). If an inventor acts as his or her own lexicographer, the definition must be set forth in the specification with reasonable clarity, deliberateness, and precision. *Renishaw PLC v. Marposs Societa' per Azioni*, 158 F.3d 1243, 1249 (Fed. Cir. 1998).

1. "marking ... the first information to alert the user"

The claim term "marking without user intervention the first information to alert the user" appears in independent claims 19, 25, and 31. Petitioner notes that the term "marking" does not appear in the '854 patent specification. Pet. 7. Petitioner argues that the plain meaning of the claim term "is that the first information is detected without user intervention and has some form of marking or highlighting applied to it to draw the user's attention to it." Pet. 7 (citing Declaration of Menasce, Ex. 1002 ¶¶ 49-50). Petitioner asserts that during prosecution the Applicant explained the support for the claimed "marking" by stating that marking could be done in a variety of ways including displaying the text to the user in a separate screen. Pet. 7 (citing Ex. 1004, 30-31); *see* Ex. 1001, 7:1-14 (marking by generating a

screen or dialog window). Thus, Petitioner argues that the marking could be accomplished by presenting the marked information in a separate window.

Patent Owner provides no proposed construction or argument regarding "marking … the first information to alert the user." *See* Prelim. Resp. 7-9. During prosecution of the application that matured into the '854 patent, the then Applicant explained that the claim term "marking" found support in the specification, referencing methods of marking that include generating a separate screen. Ex. 1004, 30-31. The Applicant also provided a dictionary definition of marking as "to pick out or designate something . . . as special in some way." Ex. 1004, 30. For purposes of this institution decision, "marking" encompasses highlighting, designating or displaying the information in a separate screen or window to draw a user's attention. We also determine, based on the present record, that the claim term "marking without user intervention the first information to alert the user" means that the first information is detected and has some form of marking applied to it without user intervention.

2. "performing an operation related to second information"

Petitioner argues that "performing an operation related to second information," which appears in claims 19, 25, 31, 57, 73, 85, 96, and 99, should be construed to encompass "operations relating to either already-existing information or the entry of new information, whether it is second information itself or other information related to second information." Pet. 8. Patent Owner provides no argument or construction for this term. *See* Prelim. Resp. 7-9. Petitioner contends that their proposed construction is consistent with the '854 patent examples and claims, which describe operations that include "new information," without restricting such

information to the "second information" of the claims. Pet. 8 (citing Ex. 1001, dependent claims 22, 23, and 24; Fig. 7, 6:28-33 (discussing Example 2 and user editing of data being added to database).

Based on the record before us, "performing an operation related to second information," encompasses operations on pre-existing information or new information that may be the second information itself or related to the second information.

3. Means-Plus-Function Limitations

Section 112, ¶ 6[2] permits an element in a claim for a combination to be expressed as a means for performing a specified function without the recital of structure in support thereof, but with the provision that "such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof." "[T]he corresponding structure for a § 112 ¶ 6 claim for a computer-implemented function is the algorithm disclosed in the specification." *Aristocrat Techs. Austl. Party. Ltd. vs. Int'l Game Tech*., 521 F.3d 1328, 1333 (Fed. Cir. 2008) (quoting *Harris Corp. v. Ericsson Inc*., 417 F.3d 1241, 1249 (Fed. Cir. 2005)).

Petitioner asserts that the claims with means-plus-function limitations, specifically claims 31-35, 79-84, and 99, lack corresponding structure or algorithms as required under Section 112, ¶ 6. Pet. 9-13 (providing charts and citations to Ex. 1004 showing means-plus-function limitations with

[2] Section 4(c) of the Leahy-Smith America Invents Act, Pub. L. No. 112-29, 125 Stat. 284 (2011) ("AIA"), re-designated 35 U.S.C. § 112, ¶ 6, as 35 U.S.C. § 112(f). Because the '854 patent has a filing date before September 16, 2012 (effective date of AIA), we use the citation § 112, ¶ 6.

missing algorithm disclosures for independent claims 31, 79, and 99). Patent Owner provides no discussion or argument indicating the algorithms or structures corresponding to the means-plus-function limitations of claims 31-35, 79-84, and 99.

We agree with Petitioner that independent claims 31, 79, and 99 lack corresponding algorithms as required for computer-implemented functions. Thus, we are unable to interpret independent claims 31, 79 and 99 and dependent claims 32-35 and 80-84 due to the lack of disclosed structures. A lack of sufficient disclosure of structure that corresponds to the claimed function under Section 112, ¶ 6 renders a claim indefinite. *See Blackboard, Inc. v. Desire2Learn, Inc*., 574 F.3d 1371, 1382 (Fed. Cir. 2009).

4. "associated"

"Associated" appears in the claim term "second information associated with the first information from a second application program" recited in claim 19. Patent Owner contends that "associated" should be construed as meaning "a pre-existing connection or relationship." Prelim. Resp. 8. Patent Owner claims this is the term's ordinary and customary meaning in accordance with the intrinsic evidence that shows searching for information that is "related" to text. *Id*. (citing Ex. 1001, Abstract, 3:63-66, 5:66-6:2, 4:43-45, 4:57-58). Petitioner offers no construction for this term.

We are not persuaded by Patent Owner that "associated," as claimed, should be narrowly construed as requiring "a pre-existing" connection or relationship. Indeed, the '854 specification refers to related information that possibly may match the searched data or data that corresponds to part of a typed name. *See* Ex. 1001, 3:63-67, 4:43-58. Based on the record before us, we determine that "associated" is construed as "connected or related."

5. "second information associated with the first information from a second application program"

Patent Owner contends that "second information associated with the first information from a second application program" should be construed such that "second application program" contains both the first and second information and associates the second information with the first information. Prelim. Resp. 8. Petitioner offers no construction for this term.

As discussed above, we disagree with Patent Owner that "associated" requires a "pre-existing" relationship. Furthermore, Patent Owner has not provided persuasive evidence or argument that "second information associated with the first information from a second application program" requires that the second application program contain both the first and second information. Based on the record before us, we decline to adopt Patent Owner's construction, and determine that "second information associated with the first information from a second application program" encompasses second information that is related to or connected with the first information from a second application program.

6. "user designation"

Patent Owner provides a construction for a term, "user designation," a term that does not appear in the claims of the '854 patent. Prelim. Resp. 9. Accordingly, we decline to adopt Patent Owner's construction for purposes of this institution decision as unnecessary.

7. "application program"

Patent Owner argues that "application program," which appears in independent claims 19, 25, and 31, should be construed as "an independently executable computer program designed to assist in the performance of a

specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." Prelim. Resp. 9. Patent Owner asserts that the term as used in the patent specification refers to software products, such as Microsoft Word and Excel. *Id*.

The term "application program" does not appear in the specification of the '854 patent. However, we are not persuaded by Patent Owner that "application program" is limited to programs "designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." The '854 patent expressly states that the invention is not limited to such programs. Ex. 1001, 9:64-10:10.

Based on the record before us, we determine that "application program" encompasses an independent executable program.

B. Asserted Grounds of Unpatentability Based on LiveDoc/Drop Zones (Ex. 1005)

Petitioner contends that LiveDoc/Drop Zones anticipates claims 19-22, 24-28, 30-34, 57-60, 62-69, 72-76, 78-82, 84, 85, 96, and 99 under 35 U.S.C. § 102(a). Pet. 15-26.

1. LiveDoc/Drop Zones (Ex. 1005)

LiveDoc discloses a structure detection process running in the background on the visible text of a document entered by a user. "The results of LiveDoc's analysis are . . . presented by visually highlighting the discovered structures with a patch of color around the structure." Ex. 1005, 55 (first column). "Holding down a function key places the document in 'LiveDoc mode' and presents the highlighted structures; releasing the

function key returns the document to normal." *Id.* In LiveDoc mode the highlighted structures can be selected to cause certain actions to occur. *Id.*

Drop Zones is titled, "Drop Zones – An Extension to LiveDoc" and discloses a user interface for managing LiveDoc objects in the context of a set of typical user tasks. Pet. 17. A Drop Zones "assistant" takes features identified by LiveDoc, interprets the feature's meaning, and recommends appropriate actions, made visible when the user selects structures identified by LiveDoc, and drags and drops them to the assistants. Ex. 1005, 60.

2. Anticipation Based on LiveDoc/Drop Zones (Ex. 1005)

Petitioner contends that LiveDoc/Drop Zones anticipates claims 19-22, 24-28, 30-34, 57-60, 62-69, 72-76, 78-82, 84, 85, 96, and 99 under 35 U.S.C. § 102(a). Pet. 15-26. Petitioner provides claims charts addressing the limitations of claims 19, 20-22, 24, 57-59, 62-69, 72, and 85. Pet. 17-25. Patent Owner argues that LiveDoc/Drop Zones fails to teach or suggest "marking without user intervention the first information" as recited in independent claims 19, 25, and 31 because the background highlighting in LiveDoc/Drop Zones is only shown to the user on-demand. Prelim. Resp. 28-30. A similar limitation is found in independent claims 57, 73, 79, 85, 96 and 99 specifically "identifying without user intervention or designation the first information."

Based on the record before us, Petitioner has not shown persuasively that LiveDoc/Drop Zones displays the highlighted information to the user without user intervention. Pet. 17-18 (claim limitation 19c), 20-21 (claim limitation 57b). The Live Doc reference states that "LiveDoc works quietly in the background and displays the results of its analysis *on demand*." Ex. 1005, 56 (second column) (emphasis added). The on demand marking in

LiveDoc is shown when the user holds down the function key to place the document in "LiveDoc mode" to present the highlighted structures. *Id*. at 55 (first column). When the user releases the function key, the document is returned to normal without any highlighting. *Id*.

Although Petitioner asserts that "the results of LiveDoc's analysis are . . . presented by visually highlighting the discovered structures with a patch of color around the structure" (Pet. 17), Petitioner's argument ignores the user action required to enter LiveDoc mode to view the highlighted structures. Ex. 1005, 56. Although we agree that LiveDoc runs in the background and discloses detecting structures without user intervention, the visual highlighting is displayed or brought to the user's attention only at the user's demand. Petitioner's declarant testified that "[o]ne of ordinary skill in the art at the time of the claimed invention would construe 'marking' to mean make visible for identification (e.g., drawing attention to or highlighting)." Ex. 1002 ¶ 49. In accordance with our construction of marking discussed in Section II.A.1, we agree with Patent Owner (Prelim. Resp. 29-30) that the highlighting discussed in LiveDoc is not visible to the user unless the proper display mode is first selected by the user holding down the function key. Ex. 1005, 55. Indeed, LiveDoc discloses that "on demand" the highlighted structures are displayed only after the user holds down the function key. Ex. 1005, 55. Thus, we are not persuaded that LiveDoc discloses that the first information is detected, and has some form of marking applied to it without user intervention, as required in independent claims 19, 25, and 31 and related claims.

With respect to the means-plus-function limitations in claims 31-34, 79-82, 84, and 99, we conclude in Section II.A.3 that we are unable to

interpret these claims due to the lack of disclosed structures. Thus, we also find that the information presented in the Petition and the Preliminary Response does not show there is a reasonable likelihood that Petitioner would prevail in its challenge of claims 31-34, 79-82, 84, and 99, which contain means-plus-function limitations.

We conclude Petitioner has not shown a reasonable likelihood it would prevail in challenging claims 19-22, 24-28, 30-34, 57-60, 62-69, 72-76, 78-82, 84, 85, 96, and 99 as anticipated by LiveDoc/Drop Zones.

3. Obviousness Based on LiveDoc/Drop Zones (Ex. 1005)

Petitioner contends that claims 23, 29, 35, 61, 64-71, 77, and 83 are obvious in view of LiveDoc/Drop Zones. Pet. 26-29.

As discussed above, Petitioner has not persuasively shown that LiveDoc/Drop Zones discloses that the first information is detected and has some form of marking applied to it without user intervention. Pet. 17-18 (claim limitation 19c), 20-21 (claim limitation 57b).

We conclude Petitioner has not shown a reasonable likelihood it would prevail in challenging claims 23, 29, 35, 61, 64-71, 77, and 83 as obvious over LiveDoc/Drop Zones.

4. Obviousness Based on LiveDoc/Drop Zones (Ex. 1005) and Nielsen (Ex. 1009)

Petitioner contends that claims 22-24, 28-30, 34, 35, 60-62, 76-78, and 82-84 are obvious in view of LiveDoc/Drop Zones. Pet. 29-30. Petitioner relies on Nielsen for additional claim limitations not previously discussed as being disclosed in LiveDoc/Drop Zones. For the same reasons discussed above with respect to obviousness based on LiveDoc/Drop Zones, we conclude Petitioner has not shown a reasonable likelihood it would

prevail in challenging claims 22-24, 28-30, 34, 35, 60-62, 76-78, and 82-84 as obvious over LiveDoc/Drop Zone and Nielsen.

C. Asserted Grounds of Unpatentability Based on Domini (Ex. 1006)

Petitioner contends that Domini anticipates claims 19, 20, 22-26, 28-32, 34, 35, 57, 58, 60-74, 76-80, 82-85, 96, and 99 under 35 U.S.C. § 102(e). Pet. 31-39. We determined above that the means-plus-function limitations of claims 31, 32, 34, 35, 79, 80, 82-84 and 99 are indefinite, lacking sufficient disclosure of structure that corresponds to the claimed function under Section 112, ¶ 6. *See Blackboard, Inc.*, 574 F.3d at 1382. We address Petitioner's grounds based on Domini as they apply to challenged claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85, and 96.

1. Domini (Ex. 1006)

Domini discloses identifying and correcting spelling errors in a document created by a word processing program. Ex. 1006, Abstract; 4:65-5:11. The user selects the "spelling and grammar" command to initialize the spell check program. *Id*. at 16:13-16. Without user intervention, the spell check program identifies misspelled words and presents them in red, bold typeface. *Id*. at 17:27-33; 4:12-16. The spell check program also displays a list of suggested corrections that may be selected and entered into the document by the user. *Id*. at 1:42-44; 12:1-5; 12:61-64.

2. Anticipation based on Domini (Ex. 1006)

Petitioner asserts that claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85, and 96 are anticipated by Domini. Petitioner's claim chart provides citations to Domini that Petitioner contends disclose the corresponding claim limitations in claims 19, 20, 22-24, 57, 58, 60-72, and 85. Petitioner asserts

that the spell checking and correction method in Domini, that identifies words, marks and alerts them to the user and allows a user to select from suggested corrections or add a word to the dictionary, correspond to the limitations of independent claims 19, 25, 31, 57, 73. Pet. 32-33.

Patent Owner contends that Domini fails to disclose "responding to a user selection by performing an operation related to a second information, the second information *associated* with the first information from the second application program" (emphasis added) as recited in independent claims 19, 25, 31, 57, 73 and 79 because there is not a "pre-existing relationship" between the typed information in Domini and the suggested replacement terms from the spell checker. Prelim. Resp. 41. Patent Owner asserts that the spell checking modules in Domini present words based on what the user intended to type and not on a pre-existing relationship. As discussed above, "associated," as used in the claims, does not require a "pre-existing" relationship. We find that Petitioner has shown persuasively that Domini discloses selecting words by comparing the word in the document to words in standard and custom dictionaries. *See* Pet. 30, 33-34 (citing Ex. 1006, 17:27-33). Domini discloses that the word being checked "corresponds" to words in the dictionaries. Thus, we are not persuaded on the present record by Patent Owner's argument that there is no relationship between the words in the document and the words in the dictionaries. *See* Prelim. Resp. 41.

Further, we are not persuaded by Patent Owner's contention that Domini fails to disclose a second application program. Prelim. Resp. 42-45. On the present record, we are persuaded that Domini discloses multiple application programs in Figure 1, which identifies word processing program 37A and spell checker program 37B separately. Pet. 32 (citing Ex. 1006, Fig

1). Domini expressly refers to these "application programs 37" as "different programs." Ex. 1006, 7:41-51.

We are also not persuaded by Patent Owner's contentions that adding a word to a dictionary, or replacing a word in a document, is not an "operation related to second information, the second information associated with the first information from the second application program." Prelim. Resp. 46-48. As discussed in the claim constructions above, "operation related to second information" that is "associated with the first information from a second application program" encompasses second information that is related to or connected with the first information from a second application program. On the present record, we are persuaded that in Domini, the claimed operation of inserting a new word or adding a word to a dictionary is second information that is related to or connected with the first information from a second application program.

Based on the record before us, Petitioner has shown a reasonable likelihood that it would prevail as to claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85, and 96 as anticipated by Domini.

D. Asserted Grounds of Unpatentability Based on Miller (Ex. 1007)

Petitioner contends that Miller anticipates claims 19, 21-25, 27-31, 33-35, 57, 59-63, 72, 73, 75-79, and 81-84 under 35 U.S.C. § 102(e). Pet. 40-45. In addition, Petitioner contends that claims 20, 26, 32, 58, 64-69, 74, 80, 85, 96, and 99 are obvious in view of Miller. Pet. 45-49. We address Petitioner's grounds based on Miller as they apply to all challenged claims.

1. Miller (Ex. 1007)

Miller discloses a computer method to detect and perform actions on structures identified in computer data. Ex. 1007, Abstract. Miller discloses an analyzer server that is used to detect data in an application program such as a word processor or email program. *Id*. at 3:36-38; 3:57-58; 5:5-21. At a user's request a document is analyzed for structures, such as telephone numbers, email addresses or addresses. *Id*. at 5:19-37. The Miller system provides actions that are linked to detected structures in documents that can be provided to the user.

2. Anticipation and Obviousness over Miller (Ex. 1007)

Patent Owner argues that Miller fails to disclose "second information" related to the first information from the second application, and fails to disclose marking or identifying the first information "without user intervention" as recited in claims 19, 25, 31, 57, 73, and 79. Prelim. Resp. 48-53.

We agree with Patent Owner that Petitioner failed to identify the "second information" with respect to claim 19. *See* Pet. 41. Petitioner relies on the arguments made in support of the second information as recited in claim 19, for the second information as recited in claims 57, 72, and 85. *See* Pet. 41-48. Petitioner asserts, without explanation or citation, that other information in an address book in Miller would be the claimed "second information" of claim 19. Pet. 41. Miller gives the user an option to put detected structures or information into designated storage, such as placing a detected telephone number into a telephone book. Ex. 1007, Fig. 7. Although Miller may allow additional information to be placed into a telephone or address book entry (*id*. at Fig. 5:47-50), Petitioner has not

identified second information associated with the first information as required in claim 19 and related claims. *See* Prelim. Resp. 49-50.

Patent Owner also contends that Miller fails to disclose marking or identifying the first information "without user intervention" as recited in claims 19, 25, 31, 57, and 73. Prelim. Resp. 51-53. We agree. Miller shows that user action is required to display the structures received from analyzer server. Ex. 1007, Figs. 8 (step 860) and 9 (step 920). Thus, we agree with Patent Owner that Miller requires user intervention to display and fails to disclose or teach "marking without user intervention the first information," as recited in independent claims 19, 25, and 31, or "identifying without user intervention or designation the first information" as recited in independent claims 57 and 73.

Based on the record before us, we conclude Petitioner has not shown a reasonable likelihood it would prevail in challenging claims 19, 21-25, 27-31, 33-35, 57, 59-63, 72, 73, 75-79, and 81-84 as anticipated by Miller, or claims 20, 26, 32, 58, 64-69, 74, 80, 85, 96, and 99 as obvious in view of Miller.

E. Asserted Grounds of Unpatentability Based on Luciw (Ex. 1008)

Petitioner contends that Luciw anticipates claims 19-35, 57-71, 73-85, 96, and 99 under 35 U.S.C. § 102(e). Pet. 49-58. We address Petitioner's grounds based on Luciw as they apply to all challenged claims.

1. Luciw (Ex. 1008)

Luciw discloses a pen-based, handheld device that provides user assistance based on information entered into a document, such as a note area created by a notepad application. Ex. 1008, 2:19-22; 6:24-59. When the

user writes certain information, such as a name, it is automatically recognized and converted to formal font. *Id*. at Fig. 2; 3:8-10; 10:10-21; 11:43-45. Luciw allows the user to make a selection from a list of persons with the identified name; when the user makes a selection, information associated with the person, such as the person's full name, is inserted into the document. *Id*. at 11:60-12:6.

2. Anticipation based on Luciw (Ex. 1008)

Patent Owner argues that Luciw fails to disclose a "second application program" or "second information . . . from the second application program" related to the first information from the second application. Prelim. Resp. 54-56. Independent claims 19, 25, 31, 57, 73, 79, 85, 96, and 99 each recite a "second application program" and limitations similar to "second information . . . from the second application program."

Petitioner provides claim charts indicating which portions of Luciw correspond to the identified claims. Pet. 50-57. However, Petitioner fails to identify which program or operation corresponds to the "second information . . . from the second application program" of claims 19, 57 and 85. *See* Pet. 51, 54, 56-57. In addition, Petitioner fails to identify the "second application." Although Luciw discusses that the pen-based system may use other applications (Ex. 1008, 6:55-59), Petitioner does not show how these other applications interact with the notepad application discussed in the Luciw examples cited. *See* Pet. 51-52.

Based on the record before us, we conclude Petitioner has not shown a reasonable likelihood it would prevail in challenging claims 19-35, 57-71, 73-85, 96, and 99 as anticipated by Luciw.

III. CONCLUSION

For the foregoing reasons, we determine that the information presented in the petition establishes that there is a reasonable likelihood Petitioner would prevail in challenging claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85, and 96 of the '854 as unpatentable under 35 U.S.C. § 102(e) over Domini.

The Board has not made a final determination on the patentability of any challenged claims.

IV. ORDER

For the reasons given, it is

ORDERED that *inter parties* review is instituted as to claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85, and 96 as unpatentable under 35 U.S.C. § 102(e) over Domini;

FURTHER ORDERED that pursuant to 35 U.S.C. § 314(a), *inter partes* review of the '854 patent is hereby instituted commencing on the entry date of this Order, and pursuant to 35 U.S.C. § 314(c) and 37 C.F.R. § 42.4, notice is hereby given of the institution of a trial;

FURTHER ORDERED that all other grounds are *denied*, and no ground other than those specifically granted above is authorized for the *inter partes* review as to the '854 patent.

Petitioner:

David Fehrman
dfehrman@mofo.com

Mehran Arjomand
marjomand@mofo.com

Matthew Smith
smith@turnerboyd.com

Zhuanjia Gu
gu@turnerboyd.com

Patent Owner:

Robert Asher
rasher@sunsteinlaw.com

Bruce Sunstein
bsunstein@sunsteinlaw.com

Exhibit 6O

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Apple Inc., Google Inc., and Motorola Mobility LLC

Petitioners,

v.

Arendi S.A.R.L.

Patent Owner.

Case No. IPR2014-00206

Patent No. 7,496,854

Before SALLY C. MEDLEY, HOWARD B. BLANKENSHIP, and TREVOR M. JEFFERSON, Administrative Patent Judges.

PATENT OWNER ARENDI S.A.R.L.'S RESPONSE UNDER 37 C.F.R. § 42.120

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Cases

Statutes

Regulations

EXHIBIT LIST

Arendi Exhibit Number	Description
2001	American Heritage College dictionary 3rd edition 1997 definition of the term "dictionary."
2002	American Heritage College dictionary 3rd edition 1997 definition of the term "designate."
2003	Declaration of John V. Levy, Ph.D.
2004	American Heritage College dictionary 3rd edition 1997 definition of the terms "independent", "executable", "program" and "module".

I. INTRODUCTION

Patent Owner Arendi S.A.R.L. ("Arendi" or "Patent Owner") respectfully requests that the Board find that Petitioners Apple Inc., Google Inc., and Motorola Mobility LLC ("Petitioners") have failed to show by a preponderance of evidence that claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85 and 96 of U.S. Patent No. 7,496,854 (the "854 Patent") are anticipated by U.S. Patent No. 6,085,206 ("Domini").

Petitioners have submitted proposed grounds for challenge based on anticipation. The Board instituted the current IPR based on anticipation of claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85 and 96 by Domini under 35 U.S.C. §102(e) (pre-AIA).

However, as to each of the challenged claims, at least one claim element is missing from the relied-upon reference.

II. OVERVIEW OF THE '854 PATENT

The '854 Patent is directed, among other things, to computer-implemented processes for automating a user's interaction between a first application, such as a word processing application or spreadsheet application, on the one hand, and a second application, such as contact management application having a database, on the other hand.

The invention as claimed provides a significant simplification of prior art methods. In the prior art, a user who has entered first information (e.g. a person's name) in a document must leave the first application program, (e.g. a word processor) and open and search using a second application program (e.g. contact management program) when the user wishes to locate second information related to the first information from the second application program. For example, the user of such prior art systems must search for the first information (e.g. a name) and the second information (e.g. an address) using the second program and then return to the first program and manually enter the second information into the document. This process requires a plurality of actions by the user in order to obtain related information to the information typed within the document. Thus, according to embodiments of the present invention, "the process of creating and updating records in an address database is significantly simplified, since this may now be performed directly from the word processor."

The '854 refers to multiple examples of application programs such as Microsoft Word™, Microsoft Excel™, NOTEPAD™, WORDPAD™, WORDPERFECT™, QUATROPRO™, AMIPRO™, Microsoft Outlook™, ACCESS™, ORACLE™, DBASE™, RBASE™, CARDFILE™. Col. 9 line 64 to col. 10, line 10.

In the '854 Patent, Exhibit 1001, Figs. 1 and 2 are flow charts showing for these interactions a number of scenarios, which are described from col. 4, line 25 to col. 5, line 57. Further details of the interactions are provided in discussion thereafter of the other figures of the '854 Patent and the discussion includes references back to relevant portions of the flow charts in Figs. 1 and 2. Fig. 1 is reproduced below.

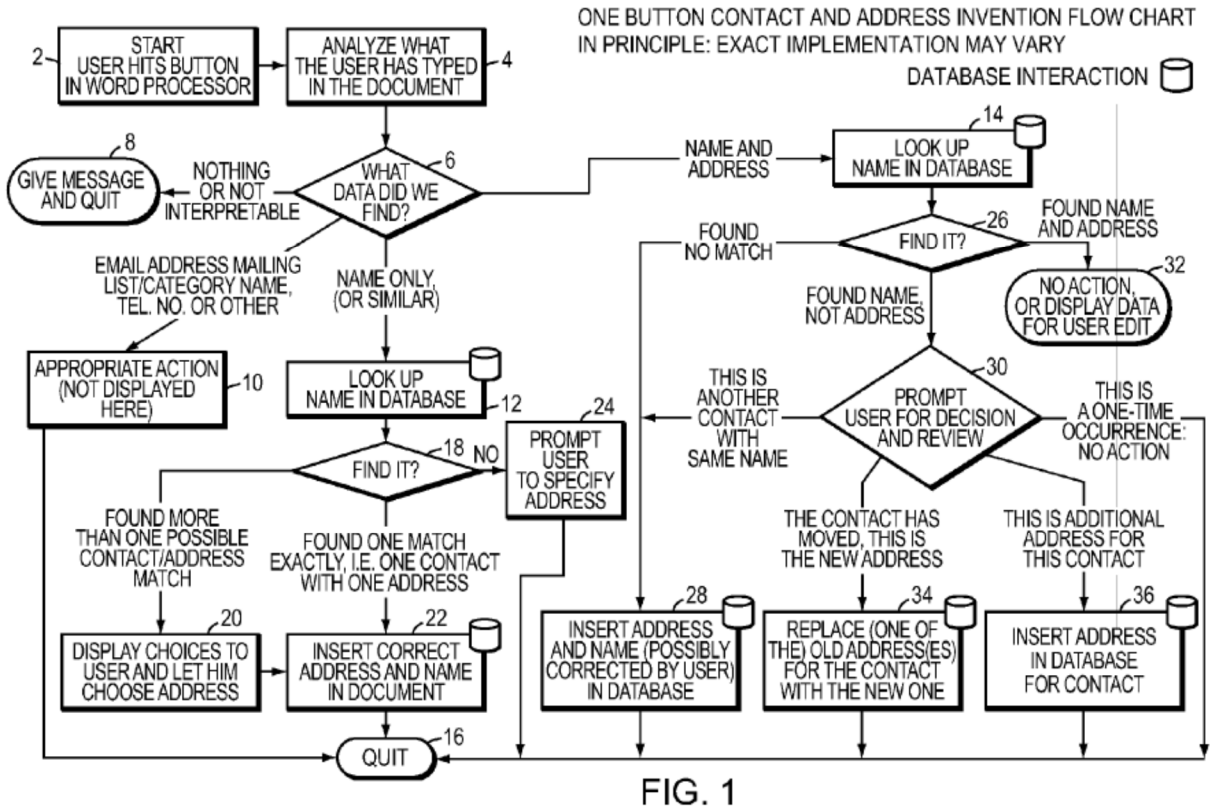


FIG. 1

In various scenarios, text in a document in the first application is analyzed (in step 2 of Fig. 1) to identify first information. Exhibit 1001, col. 4, line 25-49. The analysis takes place without user designation of a specific part of the document to be subject to the analyzing. *Id*.

Once first information has been identified, a number of different scenarios can follow, depending on the circumstances. In one scenario, if the first information includes a name, a search is initiated in the database associated with the second application for the name. *Id*., Fig. 1, steps 6, 12, and 14. If the contact information identified in the document included only a name, and if only a single entry is found in the database for the name and the entry includes a single address, then the address is inserted into the document. *Id*., Fig. 1, steps 6, 12, 18, and 22; Fig. 4; col. 5, lines 63 to col. 6 line 3. Fig. 4, which is reproduced below, shows the document displayed in Microsoft Word after the address has been inserted.

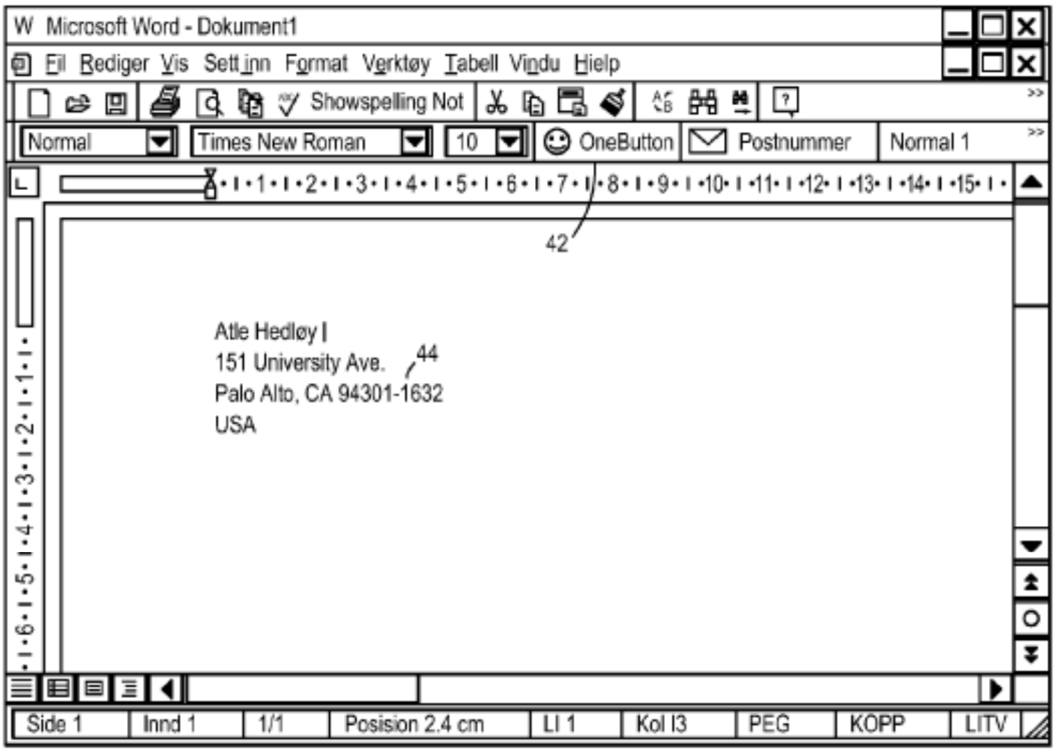


FIG. 4

Shown in Fig. 4 is the One Button 42, which, when pressed, launches the processes just recited, including analyzing the document to identify first information, the searching in the database, and inserting of the address. *Id*., Fig. 2, steps 2, 4; col. 4, lines 25-28; col. 5, lines 58 to col. 6 line 5.

On the other hand, if multiple addresses are found in searching the database for the identified name, these found addresses are displayed, and the user is presented with a choice of which of the addresses to insert. *Id*., Fig. 1, steps 18, 20, and 22; Fig. 10; col. 7, line 25 to col. 8, line 7.

In another scenario, when the user clicks on the "One Button" after having typed into the document a name and an address, the document is analyzed as before (per Fig. 1, step 4) to identify the name and the address. Next, the database is searched for the identified name (per Fig. 1, step 14). If the name happens to be in the contact database but the address in the contact database for that name differs from the address typed by the user into the document (per Fig. 1, step 26), then the user is prompted to make a choice (per Fig. 1, step 30). The user is presented with

a screen shown in Fig. 9, which is reproduced below.

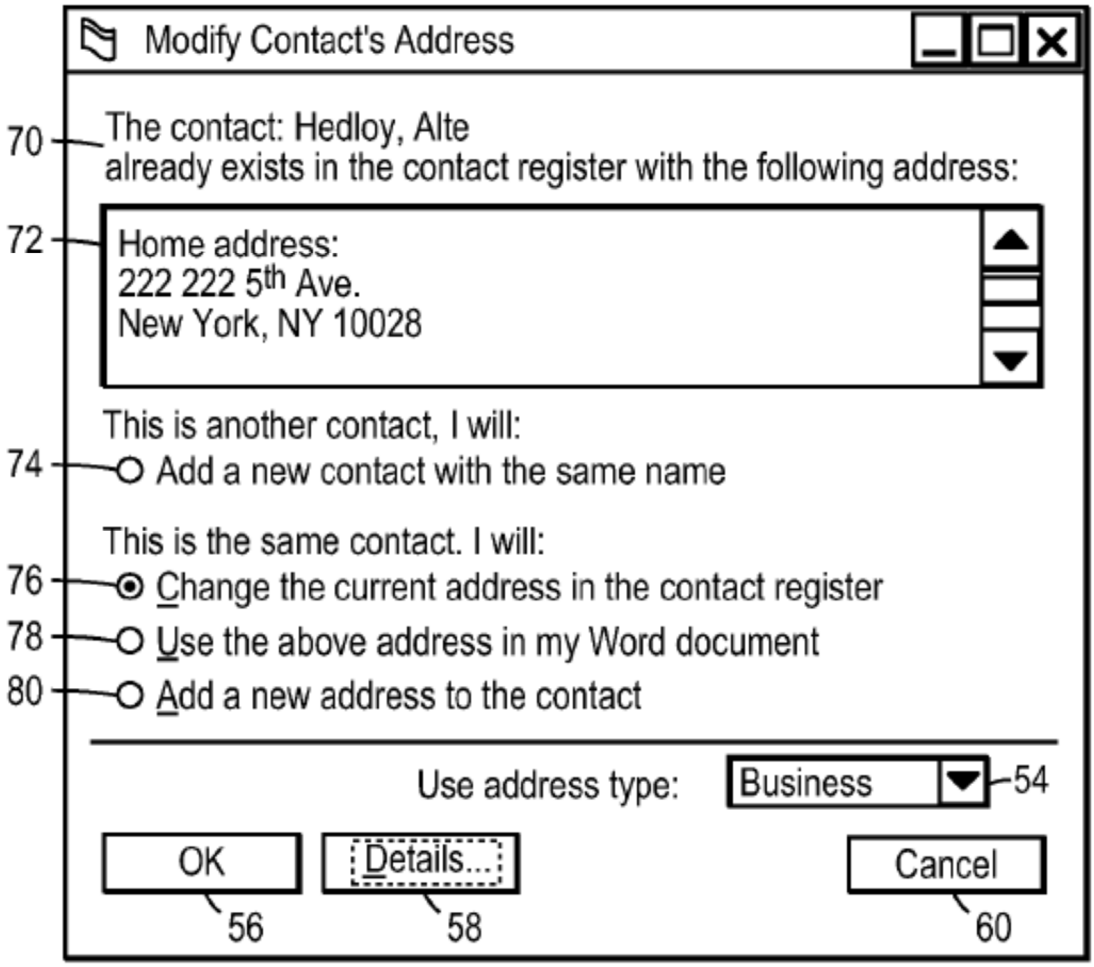


FIG. 9

Fig. 9 represents a screen presented to the user in which the user is given a series of choices that can be made in this specific context. *Id.*, Col. 6 line 66 to col. 7 lines 23. The screen reproduces the name that is both in the document and in the contact database, and it also displays the address that is in the contact database for that name. Below this information, the screen offers a total of four choices in two categories. As shown in Fig. 9 and explained in the '854 Patent, the user is

enabled to select one of the four choices. *Id*. The first category is that "This is another contact" and the choice under this category is to "Add a new contact with the same name". The second category is that "This is the same contact", and the user is given three other choices for the contact: (a) "Change the current address in the contact register"; (b) "Use the above address [reproduced from the contact database] in my Word document"; and (c) "Add a new address to the contact".

These same four choices are also illustrated in connection with item 30 of Fig. 1 of the '854 Patent, which shows logical flow followed in described embodiments of the invention. Item 30 is labeled "PROMPT USER FOR DECISION AND REVIEW", and there are four outcomes shown from this item: (1) "THIS ANOTHER CONTACT WITH THE SAME NAME"; (2) "THE CONTACT HAS MOVED, THIS IS THE NEW ADDRESS"; (3) "THIS IS A ONE-TIME OCCURRENCE: NO ACTION"; and (4) "THIS IS ADDITIONAL ADDRESS FOR THIS CONTACT". These choices are described in the '854 Patent, col. 4, line 62-col. 6 line 8.

It can be seen that the first of the four choices is to add a new contact, and two of the remaining choices are specific ways of updating an existing contact. (Another choice offered is to do neither of these and simply use the address in the Word document as typed.) Consequently, the screen of Fig. 9 presents to the user

a choice, among other things, between competing alternatives of storing a new contact or updating an existing contact.

III. CLAIM CONSTRUCTION

In an *inter partes* review, the Patent Trial and Appeal Board gives patent claims their "broadest reasonable construction in light of the specification of the patent." 37 C.F.R. § 42.100(b); *Phillips v. AWH Corp.*, 415 F.3d 1303, 1316 (Fed. Cir. 2005) (*en banc*). "To ascertain the scope and meaning of the asserted claims, we look to the words of the claims themselves, the specification, the prosecution history, and lastly, any relevant extrinsic evidence. *Phillips v. AWH Corp*., 415 F. 3d 1303, 1315-17 (Fed. Cir. 2005) (en banc)." *In re Baxter Int'l*, 678 F. 3d. 1357, 1362 (Fed. Cir. 2012) (Standard on appeal from *ex parte* reexamination.). Extrinsic evidence is relevant only to the extent it is consistent with the specification and file history. *Phillips*, 415 F.3d at 1319.

A disputed claim term should be given its plain and ordinary meaning, as the term would have been understood by a person of ordinary skill in the art at the time of the invention. The person of ordinary skill in the art is deemed to have read the claim terms in the context of the "intrinsic evidence" (claims, specification, and prosecution history), *Phillips*, 415 F.3d at 1315. The claims themselves provide substantial guidance in determining the meaning of particular claim terms. *Id*. at 1314. First, a term's context in the asserted claim can be very instructive. *Id.* Other

claims can also aid in determining the claim's meaning because claim terms are typically used consistently throughout the patent. *Id.* Differences among the claim terms can also assist in understanding a term's meaning. *Id.*

The claims "must be read in view of the specification of which they are a part." *Id.* at 1314–15. "[T]he specification 'is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term." *Id.* at 1315 (quoting *Markman v. Westview Instruments Inc*., 52 F.3d 967 (Fed. Cir. 1995) (*en banc*), *affd.* 517 U.S. 370 (1996). "The construction that stays true to the claim language and most naturally aligns with the patent's description of the invention will be, in the end, the correct construction." *Phillips*, 415 F.3d at 1323.

A. Application Program

At page 12 of the Decision by the PTAB for Institution of Inter Partes Review, the PTAB holds that the term "application program" means an "independent[ly] executable program".[1]

[1] In construing the term "application program", the PTAB rejected Patent Owner's construction as "an independently executable computer program designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." [citation], p. 10. The PTAB did not take issue with the requirement that the program be "independently

The American Heritage Collection Dictionary, 3rd Edition 1997 defines the term "independent" to mean "not dependent on…..[a] controlling group or system" and "free from the influence guidance, or control of another or others; self-reliant." This dictionary also defines the term "executable" to mean, "to run (a program or an instruction)" and defines the term "program" to mean, "a procedure for solving a problem that involve the collection of data, processing, and presentation of results. Exhibit 2004, American Heritage Collection Dictionary, 3rd Edition 1997. Thus, the Patent Owner agrees with the PTAB's interpretation of an application program as an independently executable program.

1. Application Programs are Independently Executable

The Patent Owner provides additional support from the specification and the Patent Owner's expert for the claim construction of an "application program" as an independently executable program.

---

executable. Rather the PTAB stated "we are not persuaded by Patent Owner" that "application program" is limited to programs "designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." The PTAB retained the requirement that an "application program" is "independently executable". Patent Owner therefore infers that the absence of "-ly" in "independent" is a typographical error.

The background section of the specification provides guidance as to the meaning of "application program." Col 1, lines 34-37 of the '854 patent states:

> Typically, the information is retrieved by the user from an information management source external to the word processor, such as a database program, contact management program, etc.,

And col. 1, lines 45-46 states:

> This requires the user to learn how to use and have access to the database.

Both of the above passages from the background make it clear that the problem being solved by the inventor and claimed in '854 patent relates to obtaining information from an information management program (a second application program) separate from the word processor in which the user is viewing the document (1st program), and the information management program could be used (and learned) by the user independently from the word processor. Thus, these passages do not suggest that there is a subsidiary relationship between the application programs as would be found between an application program and a module or utility.

The specification also provides guidance as to the meaning of an "application program" in the form of examples, and the definition of "application

program" can be construed by the commonly shared features of the computer programs articulated in the examples. See Exhibit 2003 the Declaration of Dr. John Levy ("Levy Decl."), ¶¶ 42-43.

The specification explicitly refers to a "Spreadsheet Application" at column 8, line 55. Other application programs are described in the specification as shall now be detailed.

The specification recites seven separate examples of the invention between col. 5 and col. 9. In six of the examples, Microsoft Word is referenced as the application in which a user enters first information. In the seventh example, first information is entered in a Microsoft Excel spreadsheet. Thus, the application programs include at least Microsoft Word and Microsoft Excel. In each of the seven examples, a database program, such as Microsoft Outlook is accessed. The database program searches a related database for the first information to locate associated second information in one or more database entries. Thus, in addition to Microsoft Word and Microsoft Excel, the inventor has also contemplated at least Microsoft Outlook as an "application program" as would be understood by one of ordinary skill in the art.

The passage running from col. 9, line 64 through col. 10, line 10 of the '854 patent further defines the intended scope of the invention according to the inventor. In this passage, the Patent Owner states that the invention is not limited to

Microsoft Word documents and Excel Spreadsheets, but is applicable to all types of word processing documents. The Patent then lists a plurality of different word processing and spreadsheet applications including NOTEPAD™, WORDPAD™, WORDPERFECT™, QUATROPRO™, AMIPRO™, etc. Based on the recitation of programs, as contemplated by the inventor, "application programs" include word processing programs and spreadsheet programs of the type articulated.

At col. 10, the specification states:

> Although the present invention is defined in terms of information management or is database programs, such as OUTLOOK ™, etc., the present invention is applicable to all types of information management or database programs such as ACCESS™, ORACLE™, DBASE™, RBASE™, CARDFILE ™, including "flat files," etc., as will be readily apparent to those skilled in the art.

In this passage, the inventor defines the scope of his invention as further relating to information management programs and database programs. Thus, the inventor contemplated that database programs and information management programs as types of application programs. The Patent Owner does not make any statements to suggest that a subsidiary program, called by and run under control of an application program, is itself an application program.

Given that the specification identifies word processors, spreadsheet programs, information management programs and database programs as examples

of application programs, the definition of an "application program" can be construed from the commonly shared features. The common features of each of these examples are that they each are computer programs that can be executed independently and none of the example programs runs synchronously under the control of a separate application program. Dr. Levy states in his declaration, that one of ordinary skill in the art reading the '854 patent would understand that the exemplary computer programs mentioned in the detailed description including "word processors, spreadsheet programs, information management programs and database programs" are application programs. Exhibit 2003, Levy Decl. ¶43.

2. Subsidiary Programs are not Independently Executable and therefore are not Application Programs.

Dr. Levy contrasts application programs with subsidiary programs such as modules and utilities. Subsidiary programs extend the functionality of the controlling application program. In order for the functionality of a subsidiary program to be useful, an application program must first be launched and the subsidiary program's functionality activated within the application program. *Id.,* ¶44.

The Patent Owner also notes that in IPR2014-00452 directed to U.S. Patent No. 6,323,853 that shares the same specification as the '854 patent, the PTAB found that "application program" should mean "an independently executable computer program designed to assist in the performing of a specific task, such as

word processing or spreadsheet processing." *Google Inc. and Motorola Mobility LLC v. Arendi S.A.R.L.,* IPR2014-00452, Paper 10. In its findings, the PTAB said: "The evidence cited by Patent Owner provides support for Patent Owner's claim construction argument. Additionally, upon reviewing the specification of the '853 patent, we do not find any disclosure that provides an explicit definition of "application program" contradicting Patent Owner's proposed claim construction."

The Patent Owner therefore construes the term "application program" to mean "an independently executable computer program."

B. "The second information associated with the first information from the second application program."

The term "associated" indicates that there is "an association between" the first information and the second information such that the second information can be found as the result of a search using the first information.

The Federal Circuit has made clear that terms should be construed consistently in the same patent. *Am. Permahedge, Inc. v. Barcana, Inc.,* 105 F.3d 1441, 1446 (Fed. Cir. 1997). Thus, the term "associated" must share the same meaning in all claims of the '854 patent.. As stated in Philips, other claims are useful in determining a claim's meaning. *Phillips*, 415 F.3d at 1315. Thus, we look to claims 57 and 64 for clarification of the term "associated" in the context of

"the second information associated with the first information from the second application program."

The term "associated" is used in independent claim 57 in the limitation: "responding to a user selection by performing an operation related to a second information, the second information associated with the first information from a second application program." "Associated" also appears in claim 64, which depends from claim 57, in the limitation "searching, using the second application program, for the second information associated with the first information."

Claim 64 places the term associated in the context of "searching." Thus, the term "associated" indicates that there is "an association between" the first and second information that can be found as the result of a search. Additionally, claim 64 requires: "retrieving the second information." One of ordinary skill in the art would understand that "searching" and "retrieving" apply to the searching of a data source and the retrieval of the second information from that data source. Exhibit 2003, Levy Decl. ¶ 48. The "association" between the first information and the second information is equivalent to the association of data in a database record. The specification itself supports an understanding that the word "associated" applies to shared information in a database record. Since "associated" in claim 64 is equivalent to the association of data in a database record, the same word "associated" must have the same meaning in the other claims in the '854 patent.

In the specification of the ‘854 patent, the term “associated” appears in Example 1 at col. 5, line 65 through col. 6, line 3.

> The user hits the button 42, for example, marked "OneButton" and the program according to the present invention retrieves the name 40 from the document, searches a database for the name 40, and inserts the retrieved address 44 associated with the name 40 into the document as shown in, for example, FIG. 4. (Emphasis Added).

As used in the specification, the term “associated” refers to the association between a name and an address for a given database entry.

The ‘854 patent discloses multiple embodiments of the invention including 7 enumerated examples, each of which is directed to the first and second information being contact related information where the contact information is associated in a database record. In all examples wherein finding and inserting the second information actually occurs in the ‘854 patent, there is a pre-existing relationship between first and second information. Only if second information is not found is there a lack of a pre-existing relationship, but likewise there is not second information available to complete the requirements of the claim by performing an action, such as inserting. Thus, there must be a pre-existing relationship for an action to be based upon the second information, such as the act of insertion.

The term "associated" must be construed in the context of the claim language, which is directed to a specific contemplated "possibility." The final limitation of claim 57 reads:

> responding to a user selection by performing an operation related to the second information, the second information associated with the first information from a second application program.

In this contemplated possibility, in which an operation is performed related to the second information, the response to user selection is predicated on the second information being identified and "associated with the first information from a second application program" prior to the user selection. This "possibility" can be seen in Fig. 1 of the '854 patent when a search for first information occurs in a database (Step 12). In block 18 of Fig. 1, if the answer to the question "Find It?" is answered affirmatively, only then is the possibility of the insertion operation in block 22 available. Thus, whenever insertion into the document is the result, an association between the first information and the second information exists. As a result, the association between the second information and the first information is necessarily pre-existing.

Independent claim 85 and independent claim 96 also requires "searching." Thus, for all of the claims of the '854 patent, the Patent Owner submits that the

claim language "the second information associated with the first information from the second application program" when read in context of the claims and the embodiments contemplated in the specification of the '854 patent requires that the association between the first and second information is a "pre-existing relationship," such as, the association between field entries for a database record in a database.

IV. OVERVIEW OF THE PRIOR ART

Domini is directed to a combined spell checking and grammar-checking module that operates within a word processing application. See, e.g., Ex. 1006 Abstract, col. 3, lines 1-12, col. 5, lines 1-8.

Fig. 3 reproduced below shows a dialog box for the spell checking and grammar-checking module after a user has selected the "spelling and grammar" command within the application program (e.g. word processing application). The preferred embodiment of the spell checking and grammar module of Domini operates within Microsoft Word. Exhibit 2003, Levy Decl. ¶ 49. According to the '854 patent, Microsoft Word is the application program.

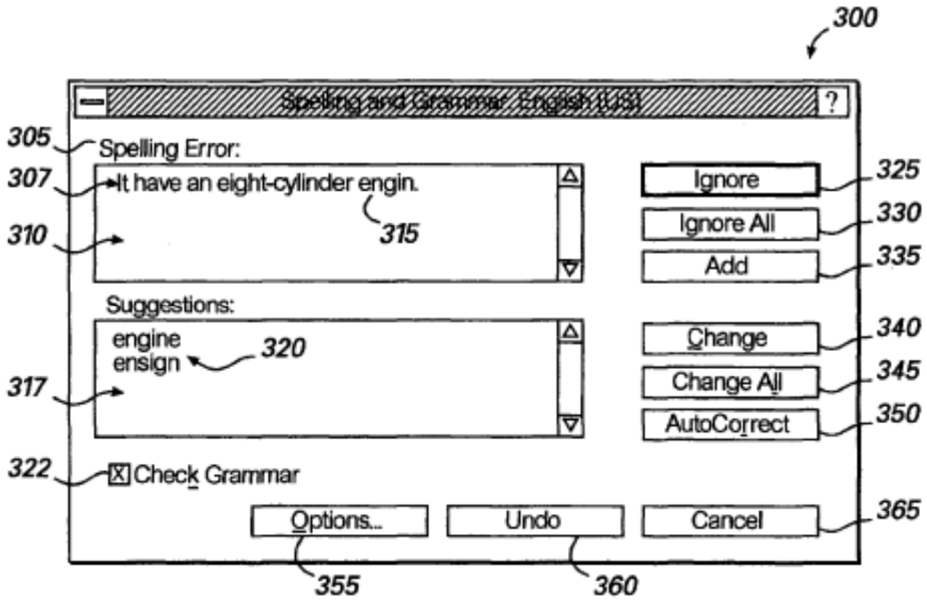


FIG.3

The preferred spell checking and grammar module employs both a Common Grammar Application Programming Interface (CGAPI) and a Common Speller Programming Interface (CSAPI). Ex. 1006, col. 8, lines 51-64 and col. 9, lines 6-19. With respect to the CGAPI and CSAPI, Domini states that the interfaces allow programmers to write applications that can utilize the grammar or spell checker "while being independent of the underlying grammar checker program module" or "spell checker program module. Exhibit 1001, col. 8, lines 58-62 and col. 9, lines 11-16. The API as used in conjunction with the program module allows for the creation of a synchronous relationship between the application program (e.g. word processor) and the program module, such that the program module operates as a subsidiary of the application program. Exhibit 2003, Levy Decl. ¶¶54-57.

Domini also contemplates a spell checking and grammar module that operates in a distributed environment. Ex. 1006 Domini at col. 5, lines 13-26. In the distributed environment, program modules communicate synchronously with each other via network communications where the execution of some modules occurs remotely from the processor used directly by the user executing Word, but the module is still under the control of Word. There is no programming distinction between interfacing synchronously with a local module or synchronously with a remote module other than simply the destination of the module. Exhibit 2003 at ¶¶19 and 50-52 Instead of employing a local procedure call for the module to interface synchronously, the word processing application may use a remote procedure call for the module to interface synchronously. *Id.* at ¶51.

It is noteworthy that the paragraph in Domini discussing a distributed environment confirms that the Domini system works "in conjunction with other program modules." Exhibit 1006 at col. 5, lines 12-17. Nowhere does Domini contemplate the spell checker and grammar checker module operating as an independently executable program because it is always working in conjunction with a word processing program with which the user is editing the document.

In Domini, upon selection of the spelling and grammar command within the word processing application, a sentence is extracted from the word processing document and the spell checking program module is called within the process of

the word processor, such that the compiled code of the module is run. *Id.* at col. 16, line 56-66. Exhibit 2003 at ¶ ¶20, 49. Each word from a sentence is extracted, and the spell checking program module verifies whether the words appears in the one or more dictionaries of the spell checking program module. Exhibit 1006, Domini at col. 17, lines 19-42. If the word does not appear within any dictionary, an error is generated indicating that the word is potentially misspelled and the word processing application accesses a string buffer that contains one or more suggestions for the potentially misspelled word. The suggestions are displayed within a combined spelling and grammar checking dialog box. *Id.* at col. 18, line 4-20. A user may then select between the one or more suggestions to replace the word within the document. *Id.* at col. 18, lines 21-26.

In order to insert a correctly spelled word into the document, it is necessary for the module of Domini to make use of an interface with the word processing application to pass the suggestion to the main word processor module so that the main word processor module can insert the selected suggestion into the document. Exhibit 2003 at ¶20. Even if the module of Domini were running on a different processor from the main word processor module, in a distributed network configuration, the module operates synchronously with the main word processor module and is not independently executable. *Id.* at ¶¶21, 50-52. The remote module receives a series of characters from the main word processor module

through a client/server service such as a remote procedure call. *Id.* at ¶¶20, 46, 51. It is clear from the Domini specification that the main word processor module is suspended/blocked on the local processor while the remote module is executed and returns one or more parameters using a remote procedure call to the processor on which the main word processor module is being executed. The main word processing module process is then unblocked and the parameters are passed to the word processor. A graphical interface is presented to the user through the main word processor module. *Id.* at ¶¶ 51-52. The main word processor module performs the act of insertion. *Id.* at ¶¶ 49, 51-52.

V. PETITIONERS HAVE FAILED TO SHOW THAT DOMINI ANTICIPATES ANY CLAIM

A. Overview of reasons why the Petitioners have failed to show that the claims under review are anticipated by Domini

For each ground asserted in the present *inter partes* review [IPR2014-00206] the reference relied upon fails to teach at least one element of the claims to which the reference is applied.

Consequently, the grounds asserted fail to anticipate any claim under 35 U.S.C. §102(e) as alleged.

Each of the independent claims for review in the '854 patent require both a first application program and a second application program. The Petitioners look to Domini as anticipatory.

However, Domini fails to teach a second application program as required by the independent claims. Domini discloses a spell checker module that operates inside of a word processing application. Thus, Domini discloses only a single application program and therefore, fails to disclose a second application and fails to disclose second information from a second application as required by these claims. Domini discloses operation of its module in a distributed computing environment. Even though the software code of the application program and the module may be on different machines in a distributed computing environment, the code still operates in a synchronous manner, such that the spell-checking module is activated from within the application program and therefore, the spell-checking module is not an independently executable computer program. Thus, Domini fails to teach a second application program.

In fact, even if Domini were created as an independent application program, it could not perform the tasks it is created to perform, including insertion of a correctly spelled word into a document while the user is working on the document with another program such as a word processor.

B. Because Domini's spell checker is a module operating under the control of the first application program, Domini fails to disclose a "second application program", and therefore Domini fails to establish anticipation.

The '854 Patent discloses an embodiment including the retrieval of information (*e.g.*, a name or address) from one application program (e.g. Microsoft Outlook) while a user works simultaneously in another application program (e.g., Microsoft Word). See Exhibit 1001 col. 2, lines 14-23. The independent claims (19, 25, 57, 73 85 and 96) all require "a first application program" and "a second application program." As would be understood by one of ordinary skill in the art at the time of the invention, an "application program" is an "independently executable program" that is independent of and not under the control of another program. *See*, Exhibit 2003, Levy Decl. at, e.g., paragraph 42 – 48. An application program can be contrasted with a subsidiary program, such as a module, or a utility that adds functionality to a host application program and the module or utility operates synchronously with the host application program. Exhibit 2003, Levy Decl. at ¶18. Such subsidiary programs require the operation of a separate application program ("host") in order to be activated and to provide any functionality. *Id.* These modules do not operate independently and are not independently executable programs. *Id.* at ¶¶43-44.

When a subsidiary program resides on the same computer as the host application program, the module operates synchronously within the process of the host application program and the operating system does not block the application program in order for the module to function. *Id.*at ¶18. The module operates as a component of the word processing application program.

When a subsidiary program resides in a distributed computing environment on a server, the module is accessed by a remote procedure call in contrast to a local procedure call from the application program, and operation of the module still depends on the process of the application program, since the subsidiary program operates synchronously with the application program. *Id.* at ¶50-52. In this synchronous communication, the application program suspends/blocks operation while the subsidiary program is executed. When the application program receives results back from the subsidiary program, the application process is continued. In this distributed environment, the subsidiary program functions exactly as if run on a local computer. *Id.* at ¶52.

The '854 Patent describes the interaction between two application programs wherein a user may operate within a first application program without the need for exiting and initiating a second application program to find information managed by the second application program that is related to information entered into a document within the first application program. Examples of application programs

as specified within the specification of the '854 Patent encompass only independently executable computer programs: such as a word processing program, including Microsoft Word, NOTEPAD™, WORDPAD™, WORDPERFECT™, QUATROPRO™, AMIPRO™ a spread sheet program, such as Microsoft Excel, and a contact database program, such as Microsoft Outlook. ACCESS™, ORACLE™, DBASE™, RBASE™, CARDFILE™ Ex. 1001, col. 1, lines 39-42 and col. 9, line 64, to col. 10, line 10.

In contrast, the Domini reference discloses the retrieval of information (e.g., suggested spelling corrections) while a user works within a single program—i.e. a word processor. See, e.g., Ex. 1006 Abstract, col. 3, lines 1-12, col. 5, lines 1-8, col. 8, lines 51-57 , col. 9, lines 6-12 etc. Domini presents the suggested spelling corrections to the user and then allows for insertion of information into the document being drafted in the word processor.

Domini contemplates two different versions of the spell checker and grammar checker program module and mentions a less desirable non-integrated prior art version

1. Word processor and Domini's module on the same computer

The first version operates on a personal computer, such that the word processor and the program module exist within the same personal computer. *Id.* at col. 5, lines 13-15. The spell checker program module of Domini is activated from

within the word processor col. 16, lines 12-16 Exhibit 2003, Levy Decl. at ¶¶18-20, 49. One of ordinary skill in the art reading Domini would understand Domini's module to be a subsidiary program that is activated by the controlling host application program and provides additional functionality to the application program, but is not itself an independently executable program. *Id.*

Thus, this first version of the spell checker program module of Domini is not an "application program" and therefore, this embodiment of Domini lacks the required "second application program" of the independent claims.

2. Word processor and Domini's module in a distributed computing environment

In the second embodiment, the spell checker and grammar checker program module of Domini is part of a distributed computing environment. Exhibit 1006 Domini at col. 5, lines 15-27. In such a configuration, the spell checker and grammar checker program module may be running on a different computer than the word processor, but is still under the control of the word processor. In such a configuration, the application program (word processor) synchronously may execute a remote procedure call to the computer that contains the Domini module. Exhibit 2003 ¶51.The remote procedure call includes the required interface, such as parameters passed from the application program to the module. The application program waits for the module to return with the proposed spellings. *Id.* at¶¶50-52 and the application program allows a user to insert a selected spelling into the

document. *Id.* at ¶20. Consequently, operation of the module in this context still depends on the word processor and the spelling and grammar-checking module operating synchronously.

Therefore, the spelling and grammar-checking module of Domini in a distributed system does not operate independently and only responds to actions through the word processor application program. *Id.* at ¶50. Thus, insertion can only be achieved in Domini by the word processor itself, which is passed the selected correctly spelled word. *Id.* at ¶20. As with the first embodiment, this second embodiment of the spell checker and grammar checker module is a subsidiary program to the word processor and is not itself an application program. *Id.* at ¶¶18, 21, 44, and 50. Also in this environment, the Domini spell checking program module is useless without its host application program and cannot function independently. Therefore, this second embodiment in Domini of the spell checker and grammar checker module lacks the required second application program of the independent claims.

3. Domini's "stand alone" spell checker cannot replace a misspelled word with a suggested correction

The background section of Domini discloses a "stand alone" spell checker Exhibit 1006 col. 1, lines 56-66. A person of ordinary skill in the art would understand that a "standalone spell checker" was an independently executable

computer program that does not interact with a word processor program in order to perform its spell checking function. Such a spell checker, being independently executable, would generate and output its own user display and receive inputs directly from the user. Such a separate user interface is cited by Domini as a disadvantage of performing spell checking using an independently executable program (see Exhibit 1006 col. 1, line 56 to col. 2, line 26). Exhibit 2003, Levy Decl. at ¶24.

Domini describes these stand-alone spell checkers as not being integrated with the word processor. As such, Domini suggests to one of ordinary skill that a stand-alone spell checker would not be capable of inserting text into a word processor. Exhibit 2003 ¶¶24-25, 35.

We have shown that the Domini module is not independently executable, and therefore is not a "second application program" as required by the independent claims. Moreover, nothing in Domini suggests that it could operate as an independently executable program. Because Domini fails to teach a second application program, Domini similarly fails to disclose or suggest using both a "first application program" and a "second application program" as required in independent claims 19, 25, 57, 73, 85 and 96. As a result, Petitioners' Ground 4 fails to establish anticipation of the independent claims listed above.

Additionally, since Domini fails to disclose "a second application", Domini also fails to disclose "second information" "associated with the first information from a second application program" as required by independent claims 19, 25, 57, 73, 85 and 96. Thus, Domini fails to teach the "associated information" limitation of the independent claims in addition to failing to teach a second application program.

For at least these reasons, Petitioners' Ground 4 fails to establish by a preponderance of evidence anticipation by Domini of the above listed independent claims or for dependent claims 20, 22-24, 26, 28-30, 58-, 60-72, 74, and 76-78.

C. Because the claims require a pre-existing association between first information and second information, Domini fails to anticipate the claims.

In the Decision instituting this *inter partes* review, the PTAB declined to read the term "association" to require a pre-existing relationship between first information and second information. However, as discussed above in Section III.B, the Patent Owner submits that the claim language "the second information associated with the first information from the second application program" when read in context of the claims and the embodiments contemplated in the specification of the '854 patent requires that the association between the first and second information is a "pre-existing relationship." Domini does not teach this limitation.

Domini's spell checker and grammar checker module operates by looking at every term in the text (e.g., textually delineated grouping of characters) and comparing it to a dictionary list, and if the term is not found in the list, then the term is highlighted, and a listing of suggested words is presented to the user from which the user makes a selection. Exhibit 1006 col. 4, lines 13-16. The independent claims 19, 25, 57, 73, 85, and 96 each require an association between second information and first information and more specifically require "the second information associated with the first information from a second application program." As explained above in the claim construction section III.B., at least for independent claim 57 and its dependent claims, and claims 85 and 96, the association is of a type that the second information can be found as a result of a successful search for the first information. The association between the first and second information must be pre-existing, such as the relationship between field entries of a database record. Independent claim 57 and dependent claim 64 both use the term "associated" and the Federal Circuit has found that terms within a patent should be construed consistently. *Phillips,* 415 F.3d at 1314. In claim 64, the term "associated" appears in the context of the limitations:

> searching, using the second application program, for the second information associated with the first information; and
>
> retrieving the second information.

In this context, the association between the first and second information is such that the association can be searched and the second information can be retrieved as a result of the search. This type of association is one that includes a "pre-existing" relationship. If the association is not pre-existing, a search for second information based upon first information cannot be performed. Exhibit 2003, Levy Decl. at ¶48.

In Domini, there is a search for each delineated set of characters (i.e. a word in the document), but the list of suggested alternative spelled words comes only if the search is unsuccessful. Exhibit 1006 Fig. 7 no. 725-735. The suggestions that are presented to the user are retrieved from a string buffer ("SRB") and do not share a pre-existing relationship with the misspelled word, such as, the association between field entries for a database record in a database. An association is only made between a suggestion and the misspelled word when the user recognizes the correctly spelled word and selects the suggested word as a replacement for the misspelled word. Thus, the "association" in Domini between the misspelled word (e.g. first information) and the selected correctly spelled word (e.g. second information) is not an "association" as contemplated by claims 57, 85 and 96 in the '854 patent, since the association is not of the type that can be searched for in a database.

Thus, for at least claims 57, 58, 60-72, 85 and 96, Domini fails to teach the limitation of:

> responding to a user selection by inserting a second information into the document, the second information associated with the first information from a second application program.

Domini's embodiments of its spell checker system fail to have the type of association required by the claim limitation.

For at least these reasons, Petitioners fail to establish by a preponderance of evidence anticipation by Domini of at least claims 57, 58, 60-72, 85 and 96. The Patent Owner submits that the same meaning should apply to all claims at issue in this *inter partes* review, and so Petitioners fail to establish by a preponderance of evidence anticipation by Domini of the above listed independent claims 19, 25, 57, 73, 85 and 96, or for any of the claims that depend from those independent claims – claims 20, 22-24, 26, 28-30, 58-, 60-72, 74, and 76-78.

D. Because Domini fails to disclose the "operation related to a second information" of "entering additional data into a database," Ground 4 fails to establish by a preponderance of evidence anticipation of dependent claims 22-24, 28-30, 60-62 and 76-78.

Dependent claims 22, 28, 60 and 76 require "responding to a user selection by performing an operation related to a second information, the second information

associated with the first information from the second application program," and further require that "the operation performed is entering additional data into a database." When the claims dependent on these claims are considered, then all of claims 22-24, 28-30, 60-62 and 76-78 include this limitation.

Domini does not teach this limitation.

The Petitioners' allegation against claim 22 follows from Petitioners' allegation against claim 19, and so the Patent Owner also begins with claim 19. Claim 19 requires, among other things, "responding to a user selection by performing an operation related to a second information, the second information associated with the first information from the second application program."

In their allegation of anticipation by Domini, the Petitioners allege as follows for the claim 19 terms "first information," "second information," and "operation related to a second information":

First Information	**Second Information**	**Operation Related to 2nd Info.**
Words and sentences entered into a word processing program	List of suggested corrections for an identified misspelled word	Replace misspelled word with a word that has been selected by the user from the suggested corrections. [Petition at page 32]

[Petition at page 31]	[Petition at page 32]	

Claim 22 further limits the "operation related to a second information" of claim 19 by requiring that "the operation performed is entering additional data into a database." For this limitation, the Petitioners allege that the "spell check program allows the user to add a misspelled word identified in the document (and for which a suggested correction is provided) to the dictionary (database)" [Petition at page 33].

Specifically, the Petitioners refer to the "Add" button in Domini [Exhibit 1006, col. 12, lines 50-58], which allows users to instruct the spell checker that the supposedly misspelled word is acceptable. The "misspelled word is added to the custom dictionary" [Exhibit 1006, col. 12, lines 53-54]. Thereafter, "[t]he spell checking program module will then skip over every instance of the misspelled word, even in documents other than the present document that is being spell checked and even after the program module has been quit and restarted." [Exhibit 1006, col. 12, lines 54-58]. In summary, adding the supposedly misspelled word to the "custom dictionary" causes the spell checker to ignore that word in the future.

We must bear in mind that:

(i) the "operation" must be "related to a second information . . . ."; and

(ii) the allegedly "misspelled word" is first information (i.e., that word is what the user typed into the document).

Against that background, it can be seen that Domini's act of adding a (misspelled) word from the document (first information) into the dictionary is not an operation related to second information. Rather, it is an operation relating to the first information.

Even if the "second information" (i.e., list of suggested corrections for an identified misspelled word) is associated with the "first information" (i.e., the allegedly misspelled word), that does not address the failure in the Petitioners' allegation; it does not transform the operation (adding "first information" to a dictionary) into an operation relating to the "second information."

Therefore, Domini fails to teach an operation related to a second information, where the operation performed is entering additional data into a database, as required by claim 22.

For at least this reason, the Petitioners' allegation against claim 22 does not support a finding of anticipation by Domini. Consequently, Domini does not anticipate claim 22, and/or any claims that depend from claim 22.

Claims 23 and 24 depend from claim 22, are therefore also not anticipated by Domini. In their allegations of anticipation against 28-30, 60-62 and 76-78, the

Petitioners simply cite to their analysis of claims 22-24. Consequently, Domini also fails to anticipate claims 28-30, 60-62 and 76-78.

For at least this reason, Petitioners' Ground 4 fails to establish anticipation by Domini of dependent claims 22-24, 28-30, 60-62 and 76-78.

E. Because Domini fails to disclose "searching, using the second application program, for the second information associated with the first information," Ground 4 fails to establish by a preponderance of evidence anticipation of dependent claims 64-69, and independent claims 85 and 96.

Independent claims 85 and 96, and dependent claim 64, each require "searching, using the second application program, for the second information associated with the first information." When the claims dependent on these claims are considered, then all of claims 85, 96 and 64-69 include this limitation.

Domini does not teach this limitation.

The Petitioners' allegation against claim 64 follows from Petitioners' allegation against claim 57, from which claim 64 depends. The Petitioners' allegation against claim 57, in turn, refers to the Petitioners' analysis of claim 19. Consequently, the Patent Owner also begins with claims 19 and 57.

Claim 57 requires, among other things, "responding to a user selection by performing an operation related to a second information, the second information associated with the first information from the second application program."

The Petitioners defined the terms of this limitation in their analysis of claim "19d," as follows:

First Information	**Second Information**	**Operation Related to 2nd Info.**
Words and sentences entered into a word processing program [Petition at page 31]	List of suggested corrections for an identified misspelled word [Petition at page 32]	Replace misspelled word with a word that has been selected by the user from the suggested corrections [Petition at page 32]

Thus, in the Petitioners' analysis of claims 57 and 64, those terms have the foregoing meanings.

Turning to claim 64, claim 64 specifies that the "operation" of claim 57 "further comprises . . . searching, using the second application program, for the second information associated with the first information."

However, Domini does not perform such "searching," because Domini teaches only searching for first information, i.e., searching for each word in a dictionary or custom dictionary.

1. Domini does not teach searching a dictionary for "second information" because Domini searches for "first information."

Thus, the question at hand is whether Domini teaches a search of a spell-checker's dictionary "for the second information associated with the first information." It does not. The Petitioners simply allege that there is such a search, but fail to base their allegation with cites from Domini that provide support.

The Petitioners' first citation is to Domini at "16:66-17:57," which teaches:

> "As is well-known in the art, a spell checker program module checks the spelling of a word by comparing the word to the list of words in the standard dictionary and custom dictionaries. If the word does not correspond to one of the words in the standard dictionary or custom dictionaries, then the spell checker program module flags the word as a word that is possibly misspelled." [Domini, column 17, lines 27-33].

Therefore, Domini does not teach searching a dictionary for second information (i.e., suggested corrections), as required by claim 64. Rather, Domini teaches searching the dictionary for first information. (i.e., to determine whether the alleged misspelled word is in the dictionary) and is not looking for any other, i.e., second, information.

For at least this reason, the Petitioners have failed to show by a preponderance of evidence that Domini anticipates claim 64, or any claims that depend from claim 64.

However, this may suggest the following question: How does Domini produce the alleged second information (i.e., how does Domini produce the list of suggested corrections for an identified misspelled word)? The Patent Owner addresses that issue, below.

2. Domini does not teach how the "second information" is created, or that the "second information" is created by searching for the "second information"

How does Domini produce the alleged second information (i.e., how does Domini produce the list of suggested corrections for an identified misspelled word), and does it involve "searching . . . for the second information?"

Petitioners do not allege any portion of Domini as teaching or suggesting an answer for how Domini produces the second information. Rather, Domini teaches only that the suggested corrections somehow appear in a "string buffer" within the "Spell Return Buffer" (aka "SRB") (see Domini, at column 18, lines 4-9). Domini does not teach how the suggested corrections appear in a "string buffer" within the "SRB." *See* Exhibit 2003 ("Levy Decl.") ¶¶ 60-62. Consequently, Domini at "16:66-17:57" does not teach or suggest that the suggested corrections (i.e., the

"second information") appear in the SRB as a result of "searching, using the second application program, for the second information."

For at least this reason, the Petitioners have failed to show by a preponderance of evidence that Domini anticipates claim 64, or any claims that depend from claim 64.

3. Domini does not teach searching an "SRB" for "second information."

The Petitioners' second citation is to Domini at "18:4-9" which teaches:

> "After receiving the error type information at step 730, the preferred application program consults another part of the SRB to locate a string buffer containing suggestions from the spell checker program module at step 735. The suggestions are the information that is displayed in the suggestions list box 317 as shown in FIG. 3." [Domini, column 18, lines 4-9].

This does not teach or suggest that Domini performs a step of "searching, using the second application program, for the second information associated with the first information."

Since Domini teaches that the suggested corrections appear in a "string buffer" within the "Spell Return Buffer" (aka "SRB") (see Domini, at column 18,

lines 4-9), locating the suggested corrections does not involve a search. There is no need to search, because Domini knows the location of the suggested corrections. – i.e., the "string buffer." Indeed, all that is required is to "consult" the "string buffer" to retrieve that information for display. In fact, the Petitioner's cite this same feature of Domini for the "retrieving the second information" element of claim 64.

Therefore, Domini does not teach searching in a "string buffer" within the "Spell Return Buffer."

For at least this reason, the Petitioners have failed to show by a preponderance of evidence that Domini anticipates claim 64, or any claims that depend from claim 64.

As described above, Domini does not anticipate claim 64, or any claim that depends from claim 64, at least because Domini does not teach an "operation" that comprises "searching, using the second application program, for the second information associated with the first information."

Claims 85 and 96 include the same limitation as discussed above for claim 64, and Domini fails to anticipate claims 85 and 96 for the same reasons.

Consequently, the Petitioners have failed to show by a preponderance of evidence that Domini anticipates any of claims 85, 96 and 64-69.

VI. CONCLUSION

For the foregoing reasons, Petitioners have failed to establish by a preponderance of evidence anticipation by Domini of claims 19, 20, 22-26, 28-30, 57, 58, 60-74, 76-78, 85 and 96 of U.S. Patent No. 7,496,854.

Dated: August 26, 2014

Respectfully submitted,

/Robert M. Asher, #30,445/

Robert M. Asher
Registration No. 30,445
Bruce D. Sunstein
Registration No. 27,234
John J. Stickevers
Registration No. 39,387
Thomas J. Tuytschaevers
Registration No. 42, 190
Sunstein Kann Murphy & Timbers LLP
125 Summer Street
Boston, MA 02110
Tel: (617) 443-9292
Fax: (617) 443-0004

CERTIFICATE OF SERVICE

It is certified that on August 26, 2014, copies of Patent Owner Arendi S.A.R.L.'s Response under 37 C.F.R. § 42.120 and Exhibits 2003 and 2004 have been served on Petitioners as provided in 37 C.F.R. § 42.6(e) via electronic mail transmission addressed to the persons at the following addresses:

LEAD COUNSEL FOR PETITIONER APPLE DAVID L. FEHRMAN dfehrman@mofo.com Registration No. 28,600 MORRISON & FOERSTER LLP 707 Wilshire Blvd., Suite 6000 Los Angeles, CA 90017-3543 Tel: (213) 892-5200 Fax: (213) 892-5454	BACK-UP COUNSEL FOR PETITIONER APPLE MEHRAN ARJOMAND marjomand@mofo.com Registration No. 48,231 MORRISON & FOERSTER LLP 707 Wilshire Blvd., Suite 6000 Los Angeles, CA 90017-3543 Tel: (213) 892-5200 Fax: (213) 892-5454
LEAD COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC. MATTHEW A. SMITH smith@turnerboyd.com Registration No. 49,003 TURNER BOYD LLP 2570 W. El Camino Real, Suite 380 Mountain View, CA 94040 Tel: (650) 265-6109 Fax: (650) 521-5931	BACK-UP COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC. ZHUANJIA GU gu@turnerboyd.com Registration No. 51,758
BACK-UP COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC.	

JULIE S. TURNER turner@turnerboyd.com TURNER BOYD LLP 2570 W. El Camino Real, Suite 380 Mountain View, CA 94040 Tel: (650) 265-6109 Fax: (650) 521-5931 kent@turnerboyd.com docketing@turnerboyd.com	

Date: August 26, 2014

/Robert M. Asher, #30,445/
Robert M. Asher
Registration No. 30,445
Sunstein Kann Murphy & Timbers LLP
125 Summer Street
Boston, MA 02110
Tel: (617) 443-9292
Fax: (617) 443-0004

Exhibit 6P

Trials@uspto.gov
Tel: 571-272-7822

Paper 32
Entered: June 9, 2015

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC., GOOGLE INC., and MOTOROLA MOBILITY LLC,
Petitioner,

v.

ARENDI S.A.R.L.,
Patent Owner.

Case IPR2014-00206
Patent 7,496,854 B2

Before HOWARD B. BLANKENSHIP, SALLY C. MEDLEY, and TREVOR M. JEFFERSON, *Administrative Patent Judges.*

JEFFERSON, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
35 U.S.C. § 318(a) and 37 C.F.R. § 42.73

I. INTRODUCTION

A. Background

Apple Inc., Google Inc., and Motorola Mobility LLC (collectively "Petitioner") filed a Petition (Paper 3, "Pet.") to institute an *inter partes* review of claims 19–35, 57–85, 96, and 99 of U.S. Patent 7,496,854 B2 (Ex. 1001, "the '854 patent"). Pet 1; *see* 35 U.S.C. § 311. Arendi S.A.R.L. ("Patent Owner") filed a Preliminary Response (Paper 7, "Prelim. Resp."). Pursuant to 35 U.S.C. § 314, in our Decision to Institute (Paper 9, "Dec."), we instituted trial as to claims 19, 20, 22–26, 28–30, 57, 58, 60–74, 76–78, 85, and 96.

After the Decision to Institute, Patent Owner filed a Patent Owner Response (Paper 16, "PO Resp.") and Petitioner filed a Reply to the Patent Owner Response (Paper 20, "Reply"). An oral hearing (Paper 31, "Tr.") was held on February 4, 2015.

B. Related Matters

Patent Owner has sued Petitioner for infringement of the '854 patent in *Arendi S.A.R.L. v. Apple Inc.*, No. 1:12-cv-01596-LPS (D. Del.); *Arendi S.A.R.L. v. Google Inc.*, No. 1:13-cv-00919 (D. Del.); and *Arendi S.A.R.L. v. Motorola Mobility LLC*, Case No. 1:12-cv-01601-LPS (D. Del.). Pet. 1; Paper 6, 2–3. The '854 patent is also the subject of a petition in IPR2014-00207 filed by Petitioner. *Id*. at 3–4. We granted *inter partes* review as to claims 1–12 and 36–49 of the '854 patent in *Apple Inc. et al v. Arendi S.A.R.L.*, Case IPR2013-00207, slip op. at 23–24 (PTAB June 11, 2014) (Paper 9).

C. The Asserted Ground

We instituted trial based on the ground of unpatentability set forth in the table below. Dec. 16–18, 22.

Reference	Basis	Claims Challenged
Domini[1]	35 U.S.C. § 102(e)	19, 20, 22–26, 28–30, 57, 58, 60–74, 76–78, 85, and 96

D. The '854 Patent

The '854 patent, titled "Method, System and Computer Readable Medium for Addressing Handling From a Computer Program," relates to computer implemented processes for providing a computer program, such as a word processing program or spreadsheet program, that is coupled to an information management source, such as a database program or contact management program. Ex. 1001, 1:19–50.

Figures 3 and 4 of the '854 patent are reproduced below.

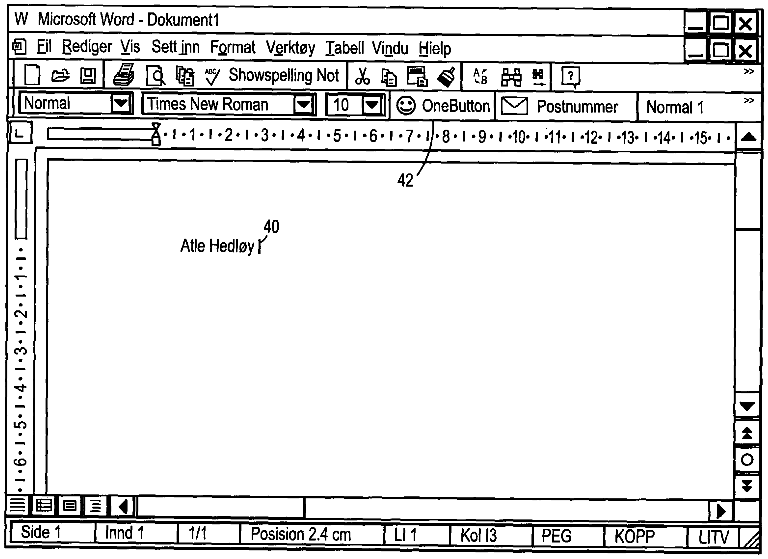


FIG. 3

[1] U.S. Patent No. 6,085,206, issued July 4, 2000, filed June 20, 1996 (Ex. 1006, "Domini")

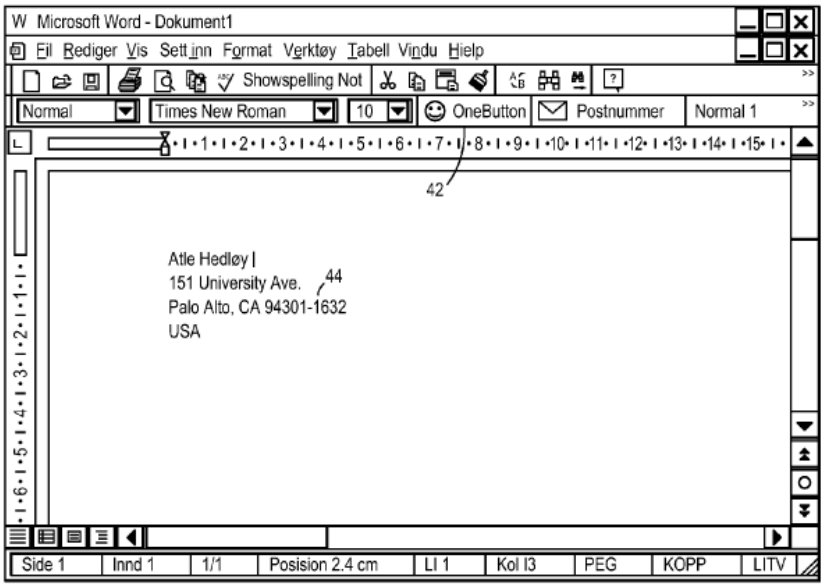


FIG. 4

Figure 3 illustrates the inputting of a name to be searched into a document. Figure 4 illustrates a retrieved address that is inserted into a document. Ex. 1001, 2:51–57. The user types a name into the document. When the user clicks on OneButton 42, the claimed process is launched, retrieving name 40 from the document, searching a database for name 40, and inserting the retrieved address associated with name 40 into the document as shown in Figure 4. *Id*. at 5:60–6:5.

Figure 2 of the ’845 patent, illustrating a flow chart of a method for address handling within a computer program, is reproduced below.

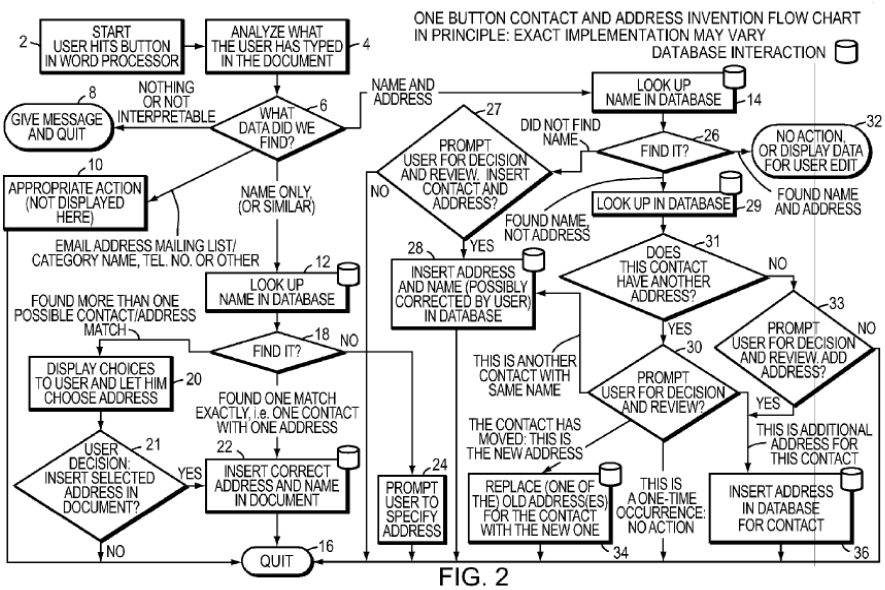


FIG. 2

Figure 2 depicts a flow chart of the address handling process initiated by the user clicking on OneButton 42 of Figure 4. At step 4, text typed by the user in a document is analyzed for contact information. At step 6, if the identified contact information includes a name, a search occurs in the database at step 12. When the database finds a name with more than one possible matching address, the user is prompted for a decision, and that selected information is added to the document at step 22. *Id*. at 5:10–22, 6:4–5.

Independent claim 19, reproduced below, is illustrative of the claimed subject matter:

> 19. A method for information handling within a document created by a first application program comprising the steps of:
>
> entering a first information in the first application program;
>
> marking without user intervention the first information to alert the user that the first

> information can be utilized in a second application program; and
>
> responding to a user selection by performing an operation related to a second information, the second information associated with the first information from the second application program.

II. ANALYSIS

A. Claim Construction

The Board interprets claims of an unexpired patent using the broadest reasonable construction in light of the specification of the patent in which they appear. *See* 37 C.F.R. § 42.100(b); *In re Cuozzo Speed Techs., LLC*, 778 F.3d 1271, 1279–81 (Fed. Cir. 2015). Claim terms generally are given their ordinary and customary meaning, as would be understood by one of ordinary skill in the art in the context of the entire disclosure. *See In re Translogic Tech., Inc.*, 504 F.3d 1249, 1257 (Fed. Cir. 2007).

1. "marking . . . the first information to alert the user"

The claim phrase "marking without user intervention the first information to alert the user" appears in independent claims 19 and 25. In the Decision to Institute, we preliminarily construed "marking without user intervention the first information to alert the user" to mean that the first information is detected and has some form of marking applied to it without user intervention. Dec. 8. We also determined that "marking" included the acts of highlighting, designating, or displaying the information in a separate screen or window to draw a user's attention. *Id*.

The parties do not dispute this preliminary construction. Based on the record before us, we determine that "marking" includes highlighting,

designating, or displaying the information in a separate screen or window to draw a user's attention.

2. "performing an operation related to second information"

The claim phrase "performing an operation related to second information," appears in independent claims 19, 25, 57, 73, 85, and 96. In the Decision to Institute, we determined that "performing an operation related to second information" encompasses operations on pre-existing information or new information that may be the second information itself or related to the second information. Dec. 8–9. The parties do not dispute this preliminary construction. Based on the complete record, we determine that "performing an operation related to second information" includes operations on pre-existing information or new information that may be the second information itself or related to the second information.

3. "associated" and "second information associated with the first information from a second application program"

"Associated" appears in the claim phrase "second information associated with the first information from a second application program" recited in independent claims 19, 25, 57, 73, 85, and 96. In the Decision to Institute, we determined preliminarily that "associated" is construed as "connected or related" (Dec. 10) and that "second information associated with the first information from a second application program" included second information that is related to or connected with the first information from a second application program (Dec. 11).

Patent Owner contends that because "associated" in dependent claim 64 describes searching for and retrieving the second information "associated" with the first information, this indicates that the "association" is

equivalent to the association of a data in a database record. PO Resp. 16. Thus, Patent Owner argues that the searching limitations found in dependent claim 64 and independent claims 85 and 96 "requires that the association between the first and second information is a 'pre-existing relationship,' such as, the association between field entries for a database record in a database." PO Resp. 18–19.

Patent Owner further contends that the '854 patent discloses embodiments that refer to a connection between a name and address, or other pre-existing relationship that is akin to a database entry. PO Resp. 17 (citing Ex. 1001, 5:65–6:3). Patent Owner argues that the '854 patent embodiments refer to finding and inserting the second information, showing that "there must be a pre-existing relationship for an action to be based upon the second information, such as the act of insertion." PO Resp. 17.

We are not persuaded by Patent Owner's argument that a "pre-existing" relationship is required for "second information associated with the first information from a second application program" as recited in independent claims 19, 25, 57, 73, 85, and 96. The '854 specification refers to related information that may match the searched data or data that corresponds to part of a typed name. Ex. 1001, 3:63–67, 4:43–58; *see* Dec. 10; Reply 11–12. Indeed, the '854 patent written description states that there may be "more than one *possible* contact/address match" to the first information and that "the program displays menu choices to the user to let him choose an appropriate answer" to insert. Ex. 1001, 4:46–49 (emphasis added). Thus, Patent Owner has not demonstrated that a pre-existing relationship is described in the '854 specification.

Patent Owner's arguments limiting the term "associated" to the examples in the '854 specification referring to databases also is not commensurate in scope with the breadth of the claims or the broadest reasonable interpretation. We must be careful not to read a particular embodiment appearing in the written description into the claim if the claim language is broader than the embodiment. *See In re Van Geuns*, 988 F.2d 1181, 1184 (Fed. Cir. 1993) (stating that "limitations are not to be read into the claims from the specification"). In the present case, Patent Owner has not shown persuasively that a person of ordinary skill would have understood the "second information associated with the first information from a second application program" as being limited to a "pre-existing relationship" between two pieces of information based on the claims, embodiments, and examples in the '854 specification. To the contrary, the '854 specification describes that a program operation based on a name or initials (the first information) could return more than one possible matching second information for insertion. Ex. 1001, 4:43–58.

In sum, under the broadest reasonable interpretation, we do not determine that "associated" as recited in "second information associated with the first information from a second application program" is limited to a pre-existing relationship. We determine that "associated" is construed as "connected or related" and that that "second information associated with the first information from a second application program" includes second information that is related to or connected with the first information from a second application program.

4. "application program"

In the Decision to Institute, we determined that "application program" encompasses an independent executable program. Dec. 11–12. In so determining, we rejected Patent Owner's narrow construction of application program as "an independently executable computer program designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." Dec. 11–12; Prelim. Resp. 9.

Patent Owner contends that it agrees with our interpretation of an "application program" as an independently executable program, but interprets "independently executable program"[2] in a manner that excludes programs that do not have certain attributes. PO Resp. 8–15. Based on the background section of the '854 patent that refers to retrieval of information from sources external to a word processor, such as a database or contact management program, Patent Owner asserts that the claimed invention is limited to obtaining information from an information management program that can be used separately and independently from the word processor. PO Resp. 11 (citing Ex. 1001, 1:34–37, 1:45–46).

Patent Owner's interpretation of "application program" is based on the commonly shared features of the example programs from the

[2] Patent Owner suggests that the Decision to Institute's use of the term "independent" rather than "independently" in construing "'application program' to encompass an independent executable program" (Dec. 11) was a typographical error. PO Resp. 9–10. Patent Owner does not explain how "independent" differs from "independently" and defines the term "independent" as part of its analysis. *Id*. at 10. For purposes of this Decision, we address Patent Owner's contentions as if they apply to both "independent" and "independently."

specification. PO Resp. 13 ("Given that the specification identifies word processors, spreadsheet programs, information management programs and database programs as examples of application programs, the definition of an 'application program' can be construed from the commonly shared features."). Patent Owner relies on the Declaration of Dr. John Levy (Ex. 2003), paragraphs 42–43, to support its interpretation that subsidiary programs, which extend the functionality of the controlling application, are not "independently executable computer programs" as recited in the claims. PO Resp. 14–15.

We disagree with Patent Owner's narrow interpretation. The term "application program" does not appear in the specification of the '854 patent. However, we are not persuaded that the term is limited by the commonly shared features of the examples in the '854 patent specification. *See Van Geuns*, 988 F.2d at 1184 ("[L]imitations are not to be read into the claims from the specification."). Patent Owner has not provided sufficient evidence to limit "application program" to programs that are not under the control of another program or run synchronously under the control of a separate application program (PO Resp. 13–14). We do not find Dr. Levy's testimony persuasive that the broadest reasonable interpretation of "application program" by one of ordinary skill in the art at the time of the invention is defined by "commonly shared features" of examples of computer programs in the patent specification. *See* PO Resp. 11–15 (citing Ex. 2003 ¶¶ 42–44); 25 (citing Ex. 2003 ¶¶ 18, 42–48)). Construing application program as Patent Owner suggests improperly limits the claim term to the embodiments and examples in the '854 patent specification and imports negative limitations unsupported by the intrinsic evidence.

Patent Owner has not shown that the broadest reasonable construction of "application program" excludes subsidiary programs. *See* PO Resp. 11–15; Reply 6–9. On the complete record, we determine that "application program" is construed as an independent executable program.

B. Unpatentability Based on Domini (Ex. 1006)

Petitioner contends that claims 19, 20, 22–26, 28–30, 57, 58, 60–74, 76–78, 85, and 96 are anticipated by Domini. Pet. 30–39. Petitioner's claim chart provides citation to Domini, which Petitioner contends disclose the corresponding claim limitations in claims 19, 20, 22–26, 28–30, 57, 58, 60–74, 76–78, 85, and 96. *Id*.

1. Overview of Domini (Ex. 1006)

Domini discloses identifying and correcting spelling and grammar errors in a document created by a word processing program. Ex. 1006, Abstract, 4:65–5:11. Figure 1, below, shows an embodiment of the Domini invention. *Id*. at 4:39–41.

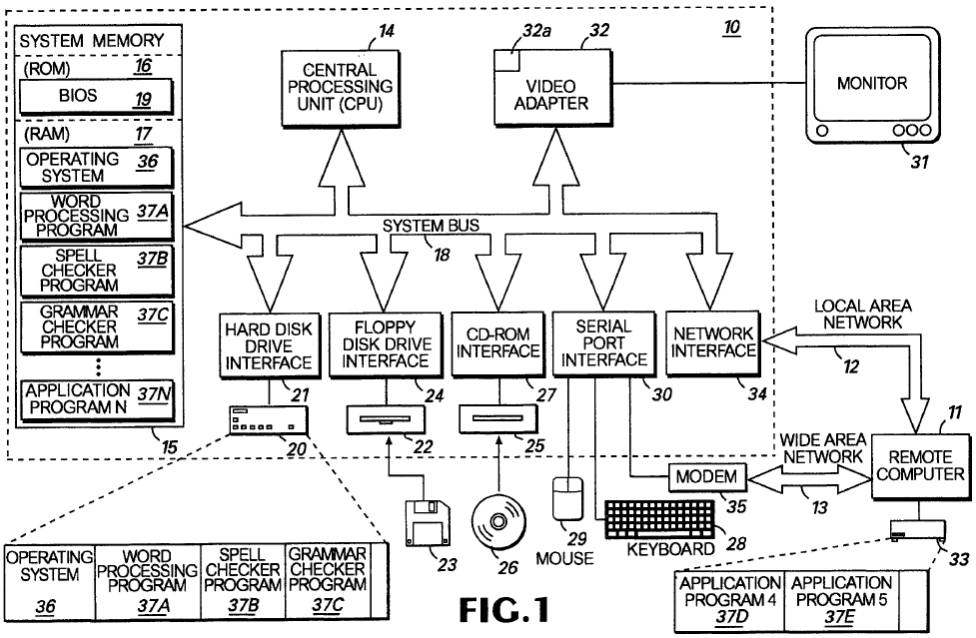


FIG.1

Figure 1 depicts personal computer 10 connected by networks 12 and 13 to remote computer 11. *Id*. at 7:13–16. Domini discloses that "[t]hose skilled in the art will understand that program modules such as an operating system 36, application programs 37, and data are provided to the personal computer 10." *Id*. at 7:41–43. Thus, personal computer 10 and remote computer 11 contain program modules, such as operating system 36 and application programs 37. *Id*. at 6:33–42, 7:41–44. Domini states further that:

> [t]he *application programs* 37 may include a number of different programs such as a word processing program 37a, a *spell checker program **37**b*, and a grammar checker program 37c. In the preferred personal computer 10, the local hard disk drive 20 is used to store data and programs, including the operating system and programs.

Id. at 7:46–52 (italics added).

In the spelling and grammar programs disclosed in Domini, the user selects the "[s]pelling and [g]rammar" command to initialize the spell check program. *Id*. at 16:13–16. Without user intervention, the spell check program identifies misspelled words and presents them in red, bold typeface. *Id*. at 17:27–33, 4:12–16. The spell check program also displays a list of suggested corrections that may be selected and entered into the document by the user. *Id*. at 1:42–44, 12:1–5, 12:61–64.

2. Anticipation based on Domini (Ex. 1006)

a. Application Programs

Patent Owner argues that Domini fails to teach the "second application program" as recited in each of the challenged independent claims, because the spell checker described in Domini is a "module" that operates inside of a word processing document and not an "application

program" that is an "independent executable program" as the term is construed. PO Resp. 23–24.

Patent Owner's arguments and analysis are based on its proposed claim interpretation that excludes subsidiary programs from "application program" as recited in the challenged claims. PO Resp. 25 (contrasting subsidiary programs with application programs and stating that it "would be understood by one of ordinary skill in the art at the time of the invention, an 'application program' is an 'independently executable program' that is independent of and not under the control of another program." (citing Ex. 2003 ¶¶ 42–48)). As discussed above, we do not construe "application program" to exclude subsidiary programs based on characteristics of the example programs described in the '854 patent specification. *See* Section II.A.4, *supra*.

We also are not persuaded by Patent Owner's arguments and testimony of Dr. Levy that the program modules in Domini that are explicitly identified as "application programs" do not meet the claim limitation for "application program." PO Resp. 25–29. Dr. Levy's narrow interpretation of application program reads limitations into the claim (PO Resp. 25) that are not supported by the '854 specification. *See* Reply 8–9 (citing Ex. 1001, 7:25–8:7).

Patent Owner's argument that one of ordinary skill in the art would have understood spell checker program 37b to be a program module (PO Resp. 27–28) and not an application program—as it is expressly described—is not supported by a plain reading of the Domini disclosure. Patent Owner's arguments and evidence fail to address the term "application program" as it is used in Domini and show that it differs from the

"application program" as recited in the challenged claims. Indeed, Dr. Levy has not provided testimony that the term "application program" as used in Domini (Ex. 1006, 7:41–52, Fig. 1) differs from or is inconsistent with "application program" as recited in the challenged claims. *See* Tr. 35:13–20 (stating that Patent Owner's expert found Domini consistent with Patent Owner's construction).

With respect to the "application programs" in Domini, Patent Owner concedes that Domini discloses stand-alone spell checkers (PO Resp. 29–31 (citing Ex. 1006, 1:56–2:26; Ex. 2003 ¶ 24)), but argues that "a stand-alone spell checker would not be capable of inserting text into a word processor" (PO Resp. 30 (citing Ex. 2003 ¶¶ 24–25, 35)). We agree with Petitioner (Reply 11), however, that Domini discloses incorporating changes into a document by replacing words in the word processing document. Ex. 1006, 12:59–13:31, 14:42–67.

We disagree with Patent Owner's narrow interpretation of the term "application program" that excludes spell checker program 37b explicitly disclosed in Domini (Ex. 1006, 7:46–52). PO Resp. 25–29. Instead, we find that spell checker program 37b and other application programs in Domini disclose the "application program" recited in the challenged claims. Based on the complete record and in light of Patent Owner's and Petitioner's arguments and evidence, we find, by a preponderance of the evidence, that Domini discloses an "application program" as recited in the challenged claims.

b. "the second information associated with the first information from the second application program" and "operation related to a second information" of "entering additional data into a database"

Patent Owner's argument that Domini fails to disclose "the second information associated with the first information from the second application program" is based on the erroneous claim construction that "associated" requires a pre-existing relationship between the first and second information. PO Resp. 31–34. Because we determined previously that "associated" is construed as "connected or related" (*see* Section II.A.3, *supra*), we are not persuaded by Patent Owner's argument. Patent Owner's argument that Domini does not disclose a pre-existing relationship, "such as the relationship between field entries of a database record" (PO Resp. 32) is premised on an overly narrow interpretation of the claim term that is not commensurate in scope with the challenged claims.

We also are not persuaded by Patent Owner's contention that Domini fails to disclose the "operation related to a second information" of "entering additional data into a database" limitations of dependent claims 22–24, 28–30, 60–62, and 76–78. PO Resp. 34–38. Patent Owner argues that "Domini's act of adding a (misspelled) word from the document (first information) into the dictionary is not an operation related to second information. Rather, it is an operation relating to the first information." PO Resp. 37 (emphasis omitted).

We disagree. Patent Owner's contention assumes erroneously that the first and second information are not related and that "an operation related to the second information" is limited to actions involving only the second information. However, Patent Owner has not shown that "performing an

operation related to second information," includes operations on pre-existing information or new information that may be the second information itself or related to the second information. *See* Section II.A.2, *supra*. Thus, the operation could be on new information that is related to the second information.

We find that the act of adding a related spelling of a word, such as a plural form or alternate spelling of a word, to a dictionary or database is an operation related to the second information as recited in the claims. *See, e.g.*, Reply 12–13 (discussing "neighbo**u**r" and "neighbor" as alternate spellings that are related first and second information); Pet. 32 (citing Ex. 1006, Fig. 3, 5, 7, 12:1–5, 12:61–64). In addition, the act of adding the second information (e.g., alternate spelling) to the document is also an "operation related to a second information." Thus, we find that Domini's disclosure regarding the storing of first information in the dictionary is an operation related to the second information. *See* Pet. 31–32.

Accordingly, we do not agree with Patent Owner that Domini fails to disclose "an operation related to a second information" as required in dependent claims 22–24, 28–30, 60–62, and 76–78. Based on the full record, we find that Petitioner has shown by a preponderance of the evidence that Domini discloses "the second information associated with the first information from the second application program" as recited in independent claims 19, 25, 57, 73, 85 and 96; and an "operation related to a second information" of "entering additional data into a database" as recited in dependent claims 22–24, 28–30, 60–62, and 76–78.

c. "searching, using the second application program, for the second information associated with the first information"

Patent Owner contends that that Domini fails to disclose "searching, using the second application program, for the second information associated with the first information" as recited in independent claims 85 and 96 and dependent claims 64–69. PO Resp. 38–43. Patent Owner's argument is that Domini searches for the individual words in the document but does not search for other information, such as "second information." PO Resp. 40. In the context of Domini's spell checker program, Patent Owner argues that Domini only determines whether the word is correctly spelled but is not looking for second information.

We disagree with Patent Owner. Although Patent Owner acknowledges that Domini provides suggested words, it argues that these words are not searched for but instead are located in the Spell Return Buffer. PO Resp. 42–43. We find that that Domini provides suggested words obtained from the Spell Return Buffer as suggested corrections. Pet. 30 (citing Ex. 1006, Fig. 3, 1:42–44, 12:1–5). Patent Owner's argument that Domini does not describe explicitly searching for the suggested spelling corrections ignores the fact that suggested corrections are retrieved and displayed as part of the Domini spell checker process. Ex. 1006, Fig. 3, 1:42–44, 12:1–5. We also agree with Petitioner's argument that the spell check program has one or more dictionaries from which suggested corrections are obtained. Pet. 35. In addition, Petitioner's expert, Daniel A. Menascé, Ph.D, testified persuasively that the Domini spell checker retrieves possible words that are related to the words that a user types. *See* Ex. 1012, Deposition of Daniel A. Menascé, Ph.D, 127:3–130:5.

Based on the complete record and the evidence and arguments presented by Petitioner and Patent Owner, we find, by a preponderance of the evidence, that Domini discloses "searching, using the second application program, for the second information associated with the first information" as recited in independent claims 85 and 96 and dependent claims 64–69.

d. Conclusion

We have considered the evidence presented by Petitioner and Patent Owner. On the full record, we find that Petitioner has shown, by a preponderance of the evidence, that Domini anticipates claims 19, 20, 22–26, 28–30, 57, 58, 60–74, 76–78, 85, and 96.

C. Patent Owner's Motion to Exclude Evidence

Patent Owner moves to exclude the deposition transcript of Petitioner's expert, Dr. Daniel A. Menascé, Exhibit 1012. Paper 22. Dr. Menascé was deposed by Patent Owner's counsel on August 7, 2014 for this proceeding and for proceedings IPR2014-00207 and IPR2014-00208. Ex. 1012, 1. Patent Owner contends the Menascé transcript is improper supplemental information that is not submitted in compliance with 37 C.F.R. § 42.123 and re-uses a previously used Exhibit number and should be excluded. Paper 22, 2.

With respect to the mis-numbered Exhibit, we deny Patent Owner's motion to exclude the Exhibit for failing to meet numbering requirements of 37 C.F.R. § 62.63(c).

With respect to the transcript being improper supplemental information, Petitioner argues and we agree that 37 C.F.R. § 42.53(f)(7) states that deposition testimony must be filed by its proponent as an exhibit.

Paper 26, 3–4. Consistent with Petitioner's position, the rule recently has been clarified. *See* Amendments to the Rules of Practice for Trials Before the Patent Trial and Appeal Board, 80 Fed. Reg. 28,561, 28,563 (May 19, 2015) ("To clarify that either party is permitted to file testimony as an exhibit, the Office amends 37 CFR 42.53(f)(7) to delete the phrase 'by proponent' in the second sentence."). Because either party is permitted to file testimony as an exhibit, Petitioner's filing of the exhibit is proper. Accordingly, we *deny* Patent Owner's motion to exclude Exhibit 1012, the deposition transcript of Petitioner's expert, Dr. Daniel A. Menascé.

Patent Owner also seeks to exclude "Exhibit 1015," Visual Studio 2012 ("VS2012"), filed as an attachment to the deposition transcript of John V. Levy, Ph.D. (Ex. 1011).[3] Paper 22, 10. Patent Owner implicitly acknowledges that Petitioner's Reply to Patent Owner's Response does not expressly discuss or rely on VS2012. Paper 22, 11. Because we do not consider or rely on VS2012, or the portion of Dr. Levy's testimony discussing VS2012, in reaching our determinations in this Decision, Patent Owner's motion to exclude "Exhibit 1015" of Exhibit 1011 is *dismissed* as moot.

III. CONCLUSION

Petitioner has demonstrated, by a preponderance of the evidence, that claims 19, 20, 22–26, 28–30, 57, 58, 60–74, 76–78, 85, and 96 are unpatentable as anticipated by Domini.

[3] Although the attachment is labelled "Exhibit 1015," VS2012 was not entered into the file as Exhibit 1015.

IV. ORDER

For the reasons given, it is

ORDERED that, based on a preponderance of the evidence, claims 19, 20, 22–26, 28–30, 57, 58, 60–74, 76–78, 85, and 96 of U.S. Patent No. 7,496,854 B2 are held unpatentable; and

FURTHER ORDERED that Patent Owner's motion to exclude "Exhibit 1015" is *dismissed*, and the motion to exclude Exhibit 1012, the deposition transcript of Dr. Daniel A. Menascé is *denied*; and

FURTHER ORDERED that, because this is a Final Written Decision, parties to this proceeding seeking judicial review of our Decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

For Petitioner:

David L. Fehrman
dfehrman@mofo.com

Mehran Arjomand
marjomand@mofo.com

Matthew A. Smith
smith@turnerboyd.com

Zhuanjia Glu
gu@turnerboyd.com

For Patent Owner:

Robert Asher
rasher@sunsteinlaw.com

Bruce Sunstein
bsunstein@sunsteinlaw.com

Exhibit 6Q

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Apple Inc., Google Inc., and Motorola Mobility LLC

Petitioners,

v.

Arendi S.A.R.L.

Patent Owner.

Case No. IPR2014-00207

Patent No. 7,496,854

PATENT OWNER ARENDI S.A.R.L.'S PRELIMINARY RESPONSE UNDER 35 U.S.C. § 313 and 37 C.F.R. § 42.107

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Cases

Statutes

EXHIBIT LIST

Arendi Exhibit Number	Description
2001	American Heritage College dictionary 3rd edition 1997 definition of the term "dictionary."

I. INTRODUCTION

Patent Owner Arendi S.A.R.L. ("Arendi" or "Patent Owner") respectfully requests that the Board decline to initiate *inter partes* review of claims 1-18, 36-56, 86-95, 97, 98, 100 and 101 of U.S. Patent No. 7,496,854 (the "'854 Patent") because Petitioners Apple Inc., Google Inc., and Motorola Mobility LLC ("Petitioners") has failed to show that it has a reasonable likelihood of prevailing with respect to any of the challenged claims. 35 U.S.C. § 314.

Petitioners have submitted proposed grounds for challenge based on anticipation or obviousness. However, for each proposed ground, at least one claim element is missing from the relied-upon reference or combination of references. Thus, Petitioners has failed to meet its initial burden to show that each element was known in the prior art.

II. OVERVIEW OF THE '854 PATENT

The '854 Patent is directed, among other things, to computer-implemented processes for automating a user's interaction between a first application, such as a word processing application or spreadsheet application, on the one hand, and a second application, such as contact management application having a database, on the other hand.

The invention as claimed provides a significant simplification of prior art methods. In the prior art, a user who has entered first information (e.g. a person's

name) in a document must leave the first application program, (e.g. a word processor) and open and search using a second application program (e.g. contact management program) when the user wishes to locate second information related to the first information from the second application program. The user of such prior art systems must search for the first information (e.g. a name) and the second information (e.g. an address) using the second program and then return to the first program and manually enter the second information into the document. This process requires a plurality of actions by the user in order to obtain related information to the information typed within the document. Thus, according to embodiments of the present invention, "the process of creating and updating records in an address database is significantly simplified, since this may now be performed directly from the word processor."

In the '854 Patent, Exhibit 1001, Figs. 1 and 2 are flow charts showing for these interactions a number of scenarios, which are described from col. 4, line 25 to col. 5, line 57. Further details of the interactions are provided in discussion thereafter of the other figures of the '854 Patent and the discussion includes references back to relevant portions of the flow charts in Figs. 1 and 2. Fig. 1 is reproduced below.

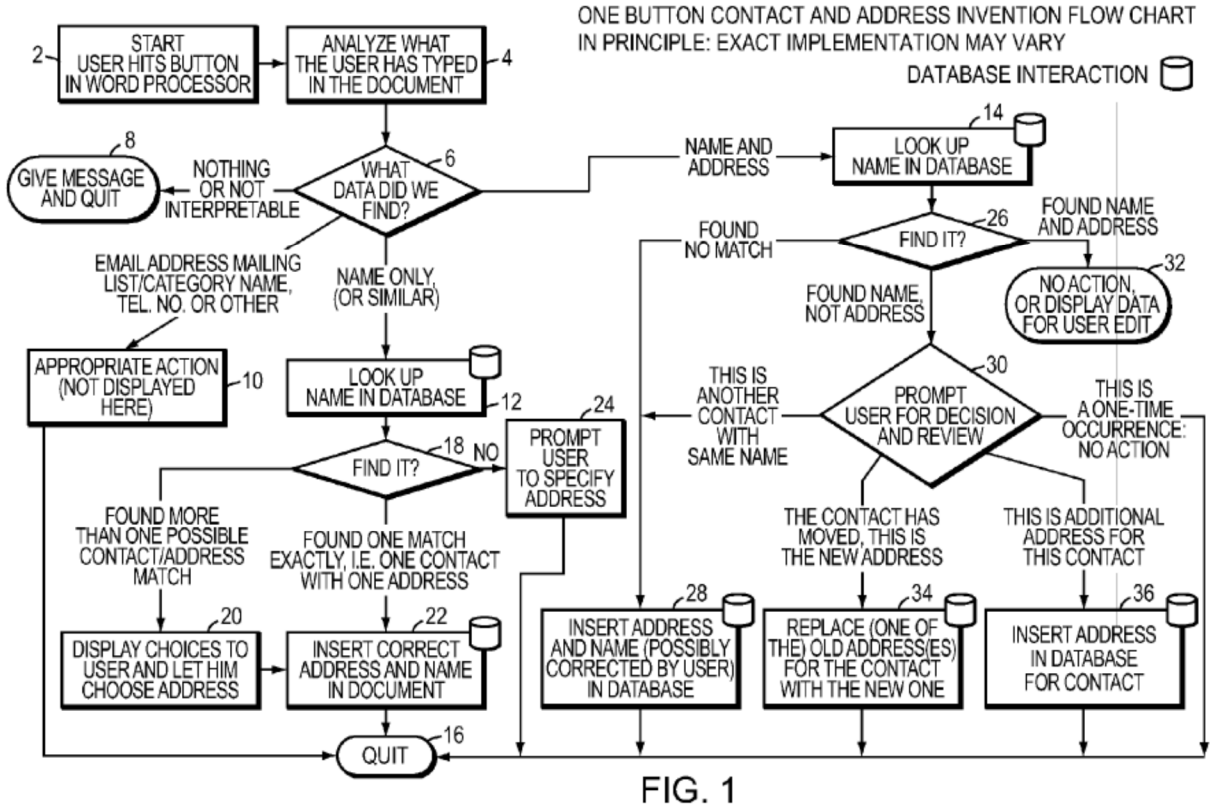


FIG. 1

In various scenarios, text in a document in the first application is analyzed (in step 2 of Fig. 1) to identify first information. Exhibit 1001, col. 4, line 25-49. The analysis takes place without user designation of a specific part of the document to be subject to the analyzing. *Id*.

Once first information has been identified, a number of different scenarios can follow, depending on the circumstances. In one scenario, if the first information includes a name, a search is initiated in the database associated with the second application for the name. *Id*., Fig. 1, steps 6, 12, and 14. If the contact information identified in the document included only a name, and if only a single entry is found in the database for the name and the entry includes a single address,

then the address is inserted into the document. *Id*., Fig. 1, steps 6, 12, 18, and 22; Fig. 4; col. 5, lines 63 to col. 6 line 3. Fig. 4, which is reproduced below, shows the document displayed in Microsoft Word after the address has been inserted.

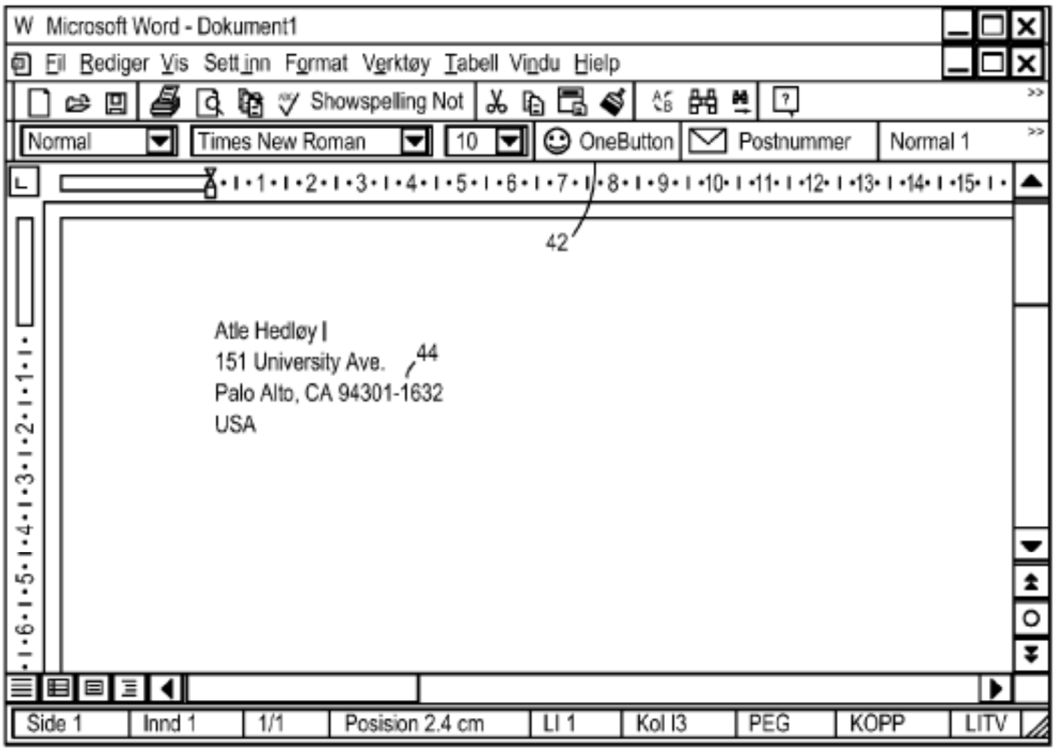


FIG. 4

Shown in Fig. 4 is the One Button 42, which, when pressed, launches the processes just recited, including analyzing the document to identify first information, the searching in the database, and inserting of the address. *Id*., Fig. 2, steps 2, 4; col. 4, lines 25-28; col. 5, lines 58 to col. 6 line 5.

On the other hand, if multiple addresses are found in searching the database for the identified name, these found addresses are displayed, and the user is

presented with a choice of which of the addresses to insert. *Id*., Fig. 1, steps 18, 20, and 22; Fig. 10; col. 7, line 25 to col. 8, line 7.

In another scenario, when the user clicks on the "One Button" after having typed into the document a name and an address, the document is analyzed as before (per Fig. 1, step 4) to identify the name and the address. Next, the database is searched for the identified name (per Fig. 1, step 14). If the name happens to be in the contact database but the address in the contact database for that name differs from the address typed by the user into the document (per Fig. 1, step 26), then the user is prompted to make a choice (per Fig. 1, step 30). The user is presented with

a screen shown in Fig. 9, which is reproduced below.

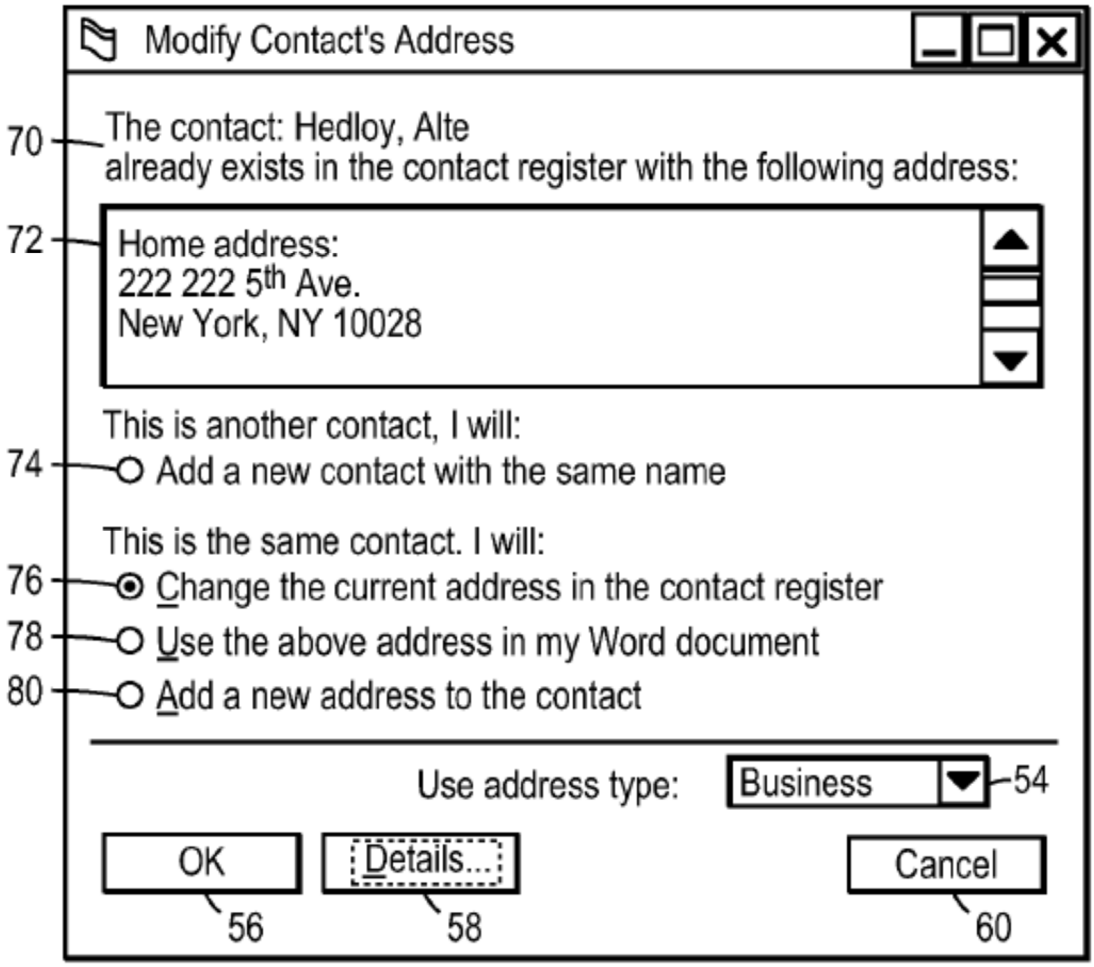


FIG. 9

Fig. 9 represents a screen presented to the user in which the user is given a series choices that can be made in this specific context. *Id.*, Col. 6 line 66 to col. 7 lines 23. The screen reproduces the name that is both in the document and in the contact database, and it also displays the address that is in the contact database for that name. Below this information, the screen offers a total of four choices in two categories. As shown in Fig. 9 and explained in the '854 Patent, the user is

enabled to select one of the four choices. *Id*. The first category is that "This is another contact" and the choice under this category is to "Add a new contact with the same name". The second category is that "This is the same contact", and the user is given three other choices for the contact: (a) "Change the current address in the contact register"; (b) "Use the above address [reproduced from the contact database] in my Word document"; and (c) "Add a new address to the contact".

These same four choices are also illustrated in connection with item 30 of Fig. 1 of the '854 Patent, which shows logical flow followed in described embodiments of the invention. Item 30 is labeled "PROMPT USER FOR DECISION AND REVIEW", and there are four outcomes shown from this item: (1) "THIS ANOTHER CONTACT WITH THE SAME NAME"; (2) "THE CONTACT HAS MOVED, THIS IS THE NEW ADDRESS"; (3) "THIS IS A ONE-TIME OCCURRENCE: NO ACTION"; and (4) "THIS IS ADDITIONAL ADDRESS FOR THIS CONTACT". These choices are described in the '854 Patent, col. 4, line 62-col. 6 line 8.

It can be seen that the first of the four choices is to add a new contact, and two of the remaining choices are specific ways of updating an existing contact. (Another choice offered is to do neither of these and simply use the address in the Word document as typed.) Consequently, the screen of Fig. 9 presents to the user

a choice, among other things, between competing alternatives of storing a new contact or updating an existing contact.

III. CLAIM CONSTRUCTION

In an *inter partes* review, the Patent Trial and Appeal Board gives patent claims their "broadest reasonable interpretation in light of the specification of the patent." 37 C.F.R. § 42.100(b); *Phillips v. AWH Corp.*, 415 F.3d 1303, 1316 (Fed. Cir. 2005) (*en banc*). The prosecution history is also relevant to identify the correct construction of claim terms. *Phillips v. AWH Corp.*, 415 F.3d at 1317. Extrinsic evidence may also be relevant to establish the meaning of terms, but such evidence is only relevant to the extent it is consistent with the specification and file history. *Id.*, 1319.

Patent Owner Arendi proposes construction of certain claim terms below pursuant to the broadest reasonable interpretation consistent with the specification standard. The proposed claim constructions are offered for the sole purpose of this proceeding and thus do not necessarily reflect appropriate claim constructions to be used in litigation and other proceedings wherein a different claim construction standard applies.

A. Associated

The term "associated" should be construed in accordance with its ordinary and customary meaning and the clear usage of the term within the intrinsic

evidence as "a pre-existing connection or relationship." The patent refers to the searching of a database for additional contact information (e.g. physical and e-mail addresses, phone numbers) that is "related" to text, identified in a word document (e.g. a name), that is in a shared entry in the database. See for example the Abstract, col. 3 lines 63-66, col. 5 line 66, col. 6 line 2, col.4 lines 43-45, 57-58.

B. Application Program

The term "application program" should be construed in accordance with its ordinary and customary meaning and the clear usage of the term within the intrinsic evidence as "an independently executable computer program designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." The patentee has used the term "application program" to refer to word processors, spreadsheet applications and contact managers within the specification such as Microsoft Word, Microsoft EXCEL and Microsoft Outlook. See col. 8 lines 30-33 and 57-67. See also Fig. 1-3 that show Microsoft Word and Example 7 beginning at Col. 8 line 55 entitled "Spreadsheet Application" that discloses using Microsoft EXCEL.

This definition is consistent with the prior art of the time including the cited Hachamovitch patent (Exhibit 1008) relied upon by the Petitioners. As shown in Fig. 1 (produced below), Hachamovitch contemplates a distinction between Application Programs 36, Program Modules 37, and auto-complete utility 100 (i.e.

specific program module). See col. 9 lines 3-5. When using the term "application program," Hachamovitch refers to "a word processing application program or an e-mail application program." See col. 7 lines 62-64. In contrast, Hachamovitch states "program modules include routines, programs, components, data structures etc. that perform particular tasks or implement particular abstract data types." col. 8 lines 27-29. Thus, the prior art is consistent with the Patent Owner's interpretation of the term "application program."

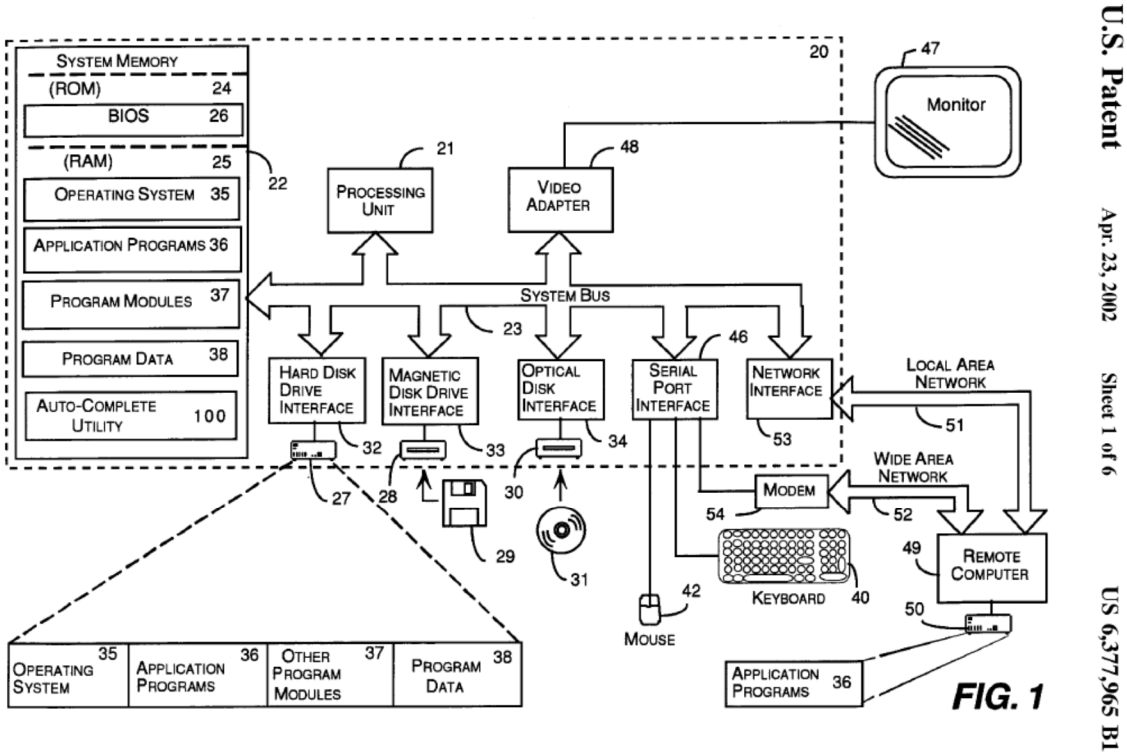


FIG. 1

C. Related Information

The term "related" should be construed in accordance with its ordinary and customary meaning and the clear usage of the term within the intrinsic evidence as

"a pre-existing association." Thus, related information would be information that has a pre-existing association. The Patentee has used the term to refer to associated data that can be searched for in a database, such as related fields for a database entry.

IV. OVERVIEW OF THE PRIOR ART

A. Overview LiveDoc

LiveDoc addresses structure detection within a document where a structure represents meaningful bits of syntactically- regular information. LiveDoc allows a user to perform a function based upon an identified structure. To accomplish this goal LiveDoc constructs "a means of passing text from a user's document for matching against a collection of recognizers." Exhibit 1005 at page 53. Thus, LiveDoc operates outside of any application program and outside of the document under the control of the application program.

The LiveDoc architecture is shown in Fig. 3 at page 56 where the LiveDoc manager communicates with an external application (i.e. a text editor) using API callbacks. See *Id.* at page 57 left col.

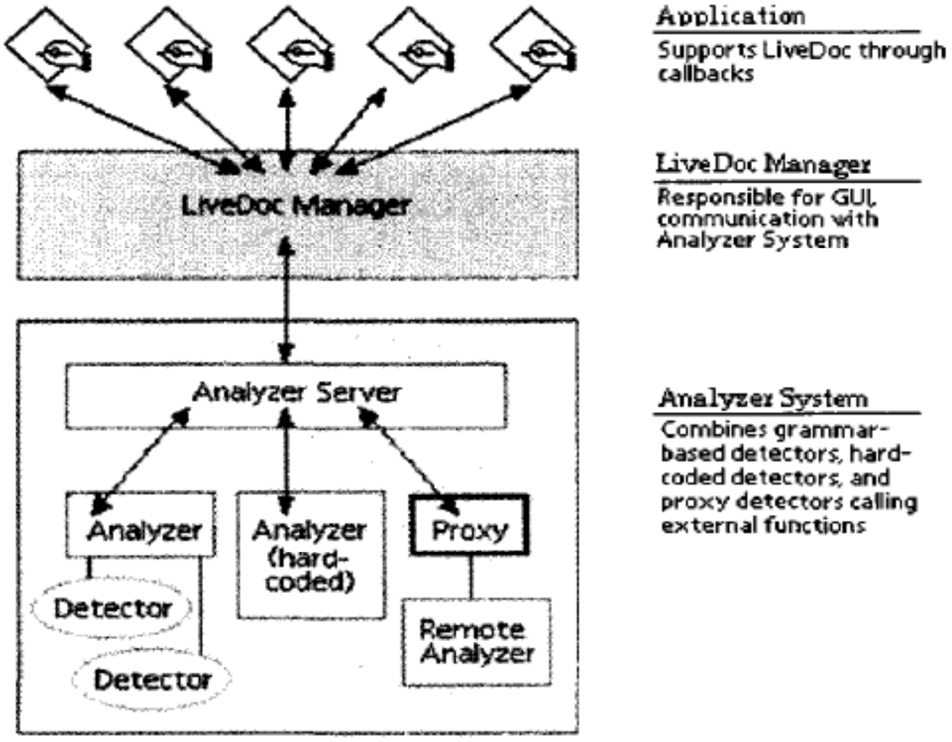


Figure 3: The high-level LiveDoc architecture

The LiveDoc application receives only the text from the text editor ("Application" in Fig.3) and analyzes the text independently of the actual document in the text editor using a set of detectors under the control of an analyzer server.

In order for the discovered structures to be visible to a user, the user must enter "LiveDoc mode" by pressing the function key causing the LiveDoc Manager to update "the display to present the highlight information over the discovered structures." *Id.* at page 56. The user can then use the mouse to move over a highlighted item and press the mouse button that causes the LiveDoc Manager to present a menu of functions associated with the highlighted item.

> LiveDoc knows where these structures appear in the text passed to it- an e-mail address might appear in characters 150 through 162 of the window's contents – but it has no idea where in the window those characters physically appear, and, thus, where the highlights should appear: this is information held by the application, not by LiveDoc. Hence, LiveDoc must ask the application for the information about the structures it has found via a callback. Once this information is available, the highlights and their associated mouse-sensitive regions can be constructed.

These passages show that the overlaid highlights are independent of and separate from the text editor and the document. Fig. 2 shown below show some of the actions that LiveDoc allows for a recognized structure.

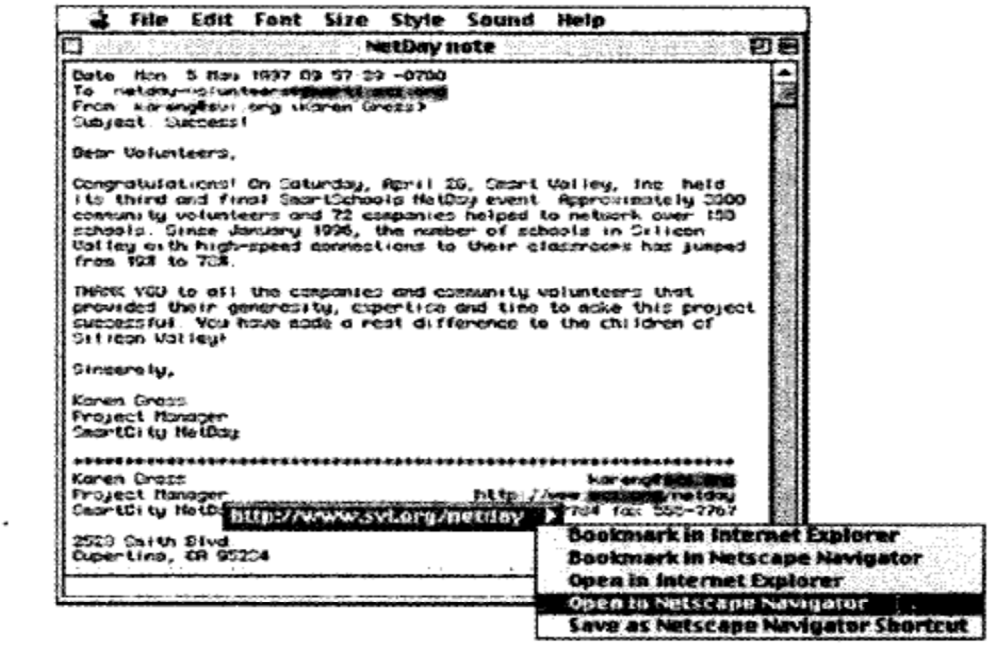


Figure 2: A sample interaction with LiveDoc. Note the highlighting of the discovered structures, the menu of actions available or the selected structure, and the nested highlighting of nested structures.

Each of the functions shown involves using the recognized text with an external application.

> "Our initial implementation of LiveDoc as LiveSimpleText assumed that actions would be handled by external applications, such as a Web browser presenting the page pointed to by a URL:" *Id.* at 57.

B. Overview of Drop Zones

Drop Zones uses the LiveDoc system, so that, when a user has entered LiveDoc mode, Drop Zones presents recommended appropriate actions based on the selected structure(s). For example, a user may use a mouse to select a

highlighted structure. Drop Zones recognizes the structure (e.g. as a name) and highlights any of the assistant function that can operate on the identified structure (e.g. a name can be used with an e-mail assistant). When the selected structure is dragged and dropped onto an assistant function, then a set of possible actions is presented to the user for using the selected structure. "DropZones goes beyond LiveDoc in allowing the user to select some subset of those terms and drag them as a group" to be operated on by the assistant. *Id.* at 62.

C. Overview of Moore

The Patent to Moore teaches a database program that includes an association between chemical formulas and graphical representations of the formulas. Moore provides a "complete chemical information system" that allows for searching based on structure and substructure, chemical names, molecular formulas.

D. Overview of Domini

Domini is directed to a combined spell checking and grammar checking module that operates within a word processing application. See, e.g., Ex. 1007 Abstract, col. 3 lines 1-12, col. 5 lines 1-8. Fig. 3 reproduced below shows a dialog box for the spell checking and grammar checking module after a user has selected the "spelling and grammar" command within the application program (e.g. word processing application).

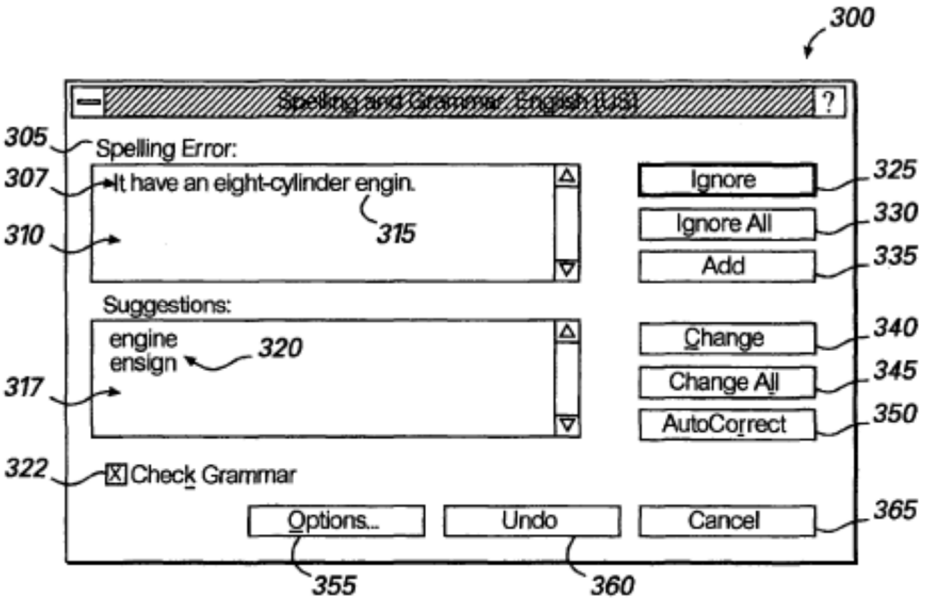


FIG. 3

Upon selection of the spelling and grammar command within the word processing application, a sentence is extracted from the word processing document and the spell checking program module is called. *Id.* at col. 16 line 56-66. The spell checking program module extracts each word from the sentence and verifies whether the words appears in the one or more dictionaries of the spell checking program module. *Id.* at col. 17 lines 19-42 If the word does not appear within any dictionary, an error is generated indicating that the word is potentially misspelled and the word processing application accesses a string buffer that contains one or more suggestions for the potentially misspelled word. The suggestions are displayed within a combined spelling and grammar checking dialog box. *Id.* at col.

18 line 4-20. A user may then select between the one or more suggestions to replace the word within the document. *Id.* at col. 18 lines 21-26

E. Overview of Hachamovitch

Hachamovitch describes a word completion utility that can be used in conjunction either with an individual application program or with a group of different application programs. Exhibit 1008 at col. 4 lines 10-28. The word completion utility is executed from within the application program. Exemplary embodiments of the word completion utility are incorporated into the MICROSOFT OFFICE 97 Suite. *Id.* at col. 9 lines 44-45. The word completion utility can predict word completions for data entries in an unstructured portion of a data file, such as a word processing document or an e-mail within either a word processor application program or an e-mail application program. The suggested word completions may be based on a suggestion list that includes name-completion pairs. The partially typed word is compared to the name-completion pairs and if a match is found within the list a suggestion list will be presented to a user. *Id.* at col. 4 line 58- col. 5 line 6. The suggestion list may be presented to the user in a pop-up user interface within a word processing application as shown in Fig. 2A. The suggestion is displayed to a user and the user may accept the word completion using an acceptance keystroke (e.g. tab or enter). Once accepted the word

completion utility replaces the partial data entry with the completion entry in the data file. *Id.* at col. 5 lines 7-10.

As shown in Fig. 2A a user begins to type "Jun" and the word completion utility identifies this partial entry as being representative of the current date. Thus, the current date is presented in the pop-up user interface and the current date can be selected by the user and inserted into the data file by the word completion utility.

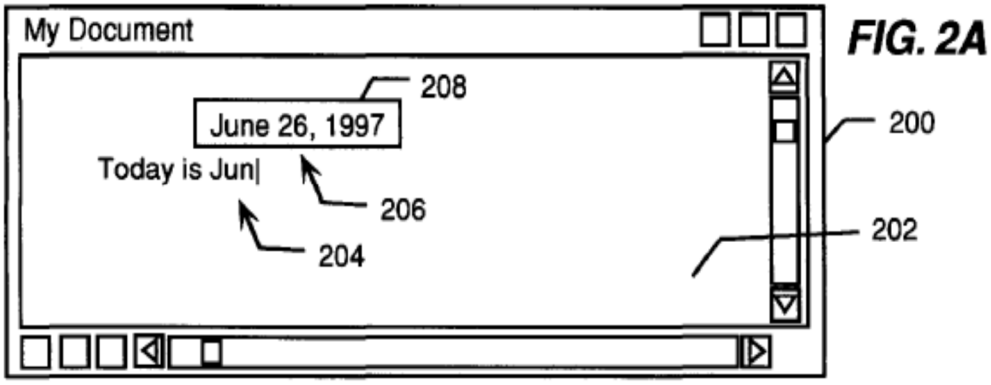


FIG. 2A

F. Overview of Luciw

Luciw describes logical processes, usable by a pen-based computer system that functions as a personal organizer, to provide "implicit or explicit assistance" for "user supportive information functions". Luciw, Exhibit 1003, col. 4, lines 14-18 (pen-based computer system); col. 2, lines 16-19 (implicit or explicit assistance).

The pen-based computer system has a database that can be queried. *Id.* at col. 8, lines 31-34. Luciw describes "implicit" assistance, wherein a user has used

a smart field to enter a word used for look up in the database or has otherwise similarly triggered a database lookup, and "explicit" assistance, wherein the user explicitly invokes assistance from the device as by using pen 38 of Fig. 2. See *Id.* at col. 8, lines 11-62.

The logical processes used by the Luciw device for providing implicit and explicit assistance are shown in Fig. 3 of Luciw. *Id.,* col. 8, lines 2-6. A review of Fig. 3 shows that the database is queried in step 106 if it is determined in step 104 that there is an implicit assist. On the other hand, if in step 104 it is determined that there is not an implicit assist, and if further it is determined that there is an explicit assist, there is no database query, because the only database query indicated is in step 106, exclusively where there is an implicit assist.

As an example of implicit assist, Luciw provides Figs. 4b, 4c, 5, 6a and 6b, which describe use of a "smart field". *Id.,* col 10, line 23 et seq. (beginning discussion of smart fields in connection with Fig. 4b). According to Luciw, "[a] smart field is considered to be a predefined region on screen 52 of computer system 10 shown in FIG. 2, or a predefined region within a window which appears on screen 52". Id. col. 8, lines 16-19. Fig. 4b is reproduced below.

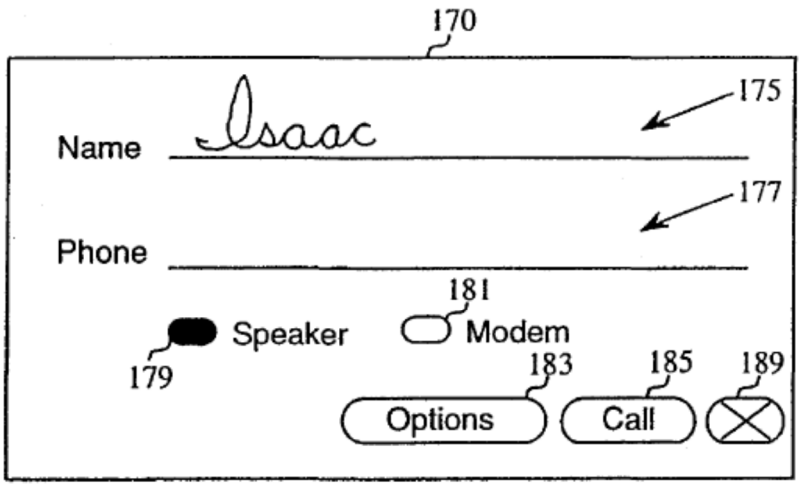


Figure 4b

According to Luciw, Fig. 4b "shows a phone slip window 170 with a smart name field 175 which has for example been evoked by either highlighting the verb 'call' or by simply writing the word on the display surface either before or after establishment of window 170." *Id.*, col 10, lines 24-28. Operation of the phone slip window is explained in the lines thereafter in Luciw:

> Once the particular window 170 is presented to the user, the name ISAAC can be handwritten into the particular smart field 175. The assistance process recognizes the handwritten name "Isaac," and either continues operation as suggested at step 106 in FIG. 3 directly, or concurrently displays the recognized name in formal font form, as suggested in FIG. 4c, in the same position of the smart field, where formerly the handwritten name "Isaac" had been established. As will readily be recognized, window 170 in FIG. 4b may contain several smart fields, in this case for example definable for either the "name" field 175 or a "phone" field shown at step 177.

Id., col. 10, lines 27-39.

Because the user of the Luciw device uses the smart field to specify the field for which a database search is desired—a name in the name field 175 or a phone number in the phone field 177—the Luciw device can then use the entered item to search for in the database for an item that has the same value for a corresponding attribute. Id., col. 10, line 51 to col. 12, line 11.

V. SINCE THE PRIOR ART DOES NOT ANTICIPATE OR RENDER ANY CLAIM OBVIOUS, NO *INTER PARTES* REVIEW SHOULD BE INITIATED

A. Overview of Reasons for Denying Inter Partes Review

For each ground asserted in the present Petition [IPR2014-00207] the reference or references relied upon fail to teach at least one element of the claims to which those references are applied.

Consequently, the grounds asserted fail to anticipate any claim under 35 U.S.C. §102(a) or 35 U.S.C. §102(e) as alleged, and fail to rending any claim obvious under 35 U.S.C. §103(a) as alleged.

The first ground for review asserted by the Petitioners relies on a combination of two articles, LiveDoc and Drop Zones, as a basis for arguing anticipation. This combination is improper, since anticipation requires that all claim elements be within a single reference. Therefore, the Petitioners fail to

establish anticipation of any of the independent claims and the dependent claims based on this combination.

Each of the claims cited by the Petitioners for review in the petition requires the insertion of second information, such as contact information, into a document as the result of identification of first information within the document. Counter to the assertion by the Petitioners, LiveDoc fails to disclose how to achieve insertion into a document. The LiveDoc system operates in a manner external to both a text editor and the document within the text editor. Therefore, LiveDoc does not know the structure of the document and cannot perform an insertion into the document. Therefore, LiveDoc alone or in combination with either Drop Zones or Moore fails to teach all of the claim limitation of the claims.

Many of the independent claims require both a first application program and a second application program. The Petitioners look to Domini, Hachamovitch, and Luciw as anticipatory references of these claims. Domini discloses a spell checker module that operates inside of a word processing application. Thus, Domini discloses only a single application program and therefore, fails to disclose a second application and fails to disclose second information from a second application as required by these claims. Similarly, the petitioners suggest that the word completion program module of Hachamovitch discloses both a first application program and a second application program. However, Hachamovitch, like

Domini, describes a module that operates within a single application program, and therefore Hachamovitch does not teach each and every limitation of these independent claims and their corresponding dependent claims. Luciw teaches operation of a pen-based system that includes a note application that operates on a database to provide personal information management. As with both Domini and Hachamovitich, Luciw fails to teach both a first application program and a second application program.

The Petitioners also argue that Domini shows both a first and a second computer program. However, upon analysis of the Petitioners arguments, only one computer program is identified. Likewise, the Petitioners argue that Hachamovitch teaches both a first computer program and a second computer program. However, the Petitioners analysis only identifies a single computer program.

Because spell checker modules, such as those described in Domini, only search a dictionary for words based upon a guess of what the user intended to type, Domini fails to search for "related information" in a database, since there is no pre-existing relationship between the typed word and the guesses presented by the spell checker.

Because Hachamovitch automatically presents possible word completion suggestions selected from a list, Hachamovitch additionally fails to disclose "marking" "to alert the user that the first information can be utilized in a second

application program." Thus in Hachamovitch, the user is not first provided with an alert followed by a subsequent opportunity to use the first information in a second application program.

Some independent claims require "marking without user intervention the first information to alert the user that the first information can be utilized in a second application program." Luciw's pen-based system provides handwriting recognition, which converts a user's handwriting to text. The recognized handwriting alerts the user that the handwriting recognition has been successful. However, the conversion to text does not alert a user that the written information can be used in a second application program. As a result, Luciw additionally fails to disclose the "marking" claim limitation.

Certain claims require "analyzing in a computer process textual information in a document … to identify a portion of the document as first contact information, without user designation of a specific part of the textual information to be subject to the analyzing" In contrast, Luciw requires the user to identify a textual item by placing it in a specially designated location in the display or otherwise to select it in order to initiate a search for a name. Thus, analysis in Luciw is performed by the user, and does not occur without user intervention.

Luciw teaches both explicit assist and also implicit assist for assisting a user to perform a desired personal information management function. As shown in the

flow chart of Fig. 3, explicit assist and implicit assist each take an independent path. Petitioners suggest the elements from both paths can be combined to show anticipation of claims that require "only a single execute command" for performing the steps of "using a first computer program to analyze the document…", "using…a second computer program to search the database….", and "inserting….into the document." However, since the paths within the flow chart are completely independent, do not overlap, and describe separate examples, the Petitioners have additionally failed to show that Luciw teaches the "single execute command".

B. Because Petitioners rely on a combination of two articles, LiveDoc and Drop Zones, as a basis for arguing anticipation, Petitioners fail to establish anticipation of any of the independent claims and the dependent claims under Ground 1.

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987).

In many of its proffered rejections, Petitioner's improperly treat LiveDoc and Drop Zones as a single reference. For example, in Ground 1, the Petitioners argue that LiveDoc and Drop Zones. reproduced together in Petitioners' Exhibit 1005 and called by Petitioner "LiveDoc/Drop Zones", anticipate all of the

independent claims, and the dependent claims at issue in the present Petition [IPR2014-00207], under 35 U.S.C. §102(a).

However, LiveDoc and Drop Zones are not a single reference, as required for a rejection based on anticipation. These are two different articles included among dozens of different articles in the SIGCHI Bulletin. (Ex. 1005, p. 1) LiveDoc is an article published beginning at page 51 of the SIGCHI Bulletin, Volume 30, No. 2 entitled "From Documents to Objects: An Overview of LiveDoc." In contrast, Drop Zones is an article published beginning at page 59 of that SIGCHI Bulletin entitled "Drop Zones: An Extension to LiveDoc." In an Information Disclosure Statement pursuant to 37 C.F.R. 1.98, each would be separately listed.

Indeed, LiveDoc and Drop Zones are listed as separate articles in the '854 Patent itself as the next-to-last prior art reference and the prior art reference immediately preceding the next-to-last reference, in the second column of page 3. Because LiveDoc and Drop Zones are separate documents, their combination cannot be a basis for anticipation and therefore, Ground 1 fails to establish anticipation.

C. Because LiveDoc fails to disclose insertion of second information into a document that has been analyzed by its system outside of any application, Grounds 1,2,and 3 fail to establish anticipation or a *prima facie* case of obviousness

The independent claims (1, 7, 13, 36, 43, 50, 86, 93, 97, 98, 100, and 101) all require "inserting" second information into a document. Contrary to the assertion by the Petitioners, the LiveDoc publication simply does not meet this requirement of the claims, because LiveDoc fails to disclose how insertion into a document is or could be achieved. Thus, the Petitioners have not established a prima facie case of invalidity for independent claims 1, 7, 13, 36, 43, 50, 93, 98, and 101.

Unlike embodiments disclosed in the subject patent, as discussed above which utilize processes running inside the application program to achieve insertion, LiveDoc teaches processes running outside of the application, and therefore lacking intrinsic knowledge of the document or the application displaying it, so as to make implementation of insertion in that context beyond the state of the art.

The LiveDoc architecture is shown in Fig. 3 at page 56 of Exhibit 1005 where the LiveDoc manager communicates with an external application (i.e. a text editor) using API callbacks. *Id*. at 57 left col.

The LiveDoc application receives only the text from the text editor (application in Fig.3) and analyzes the text independently of the actual document in the text editor using a set of detectors under the control of an analyzer server: "A

first step to bridging the document gap described above is then to construct a means of passing text from a user's document into a parser for matching against a collection of recognizers." *Id*., p. 53. The LiveDoc application, including the detectors, analyzer server, and the LiveDoc manager does not know the formatting of the document or how the text editor in conjunction with the operating system presents the document on the display device nor how to change the contents of the original document, since LiveDoc only receives text. Therefore, LiveDoc does not operate on the document, but only on a copy of the text from the document. As stated on page 56 of Exhibit 1005:

> LiveDoc knows where these structures appear in the text passed to it- an e-mail address might appear in characters 150 through 162 of the window's contents – but it has no idea where in the window those characters physically appear, and, thus, where the highlights should appear: this is information held by the application, not by LiveDoc. Hence, LiveDoc must ask the application for the information about the structures it has found via a callback. Once this information is available, the highlights and their associated mouse-sensitive regions can be constructed.

LiveDoc receives the window position of the identified structures in the document from the callback and can construct highlights that can be overlaid on the displayed text, and thus, LiveDoc does not operate on the actual document and

does not teach insertion into a document that is open in an application.[1] "The LiveDoc Manager updates the display to present the highlight information *over* the discovered structures when the function key is pressed." *Id,* at p. 56 (emphasis added). The term "over" indicates that the highlighting is an overlay and is not inserted into the document. To similar effect is this passage in LiveDoc: "The use of highlighting is one of these: adding the notion of a **sometimes-visible layer to the front of the display** is a considerable change to the graphical interface…" (Emphasis added). *Id*., p. 57.

In use, the functionality of LiveDoc becomes available only when a user pushes and holds the function key to reveal the overlaid highlighting, a process indicating that the user has left the first application and entered LiveDoc mode. *Id.,* p. 56. Thus, in the LiveDoc system, the user sees structures (e.g. contact

[1] LiveDoc teaches that the text editor uses a number of callback handlers to respond to calls from LiveDoc Manager. "The most important of these handlers inform the LiveDoc Manager of changes to the content of the document window, perhaps by the user adding or deleting content, or by the scrolling or resizing of the window." *Id.* at Page 56 left col. Thus, LiveDoc teaches that the callbacks are used to determine position information regarding the window that displays the text of the document on a display device and thus, operates external to the text editor (application program).

information) detected by the detectors (highlighted "within the context of the document"), even though the LiveDoc system operates outside of the document. Accordingly, while the LiveDoc reference discloses displaying features "within the context of the document", such features are superimposed over a display of the document and not inserted into the document.

The Petitioners suggest that insertion into the document is taught by the following passage at page 58 of LiveDoc:

> Imagine a detector that finds the formula of an organic molecule in a document, and an action that presents a three-dimensional rendering of that molecule within the context of the document itself, rather than in a separate application." (emphasis added).

The cited passage does not disclose insertion into the document; rather it states merely that the rendering is presented "within the context of the document" like the highlighting of the structures in LiveDoc mode. The three-dimensional representation may simply be a graphic that is rendered "on the fly" and displayed within a window controlled by the LiveDoc application without accessing the actual document, and without inserting the three-dimensional rendering into the document, or having any interaction at all with the text editor. The three-dimensional rendering, is just that, a "rendering" based on the formula of the molecule. "[W]ithin the context of the document" only means that the user must not explicitly start up another application to see the rendered representation.

Additionally, the cited passage from LiveDoc is merely one of conjecture on the part of its authors beginning with the term "Imagine". The passage appears in the section of LiveDoc entitled "Futures: Extensibility and Semantics" indicating future work to be done and does not even contemplate that this feature had been implemented. In fact, the immediately preceding sentence discloses what actually was implemented: "Our initial implementation of LiveDoc as LiveSimpleText assumed that actions would be handled by external applications, such as a Web browser presenting the page pointed to by a URL. However, other styles of interaction exist: "The description of the actual implementation does not refer to the ability to insert information into the document or any operation on the document itself. Rather, the implementation merely states, "actions would be handled by external applications."

Since LiveDoc operates outside the application displaying the document, it does not even know how to insert anything into the document[2]. Nor does LiveDoc know where in the document the structures appear, it only learns from a call to the application, as discussed above, where on the screen the structures are located.

[2] See also LiveDoc, 58: "…as noted earlier, the actions associated with a structure are static, rather than dynamically reflecting the nature of the user's interaction context." In other words, the actions do not reflect e.g. the document in which the structures were found.

Recall also that LiveDoc only analyzes, i.e. gets, the part of the document which is visible:

> "The most important of these handlers inform the LiveDoc Manager of changes to the content of the document window, perhaps by the user's adding or deleting content, or by the scrolling or resizing of the window. The receipt of these calls by the LiveDoc Manager signals the Analyzer Server to analyze the text provided by the calling application; this will typically be the text currently visible in the applications' front-most window." LiveDoc, 56.

But this is not enough to insert anything into the document, to insert something into a document, one needs to know, for example, which programmatic calls to make to the specific application displaying the document (and these will be different from application to application[3], and where in the document, (not the screen). LiveDoc has no such knowledge, and thus cannot insert anything into a document as required by the claim.

Drop Zones does nothing to overcome the defects of LiveDoc, because it similarly fails to disclose how to achieve insertion into a document. Thus, the combination of LiveDoc with Drop Zones fails to teach each and every limitation of the independent claims and therefore, fails to render the independent claims obvious.

[3] LiveDoc makes no mention of any special common API for the Applications to accomplish such functionality.

Moore, a chemical database program, similarly does nothing to overcome the defects of LiveDoc, because it fails to disclose how to achieve insertion into a document that has been analyzed by its system outside of any application. Moore merely teaches a database program that includes an association between chemical formulas and graphical representations of the formulas. As a result, the combination of Moore with LiveDoc and with Drop Zones fails to teach each and every limitation of independent claims and therefore, does not make the independent claims obvious. The Petitioners have also failed to provide a rationale why someone skilled in the art at the time would combine Moore with LiveDoc or with Drop Zones.

For at least these reasons, none of Grounds 1, 2, or 3 establish anticipation for claim 1-18, 36-56, 93-95, 98, and 101 or establish obviousness of claims 1-18, 36-56, 93-95, 98 and 101.

D. Because the Domini spell checker is a module operating inside of the first application, Domini fails to disclose a second application and fails to disclose second information from a second application, and therefore Ground 4 fails to establish anticipation.

The '854 Patent discloses an embodiment including the retrieval of information (*e.g.*, a name or address) from one application program (e.g. Microsoft Outlook) while a user works simultaneously in another application program (e.g., Microsoft Word). See Exhibit 1001 col. 2 lines 14-23. The independent claims (1, 7, 13, 36, 43, 50) all require "a first application program" and "a second

application program." As would be understood by one of ordinary skill in the art, an "application program" as required by these independent claims is an independently executable program, as opposed to a utility, such as a spell checker, that adds functionality to an application program, and is not independently executable. Domini uses Application Programing Interfaces (APIs) to link the spell checker program module to a word processing application program. Domini states that the common speller application program interface (CSAPI) "makes it easy for programmers to write applications" that can utilize the grammar checker program module." Col. 8 lines 57-62. The spell checker program is activated from within the word processor col. 16 lines 12-16 Thus, the spell checker program module of Domini is not an "application program" as that term is understood in the context of the claims herein.

The '854 Patent describes the interaction between two application programs wherein a user may operate within a first application program without the need for exiting and initiating a second application program to find information managed by the second application program that is related to information entered into a document within the first application program. Examples of application programs as specified within the specification of the '854 Patent encompass only separately executable computer programs: a word processing program, such as Microsoft Word, a spread sheet program, such as Microsoft Excel, and a contact database

program, such as Microsoft Outlook. Ex. 1001, col. 1, lines 39-42 and col. 9, line 64, to col. 10, line 10.

The Domini reference, in contrast, discloses the retrieval of information (e.g., suggested spelling corrections) while a user works within a single program—i.e. a word processor. See, e.g., Ex. 1007 Abstract, col. 3 lines 1-12, col. 5 lines 1-8, col. 8 lines 51-57 , col. 9 lines 6-12 etc. See specifically Col. 5 lines 1-8 reproduced below:

> The preferred embodiment of the present invention is represented by "WORD", version 8.0, which is a word processing application program produced by Microsoft Corporation of Redmond, Wash. Briefly described, the preferred program allows users to create and edit electronic documents by entering characters, symbols, graphical objects, and commands. The preferred program includes a spell checker program module and a grammar checker program module. (emphasis added).

The just quoted passage makes clear that the spell checker utility of Domini adds functionality to an application program such as Microsoft Word, and is not independently executable. Because Domini fails to teach a second application program, Domini similarly fails to disclose or suggest using both a "first application program" and a "second application program" as required in independent claims 1, 7,13,36,43, and 50. As a result, Ground 4 fails to establish anticipation of the independent claims listed above.

Additionally, since Domini fails to disclose "a second application", Domini also fails to disclose "second information" "associated with the first information from a second application program" as required by independent claim 1, 7, 13, 36, 43, and 50. Thus, Domini fails to teach the "associated information" limitation of the independent claims in addition to failing to teach a second application program.

Also, as a practical matter, the dictionary that is searched in Domini is not a "second application program" because a dictionary is merely a listing of properly-spelled words, and therefore not a "program" in any sense. See, e.g., the definition of "dictionary" from the Heritage College dictionary 3rd edition 1997 in Patentee's Exhibit 2001.

For at least these reasons, Ground 4 fails to establish anticipation by Domini of the above listed independent claims or for dependent claims 2-6, 8-12, 14-18, 37-42, 45-49, 51-52, and 54-56 .

E. Because spell checker modules, such as those described in Domini, only search a dictionary for words based upon a guess of what the user intended to type, Domini fails to search for "related information" in a database and therefore, for this additional reason, Ground 4 fails to establish anticipation by Domini of claims 93, 98, and 101.

Spell checker modules, such as those of Domini, also fail to "search" for "related information" in a "database" based upon text identified in a document as required at least by independent claims 93, 98, and 101. As a user enters a series of characters into a document within a word processor, or starts a spell checking

for part of or the whole document, the spell checking program module searches a dictionary and if the series of characters are not found within the dictionary, the series of characters is marked. See Ex. 1007 col. 11 lines 49-51. The spell checker then identifies possible words based on a best guess of what the user intended to type.

For example, a user may type a set of characters such as "reid" and the spell checker will mark the characters because the specified set of characters in the specified order do not appear within the spell checker's dictionary. The spell checker then makes a guess at the intent of the user, by displaying possible choices, such as "red, read, reader etc." See Ex. 1007 Col. 12 lines 3-7. The search in Domini is not a search for information that is "related to" the entered information. These guesses are not "related information" as there is no pre-existing "relationship" between the typed characters and the retrieved words within the dictionary. The spell checker will allow a user to select from the entries, and upon the selection of an entry, a relationship is established. *Id.* at col. 13 lines 1-6. This relationship occurs too late to meet the limitation of using the "text in the document" to "search" the "database" to located "related information," since the spell checker's search is based upon a guess and not based upon a relationship between the typed information and the related information.

In fact, there may be no relationship at all between the words that the spell checker finds and the intent of the user. As in the example presented, the user may have intended to type "reid" as a name or may have intended to type the word "ride", which was not presented to the user. Thus, in this case, no relationship exists between the typed characters and the items that are identified in the search of the dictionary. See Ex. 1007 col. 11 lines 36-39.

Spell checkers never search for related information, wherein there is an already established association between the entered text (first information) and the related information (second information). As a result, for at least these reasons, the Ground 4 fails to establish anticipation for independent claims 93, 98, and 101 based on 35 U.S.C. 102(e).

F. Because Domini fails to disclose "a second computer program" as required by independent claims 93, 98 and 101 as evidenced by the Petitioners' analysis of Domini, Ground 4, for this additional reason, fails to establish anticipation.

Independent claims 93, 98 and 101 each include limitations that require "a first computer program", "a second computer program," and a "database". Thus, three distinct elements are required to meet the limitations of these claims. Although the Petitioners have made inconsistent statements as to what corresponds to the "second computer program" in the Domini reference, under any possible interpretation, the Petitioners fail to show that Domini teaches a "second computer program" distinct from a "first computer program" and a database. Petitioners

assert that the "spell check program" of Domini is the first computer program. See petition at 35 referencing [93b](1) wherein the "spell check program" is equated to the first computer program. The Petitioners point the reader to their previous analysis of claim [1c] and [36b] wherein the first application program is clearly the "spell check program."

Petitioners suggest in their claim chart at pages 34-35 of the Petition that the "second computer program" is either (1) the spell check program itself or (2) the spell check dictionary. The Petitioners assert that the "second computer program" can be found with respect to parts 1d and 3c previously discussed in the petition. Yet part 1d on pages 30 and 31 fails to discuss a second computer program at all, while the immediately preceding (and unmentioned) part, part 1c, does reference a "second application program" wherein the spell checker program is the "second computer program". (See [1c] at page 30 "spell checker program 37B"). (Presumably, the Petitioners deem "application program" and "computer program" to be synonymous.) Even though part 1c identifies the spell check program 37B as the "second computer program", part [3c] states that "the spell check program" has "one or more dictionaries" that correspond to the "second application program". Thus, Petitioners have inconsistently specified what corresponds to the second computer program limitation, identifying separately the "spell check program" itself and the "one or more dictionaries".

Because the "first computer program" and the "second computer program" are distinct entities, the spell checker of Domini cannot correspond to both. Thus, this interpretation of Domini fails to meet the claim requirements of having a "first" and a "second" computer program. In the alternative, if the Petitioners mean that the spell checker corresponds to "first computer program" and the "one or more dictionaries" corresponds to "second computer program", this position also fails, because a computer dictionary is not the equivalent of a "computer program." A dictionary is merely a list of terms—that is, it is data—and lacks any associated computer code.

Dictionary definitions may be used to determine the ordinary and customary meaning of words. *Ferguson Beauregard/Logic Controls v. Mega Systems*, 350 F.3d 1327, 1338 (Fed. Cir. 2003) (Dictionary definitions were used to determine the ordinary and customary meaning of the words "normal" and "predetermine" to those skilled in the art. Referring to the "American Heritage College dictionary 3rd edition 1997, the term "dictionary" as used in the field of computer science is "A list of words stored in machine-readable form for reference, as by spelling-checking software." See Exhibit 2001. Thus, according to its ordinary meaning, a "dictionary" is simply a list of words and cannot be the "second computer program." The Petitioners have failed to identify in Domini distinct items

corresponding to both the "first computer program" and the "second computer program" as required by the claims.

More specifically, Claim 93 requires "using a first computer program to analyze the document, without direction from the operator, to identify text in the document" and further requires "using a second computer program" and the identified text "to search the database and to locate related information." At best, the Petitioners show that Domini teaches a spell check program (i.e. first program) that uses a dictionary, which Petitioners equate to a database, but the Petitioners still fail to show the third required element, of a second computer program. Since the Petitioners have failed to show that Domini discloses the three distinct entities of a "first computer program", a "second computer program" and a separate "database", Ground 4 fails to establish anticipation based on 35 U.S.C. 102(e). Independent claims 98 and 101 require the same limitations as claim 93 and for the same reasons as provided above, the Petitioners have failed to establish anticipation for these claims as well.

G. Because Hachamovitch describes a program module operating within a first application, Hachamovitch fails to disclose "a second application program", and therefore, Ground 5 fails to establish anticipation and Ground 6 fails to establish a *prima facie* case of obviousness.

As previously articulated with respect to the Domini reference, the independent claims 1, 7, 13, 36, 43, and 50 in the '854 Patent require "a first

application program" and a separate "second application program" where the two application programs are separately executable.

Hachamovitch describes a word completion utility for use with a word processor and further describes the utility as a program module. A program module is distinct from an application program, since a program module cannot be used independent of a separate application program. According to Hachamovitch, the word completion utility is "an interface defined within each application program through which the word completion utility may communicate with each application program." See Ex. 1008 col. 8 lines 6-9.

Hachamovitch states that the word completion utility can be deployed within an individual application program (See *Id.* at col. 7 lines 62-64) or the utility can be an application-independent utility (See *Id.* At col. 7 lines 65-67).

When referencing the word completion utility as an application-independent utility, Hachamovitch does not contemplate the word completion utility as a separately executable application program, but rather a utility that may be accessed by a plurality of application programs. "To deploy the word completion system as an application-independent utility, an interface is defined within each application program through which the word completion utility may communicate with each application program." *Id.,* col. 8, lines 6-9.

Thus, an application-independent utility is simply a program module that can be accessed by a plurality of application programs and the utility does not operate independent of an application program. Hachamovitch, like Domini, lacks a second application program.

For at least this reason, Ground 5 fails to establish anticipation of independent claims 1, 7, 13, 36, 43 and 50 and corresponding dependent claims 2-6, 8-12, 14-18, 37-42, 44-49, and 51-56 and Ground 6 fails to establish a *prima facie* case for obviousness of dependent claims 3-5, 9-11, 15-17, 38-41, 45-48, and 53.

H. Because the Hachamovitch reference fails to disclose "a second computer program" as required by independent claims 86, 93, 97, 98, 100, and 101 as evidenced by the Petitioners' analysis of Hachamovitch, Ground 5 fails to establish anticipation and Ground 6 fails to make a *prima facie* case of obviousness for claims 88, 90, and 91.

Independent claims 86, 93, 97, 98, 100 and 101 each require "a second computer program" and the Petitioners have failed to show that Hachamovitch teaches a second computer program. Petitioners suggest in their claim chart for claim 86 on page 42 and for claim 93 on page 43 of the *Petition* that the second computer program [see 86c] is a "suggestion list database." As would be understood by one of ordinary skill in the art, a database is merely the storage of related information in a memory and a database is not a computer program. Data

stored in memory cannot be equated with a computer program that includes computer executable instructions.

Further, the Patent Owner notes that claims 86 and 93 require the three elements of: "a first computer program", "a second computer program", and "a database." Thus, as used in these claims, a database is distinct from either a first computer program or a second computer program. Petitioners' interpretation of Hachamovitch as evidenced by their claim chart suggests that the database represents both the "second computer program" and the "database". Even if one were to consider the database of Hachamovitch to be a second computer program, Petitioners would still have failed to show a separate database. Thus, at most, Petitioners can show that Hachamovitch teaches two of the three required elements of the independent claims: a first computer program and a database, or a first computer program and a second computer program. However, the Petitioners have failed to show that Hachamovitch teaches all three elements of "a first computer program," "a second computer program," and "a database." Thus, the Petitioners have not established that Hachamovitch teaches each and every limitation of independent claims 86 and 93 and Ground 5 fails to establish anticipation based on 35 U.S.C. 102(e) in view of Hachamovitch. Independent claims 97, 98, 100, and 101 also require a first computer program, a second computer program and a database and therefore, Ground 5 fails to establish anticipation based on 35 U.S.C.

102(e) for these independent claims in view of Hachamovitch as well. Similarly, Ground 6 fails to establish a *prima facie* case of obviousness based on 35 U.S.C. 103(a) for claims 88, 90, and 91, since Ground 5 does not establish that Hachamovitch teaches each and every limitation of independent claim 86.

I. Because Hachamovitch automatically presents possible completion suggestions selected from a list, Hachamovitch fails to disclose "marking" "to alert the user that the first information can be utilized in a second application program", and thus, Ground 5 fails to establish anticipation and Ground 6 fails to make a *prima face* case of obviousness.

Independent claims 1, 7, and 13 each require the limitation of "marking without user intervention the first information *to alert the user that the first information can be utilized in a second application program."* (emphasis added). The phrases "to alert the user" and "can be utilized" indicate that the marking is first presented to the user and subsequently, the user can utilize the marked first information in a second application program. Because of the chosen language for this "marking" limitation, this limitation requires a sequence of actions, where the alerting of the user by marking occurs before the user can utilize the first information in a second application program.

In contrast, Hachamovitch does not teach marking of first information prior to the user utilizing the first information in a second application program. When a user types letters i.e. first information into a word processor, the auto-complete utility of Hachamovitch automatically accesses the database and identifies the most

likely completion suggestions based on the "first information". The Petitioners suggest that the "marking" of the first information is indicated by Figs. 2A-2C and col. 10 lines 31-37 of Hachamovitch where a completion suggestion is displayed along with the partial data entry. See Petition at page 38. By the time that marking is performed by the Hachamovitch invention, no further use is made of the "first information", since the suggested completion has already been retrieved. Thus, the marking in Hachamovitch does not provide an alert to the user that the "first information" "can be used" in a second application program. The contemplated marking in Hachamovitch as suggested by the Petitioners alerts the user that the first information "has already been used" to search the suggestion list.

As a result, Ground 5 fails to establish anticipation of claims 1-18and Ground 6 fails to establish a *prima facie* case of obviousness for dependent claims 3-5, 9-11, and 15-17 for at least the same reasons.

J. Because Luciw discloses only a single application for personal information management, Luciw fails to disclose "a second application", and therefore fails to disclose "marking without user intervention the first information to alert the user that the first information can be utilized in a second application program", for that reason, Ground 7 fails to establish anticipation and Ground 8 fails to make a *prima facie* case of obviousness.

Luciw fails to disclose "a second application program" as required by independent claims 1, 7, 13, 36, 43 and 50. Luciw functionally describes a personal information manager i.e. a notepad program, which allows one to store

and retrieve contact information from a computer database. Luciw provides assistance to a user based on a user's entry within the notepad application by performing a database search and retrieving related information and providing the ability to perform an operation on the information (e.g. make a call etc.). Luciw identifies the notepad application as "an application program running under the operating system…" Exhibit 1009 col. 6 lines 49-51. Although Luciw mentions that the computer system may include other "applications," Luciw does not describe operation of these "other" applications and does not disclose any interaction between the notepad application program and any "other" program. *Id.* col. 6 lines 55-59. Luciw only references the term "database" generically and does not refer to the database as an "application program". Luciw identifies the "computer database" as a "frame database system" where a "frame" describes the formatting of database entries. See Fig. 5 where a frame of type <person> would include attributes such as name, birthday, telephone number etc. See also *Id.* col. 10 lines 51-60. Thus, Luciw merely discloses the interaction between a single "application program" (notepad) and an associated computer database and fails to teach a "second application program" as required by independent claims 1, 7,13, 36, 43 and 50. Luciw therefore does not teach "marking without user intervention the first information to alert the user that the first information can be utilized in a second application program" and does not teach "responding to a user selection by

inserting a second information into the document the second information associated with the first information from a second application program" as required by independent claim 1 since, only a single application program (the operation of notepad) is discussed in Luciw.

The Petitioners allege that the second application program is the database; however, the Luciw disclosure does not refer to the database as a database program or as an application program, but merely references that the database exists and has a format. Luciw clearly contemplates "application programs" by referencing the notepad application as such. Thus, one of ordinary skill in the art would understand the "computer database" of Luciw merely to be a collection of related data entries within a memory.

The '854 Patent and the associated claims are not directed to operation of a single application program and an associated database as disclosed in Luciw, but rather to obtaining associated information from a second application program based upon first information that is entered into a document of a first application program. One of ordinary skill in the art would understand that an "application program" is a self-contained and executable computer program for performing a specific function. Because Luciw fails to teach a second application program, Ground 7 fails to establish anticipation based on 35 U.S.C. 102(e) for at least independent claims 1, 7,13, 36, 43 and 50. Similarly, since each and every

limitation of the independent claims has not been shown within Luciw, Ground 7 fails to establish anticipation of dependent claims 2-6, 8-12, 14-18, 37-42, 44-49, and 51-56. Additionally, for at least this reason, Ground 8 fails to establish a *prima facie* case of obviousness for dependent claims 3-5, 9-11, 15-17, 38-41, 45-48, and 53.

K. Because Luciw requires the user to identify a textual item by placing it in a specially designated location in the display or otherwise to select it in order to initiate a search for a name, Luciw fails to disclose "analyzing …", "identifying," or "marking" "*without user designation*" of a specific part of the textual information, and therefore for this additional reason Ground 7 fails to establish anticipation and Ground 8 fails to make a *prima facie* case for obviousness.

Independent claims 1, 7, and 13 require "marking *without user intervention* the first information to alert the user that the first information can be utilized in a second application program." Independent claims 36, 43,50 require "*identifying without user intervention or designation* the first information." Independent claims 93, 97, 98, and 101 include the limitation of "using a first computer program to analyze the document, *without direction from the operator*, to identify text in the document that can be used to search for related Information" and independent claim 86 and 100 require a similar "analysis limitation" of "using a first computer program to *analyze the document, without direction from the operator*, to identify the name."

The Petitioners argue that element [1c] as provided in the claim chart at pages 47 and 48 of the Petition meets the "marking" "without user intervention" limitation and also meets the "identifying", and "analyzing" limitations in independent claims 36 and 86. See claim chart element 36[b] at page 51 and 86[b] at page 52 of the Petition.

As discussed in further detail in section IV(E) above, the Luciw patent describes "implicit" assistance, wherein a user has used a smart field to enter a word used for look up in the database or has otherwise similarly triggered a database lookup, and "explicit" assistance, wherein the user explicitly invokes assistance from the device as by using pen 38 of Fig. 2. See Exhibit 1009, col. 8, lines 16-62.

In order to use a smart field, the user must select a name or phone field depending on whether the textual item that the user wants to be searched is a name or a phone number:

> Once the particular window 170 is presented to the user, the name ISAAC can be handwritten into the particular smart field 175. The assistance process recognizes the handwritten name "Isaac," and either continues operation as suggested at step 106 in FIG. 3 directly, or concurrently displays the recognized name in formal font form, as suggested in FIG. 4c, in the same position of the smart field, where formerly the handwritten name "Isaac" had been established. As will readily be recognized, window 170 in FIG. 4b may contain several

> smart fields, in this case for example definable for either the "name" field 175 or a "phone" field shown at step 177.
>
> *Id*., col. 10, lines 28-39.

This passage makes clear that whereas the recognition achieved by the computing device of Luciw is of handwriting—namely translating a handwritten name into displayed text—nevertheless, in order to retrieve information from the database, the user is expected to enter a name into the name field 175 or a phone number into the phone field 177. Note that in Figs. 6a, 6b, and 6c the name field and the phone field are given the same item numbers, 175 and 177 respectively, as in Figs. 4b and 4c discussed above. Thus in using a smart field, a user is expected to identify the first information and as well as to tell the computing device what type of information the user is entering—user designation is required, in contravention of the claim requirement that "marking," "identifying" or "analyzing" occurs without user intervention.

If the handwriting or text has not been not placed in a smart field, Luciw states that a user may enter an "indication or word" on screen to trigger implicit assist, but fails to disclose any mechanism for how this might be achieved. See *Id.* at Col. 8 lines 29-41. In the absence of an implicit assist, Luciw requires the user to specify an explicit assist and then select the item to identify the item as one with respect to which action is to be taken. The user must still tell the device what to do. See *Id.* at Col. 8, lines 34-40. Moreover, it should be noted that in Fig. 3, in the

context of an "explicit assist" in step 110, none of the steps that follow step 110 include any second information associated with first information or related contact information at all. This means that in the context of an explicit assist there cannot follow the claimed "inserting a second information into the document, the second information associated with the first information", or "inserting the contact related information into the document". Thus even if Luciw were to disclose analysis as required by the claim in connection with the explicit assist (which it does not), the first contact information found by this analysis would never be used for inserting the second information into the document as required by each of the independent claims.

Thus, Ground 7 and Ground 8 fail to establish that Luciw teaches the limitations of "marking," "identifying," and "analyzing" the first information(e.g. "text in a document" such as a "name") "without user intervention" as required by independent claims 1, 7, and 13, 36, 43, 50, 86, 93, 97, 98, 100 and 101 and therefore, Ground 7 fails to establish anticipation under 35 U.S.C. 102(e) for these independent claims and corresponding dependent claim 2-6, 8-12, and 14-18, 37-42, 43-49, 51-56, 87-88, 92, and 94 and Ground 8 fails to establish a *prima facie* case of obviousness under 35 U.S.C. 103(a) for at least the same reasons for claims 3-5, 9-11, 15-17, 38-41, 45-48, 53, 87, 89, 91 and 95.

L. Because Luciw fails to disclose requiring "only a single execute command" for performing the steps of "using a first computer program to analyze the document…", "using…a second computer program to search the database….", and "inserting….into the document" , Ground 7 fails to establish anticipation of independent claims 86, 97, and 100.

Independent claims 86, 97, and 100 each include the limitation of requiring "only a single execute command" for performing steps (1)-(3). Steps (1)-(3) as specified in claim 86 require (1) "using a first computer program to analyze the document, without direction from the operator, to identify the name"; (2) "using the identified name and a second computer program to search the database and to locate contact related information associated with the name"; and (3) inserting the contact related information into the document."

As shown in Fig. 1 of the '854 patent, "a single execute" command is used to begin the process.

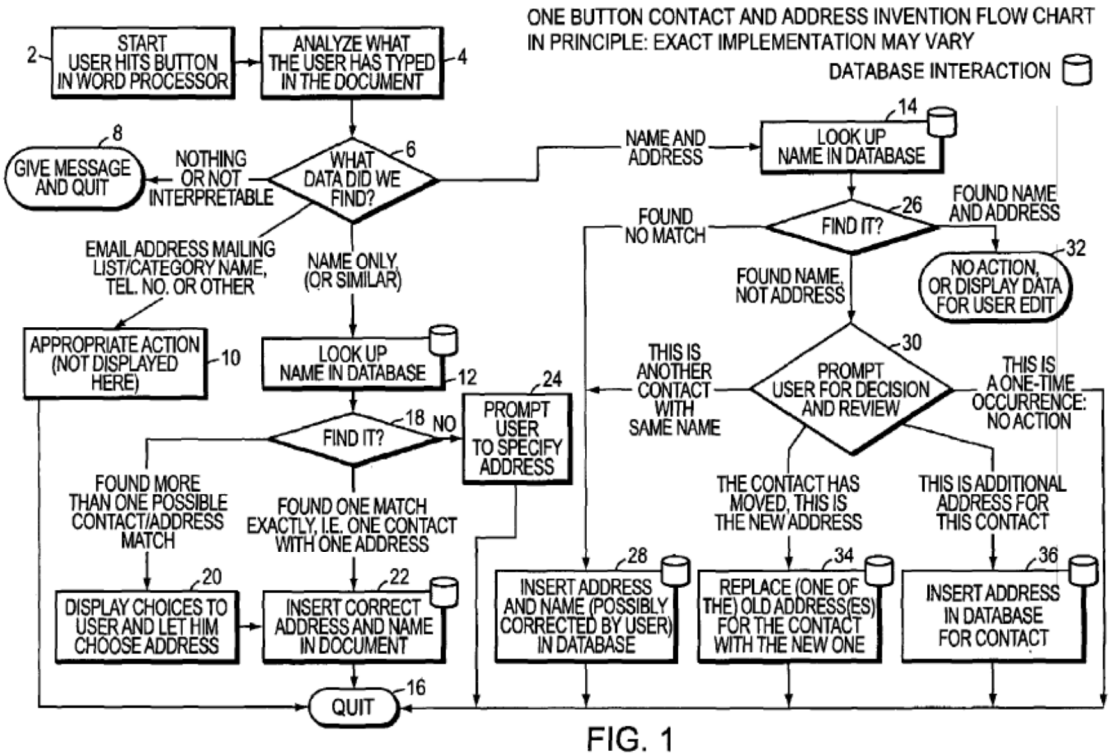


FIG. 1

In block 2, the user hits a button (i.e. the "one button"). Analysis of the document occurs in block 4 without the user having to designate the text within the document to be analyzed. The methodology determines the type of information resulting from the analysis in block 6, and as a result of the identification of a "name", a second computer program searches a database in block 12 to locate "contact related information" associated with the name. If the contact related information (e.g. an address) related to the name is found in block 18, the methodology causes the contact related information to be inserted into the document in block 22. Thus, by pressing the one button (i.e. a single execute command) steps 1-3 are performed.

The Petitioners suggests that Luciw meets this claim limitation through operation of an "explicit assist" command. " Petitioners cite to the combination of col. 9 line 16 – col. 10 line 5 with Figs. 7a-7c, col. 12 lines 7-40, Figs. 6a-c, col. 11line 60 to col. 12 line 6, and col. 12 lines 41-54 to show an "explicit assist" that requires "only a single execute command." See Petition at 53.

The characterization of Luciw by the Petitioners is inconsistent with the flow chart of Fig. 3. The Patent Owner notes that the flow chart of Fig. 3 of Luciw has two separate legs, one for "implicit assist" and one for "explicit assist" as shown below.

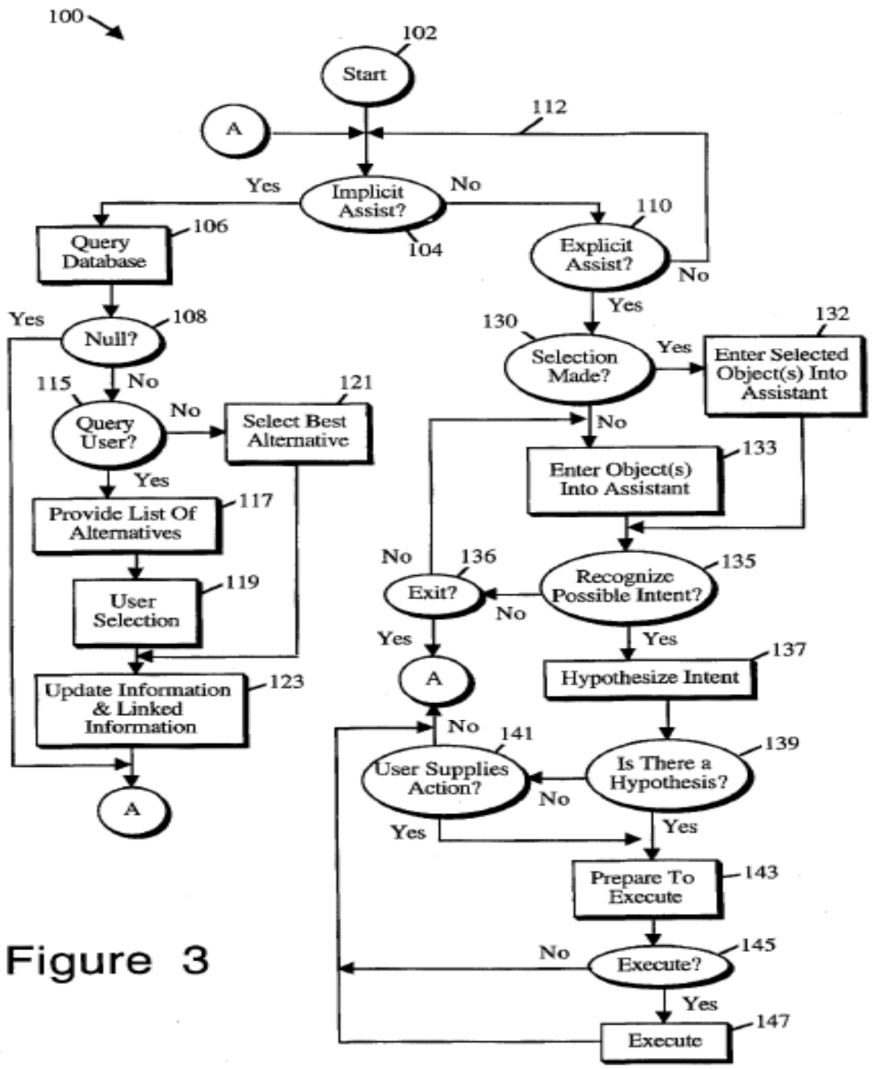


Figure 3

In their analysis, the Petitioners have cited to both the "explicit assist" leg of Fig. 3 (steps 110-147) and also to the "implicit assist" leg of Fig. 3 (steps 106-123) to show a "single execute command" in Luciw that performs steps 1-3 of the claim. The steps performed in Luciw within the "explicit assist" leg are not also performed in "implicit assist" leg and each leg is executed independently without any crossover. As a result, the combination of steps suggested by the Petitioners beginning with an explicit assist command from the explicit assist leg of Fig. 3 and

jumping to the implicit assist leg for showing steps 1-3 of the claim never actually occurs in Luciw. Thus, an "explicit assist" command cannot be a "single execute command" for performing the method steps of an "implicit assist," since the legs of the flow chart of Fig. 3 are independent. Patent Owner notes that Col. 9 line 16 – col. 10 line 5 is directed to "explicit assist" and has been used to show a "single execute command", whereas Figs. 7a-7c and col. 12 lines 7-40 are directed to "implicit assist." including step 121, whereas Figs. 6a-c, col. 11line 60 to col. 12 line 6 are directed to "implicit assist" including step 117 , whereas col. 12 lines 41-54 is directed to "implicit assist" including step 123.

Thus, Ground 7 fails to establish anticipation of independent claims 86, 97, and 100. Similarly, dependent claims 87-88, 90, 92 are also not anticipated under Ground 7.

VI. CONCLUSION

For the foregoing reasons, Petitioners have failed to establish a reasonable likelihood of prevailing as to any claim of the '854 Patent, and *inter partes* review of claims 1-18, 36-56, 86-95, 97, 98, 100 and 101 of U.S. Patent No. 7,496,854 should be denied.

Dated: March 12, 2014

Respectfully submitted,

/Robert M. Asher, #30,445/

Robert M. Asher
Registration No. 30,445
Bruce D. Sunstein
Registration No. 27,234
John J. Stickevers
Registration No. 39,387
Sunstein Kann Murphy & Timbers LLP
125 Summer Street
Boston, MA 02110
Tel: (617) 443-9292
Fax: (617) 443-0004

CERTIFICATE OF SERVICE

It is certified that on March 12, 2014, copies of the Preliminary Response of the Patent Owner under 35 U.S.C. § 313 and 37 C.F.R. § 42.107 have been served on Petitioners as provided in 37 C.F.R. § 42.6(e) via electronic mail transmission addressed to the persons at the following addresses:

LEAD COUNSEL FOR PETITIONER APPLE DAVID L. FEHRMAN dfehrman@mofo.com Registration No. 28,600 MORRISON & FOERSTER LLP 707 Wilshire Blvd., Suite 6000 Los Angeles, CA 90017-3543 Tel: (213) 892-5200 Fax: (213) 892-5454	BACK-UP COUNSEL FOR PETITIONER APPLE MEHRAN ARJOMAND marjomand@mofo.com Registration No. 48,231 MORRISON & FOERSTER LLP 707 Wilshire Blvd., Suite 6000 Los Angeles, CA 90017-3543 Tel: (213) 892-5200 Fax: (213) 892-5454
LEAD COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC. MATTHEW A. SMITH smith@turnerboyd.com Registration No. 49,003 TURNER BOYD LLP 2570 W. El Camino Real, Suite 380 Mountain View, CA 94040 Tel: (650) 265-6109 Fax: (650) 521-5931	BACK-UP COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC. ZHUANJIA GU gu@turnerboyd.com Registration No. 51,758 TURNER BOYD LLP 2570 W. El Camino Real, Suite 380 Mountain View, CA 94040 Tel: (650) 265-6109 Fax: (650) 521-5931 and kent@turnerboyd.com docketing@turnerboyd.com

Date: March 12, 2014

/Robert M. Asher, #30,445/
Robert M. Asher
Registration No. 30,445
Sunstein Kann Murphy & Timbers LLP
125 Summer Street
Boston, MA 02110
Tel: (617) 443-9292
Fax: (617) 443-0004

Exhibit 6R

Trials@uspto.gov
Tel: 571-272-7822

Paper 9
Entered: June 11, 2014

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC., GOOGLE INC., and MOTOROLA MOBILITY LLC
Petitioner

v.

ARRENDI S.A.R.L.
Patent Owner

Case IPR2014-00207
Patent 7,496,854 B2

Before SALLY C. MEDLEY, HOWARD B. BLANKENSHIP, and TREVOR M. JEFFERSON, *Administrative Patent Judges.*

JEFFERSON, *Administrative Patent Judge*.

DECISION
Institution of *Inter Partes* Review
37 C.F.R. § 42.108

I. INTRODUCTION

A. Background

Apple Inc., Google Inc., and Motorola Mobility LLC (collectively "Petitioner") filed a Petition (Paper 3, "Pet.") to institute an *inter partes* review of claims 1-18, 36-56, 86-95, 97, 98, 100, and 101 of U.S. Patent No. 7,496,854 B2 (Ex. 1001, "the '854 patent"). Pet 1; *see* 35 U.S.C. § 311. Arendi S.A.R.L. ("Patent Owner") filed a Preliminary Response (Paper 7, "Prelim. Resp.").

The standard for instituting an *inter partes* review is set forth in 35 U.S.C. § 314(a), which provides as follows:

> THRESHOLD.—The Director may not authorize an inter partes review to be instituted unless the Director determines that the information presented in the petition filed under section 311 and any response filed under section 313 shows that there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.

Upon consideration of the petition, we conclude that there is a reasonable likelihood that Petitioner would prevail with respect to claims 1-12 and 36-49 of the '854 patent. Accordingly, we authorize an *inter partes* review to be instituted as to claims 1-12 and 36-49, of the '854 patent.

B. Related Matters

Patent Owner sued Petitioner for infringement of the '854 patent in *Arendi S.A.R.L. v. Apple Inc*., No. 1:12-cv-01596-LPS (D. Del.); *Arendi S.A.R.L. v. Motorola Mobility LLC*, Case No. 1:12-cv-01601-LPS (D. Del.). Pet. 1.

Petitioner filed a petition for *inter partes* review of claims 19-35, 57-85, 96, and 99 of the '854 patent in IPR2014-00206. *Id*. at 3-4.

C. References Relied Upon

Petitioner relies upon the following prior art references:

Reference(s)	Exhibit
James R. Miller & Thomas Bonura, *From Documents to Objects: An Overview of LiveDoc*, SIGCHI BULLETIN, Vol. 30, No. 2, April 1998, pp. 53-58 ("LiveDoc") and James R. Miller & Thomas Bonura, *Drop Zones: An Extension to LiveDoc*, SIGCHI BULLETIN, Vol. 30, No. 2, April 1998, pp. 59-63 ("Drop Zones") (collectively, "LiveDoc/DropZones")	Ex. 1005
U.S. Patent No. 5,577,239 ("Moore")	Ex. 1006
U.S. Patent No. 6,085,206 ("Domini")	Ex. 1007
U.S. Patent No. 6,377,965 ("Hachamovitch")	Ex. 1008
U.S. Patent No. 5,644,735 ("Luciw")	Ex. 1009

D. The Alleged Grounds of Unpatentability

Petitioner contends that the challenged claims are unpatentable based on the following specific grounds (Pet. 16):

Reference[s]	Basis	Claim(s) Challenged
LiveDoc/DropZones[1]	§ 102(a)	1-18, 36-56, 93-95, 98, and 101
LiveDoc/DropZones	§ 103(a)	1-18, 36-56, 93-95, 98, and 101
LiveDoc/Drop Zones and Moore	§ 103(a)	1-18, 36-56, 93-95, 98, and 101
Domini	§ 102(e)	1-18, 36-38, 40-45, 49-52, 54-56, 93, 98, and 101

[1] The parties disagree on whether LiveDoc/Drop Zones can be considered a single reference for purposes of anticipation. Pet. 17; Prelim. Resp. 25-26. Because we are not persuaded that the published material, even if considered as a single reference, discloses all limitations of any of the claims that are challenged under § 102 for reasons discussed below, we do not reach the "single reference" issue.

Reference[s]	Basis	Claim(s) Challenged
Hachamovitch	§ 102(e)	1-18, 36-56, 86, 87, 89, 93, 97, 98, 100, and 101
Hachamovitch	§ 103(a)	3-5, 9-11, 15-17, 38-41, 45-48, 53, 88, 90, and 91
Luciw	§ 102(e)	1-18, 36-56, 86-88, 90, 92-94, 97, 98, 100, and 101
Luciw	§ 103(a)	3-5, 9-11, 15-17, 38-41, 45-48, 53, 87, 89, 91, and 95

E. The '854 Patent

The '854 patent, titled "Method, System and Computer Readable Medium for Addressing Handling From a Computer Program," relates to computer implemented processes for providing a computer program, such as a word processing program or spreadsheet program, that is coupled to an information management source, such as a database program or contact management program. Ex. 1001, 1:20-50.

Figures 3 and 4 of the '854 patent are reproduced below.

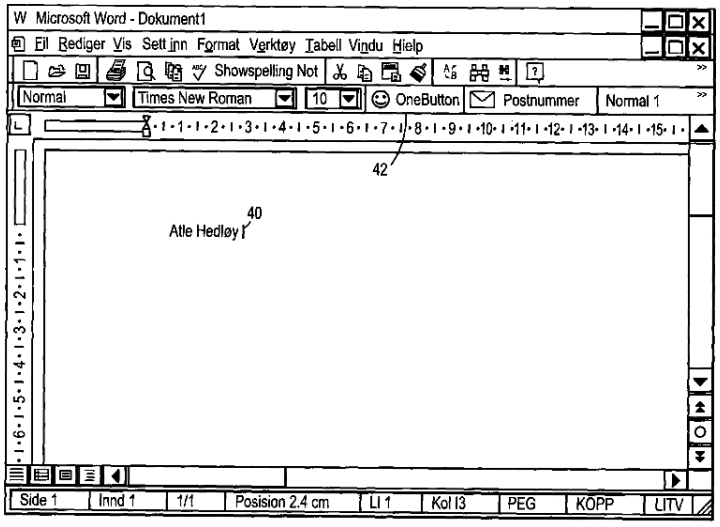


FIG. 3

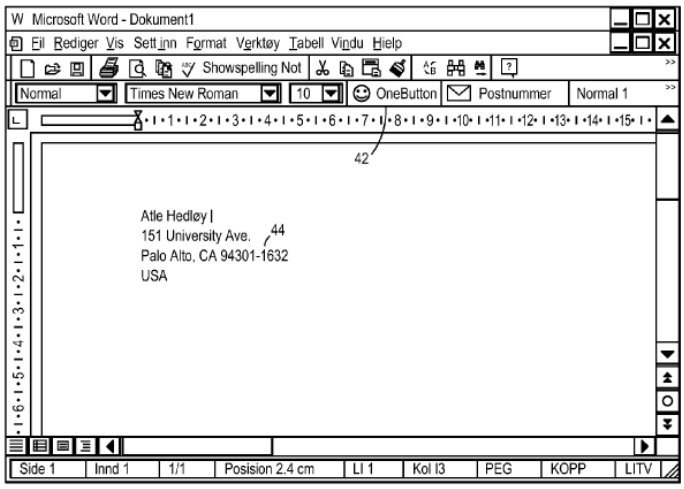


FIG. 4

Figure 3 illustrates the inputting of a name to be searched into a document. Figure 4 illustrates a retrieved address that is inserted into a document. Ex. 1001, 2:51-57. The user types a name into the document. When the user clicks on OneButton 42, the claimed process is launched, retrieving name 40 from the document, searching a database for name 40, and inserting the retrieved address associated with name 40 into the document, as shown in Figure 4. Ex. 1001, 5:60-6:5.

Figure 2 of the ’993 patent, a flow chart illustrating a method for address handling within a computer program, is reproduced below.

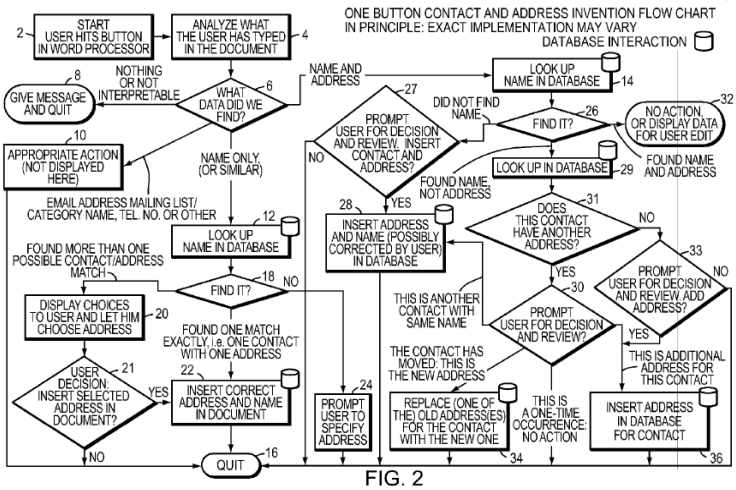


FIG. 2

Figure 2 depicts a flow chart of the address handling process initiated by the user clicking on OneButton 42 of Figure 4. At step 4, text typed by the user in a document is analyzed for contact information. At step 6, if the identified contact information includes a name, a search occurs in the database at step 12. When the database finds a name with more than one possible matching address, the user is prompted for a decision, and that selected information is added to the document at step 22. Ex. 1001, 5:10-22, 6:4-5.

F. Illustrative Claim

Independent claim 1, reproduced below with added paragraphs and roman numerals, is illustrative of the claimed subject matter:

> 1. A method for information handling within a document created using a first application program comprising the steps of:
>
> [i] entering a first information in the first application program;
>
> [ii] marking without user intervention the first information to alert the user that the first information can be utilized in a second application program; and

> [iii] responding to a user selection by inserting a second information into the document, the second information associated with the first information from a second application program.

II. ANALYSIS

A. Claim Construction

We determine the meaning of the claims as the first step of our analysis. The Board interprets claims using the broadest reasonable construction. *See* 37 C.F.R. § 100(b); Office Patent Trial Practice Guide, 77 Fed. Reg. 48756, 48766 (Aug. 14, 2012). Claim terms, generally, are given their ordinary and customary meaning, as would be understood by one of ordinary skill in the art in the context of the entire disclosure. *See In re Translogic Tech., Inc.*, 504 F.3d 1249, 1257 (Fed. Cir. 2007). If an inventor acts as his or her own lexicographer, the definition must be set forth in the specification with reasonable clarity, deliberateness, and precision. *Renishaw PLC v. Marposs Societa' per Azioni*, 158 F.3d 1243, 1249 (Fed. Cir. 1998).

1. "marking . . . the first information to alert the user"

The claim term "marking without user intervention the first information to alert the user" appears in independent claims 1, 7, and 13. Petitioner notes that the term "marking" does not appear in the '854 patent specification. Pet. 7. Petitioner argues that the plain meaning of the claim term "is that the first information is detected without user intervention and has some form of marking or highlighting applied to it to draw the user's attention to it." Pet. 7 (citing Declaration of Menasce, Ex. 1002 ¶¶ 49-50). Petitioner asserts that during prosecution the Applicant explained the support for the claimed "marking" by stating that marking could be done in a variety of ways including displaying the text to the user in a

separate screen. Pet. 7 (citing Ex. 1004, 30-31); *see* Ex. 1001, 7:1-14 (marking by generating a screen or dialog window). Thus, Petitioner argues that the marking could be accomplished by presenting the marked information in a separate window.

Patent Owner provides no proposed construction or argument regarding "marking … the first information to alert the user." *See* Prelim. Resp. 7-9. During prosecution of the application that matured into the '854 patent, the then Applicant explained that the claim term "marking" found support in the specification, referencing methods of marking that include generating a separate screen. Ex. 1004, 30-31. The Applicant also provided a dictionary definition of marking as "to pick out or designate something . . . as special in some way." Ex. 1004, 30. For purposes of this institution decision, "marking" encompasses highlighting, designating or displaying the information in a separate screen or window to draw a user's attention. We also determine, based on the present record, that the claim term "marking without user intervention the first information to alert the user" means that the first information is detected and has some form of marking applied to it without user intervention.

2. Means-Plus-Function Limitations

Section 112, ¶ 6[2] permits an element in a claim for a combination to be expressed as a means for performing a specified function without the recital of structure in support thereof, but with the provision that "such claim shall be construed to cover the corresponding structure, material, or acts described in the

[2] Section 4(c) of the Leahy-Smith America Invents Act, Pub. L. No. 112-29, 125 Stat. 284 (2011) ("AIA"), re-designated 35 U.S.C. § 112, ¶ 6, as 35 U.S.C. § 112(f). Because the '854 patent has a filing date before September 16, 2012 (effective date of AIA), we use the citation § 112, ¶ 6.

specification and equivalents thereof." "[T]he corresponding structure for a § 112 ¶ 6 claim for a computer-implemented function is the algorithm disclosed in the specification." *Aristocrat Techs. Austl. Party. Ltd. vs. Int'l Game Tech*., 521 F.3d 1328, 1333 (Fed. Cir. 2008) (quoting *Harris Corp. v. Ericsson Inc*., 417 F.3d 1241, 1249 (Fed. Cir. 2005)).

Petitioner asserts that the claims with means-plus-function limitations, specifically claims 13-18, 50-56, 100 and 101 lack corresponding structure or algorithms as required under Section 112, ¶ 6. Pet. 8-15 (providing charts and citations showing means-plus-function limitations with missing structure or algorithms). Patent Owner provides no discussion or argument indicating the algorithms or structures corresponding to the means-plus-function limitations of claims 13-18, 50-56, 100 and 101.

We agree with Petitioner that independent claims 13, 50, 100, and 101 lack corresponding structure and algorithms as required for computer-implemented functions. Thus, we are unable to interpret independent claims 13, 50, 100, and 101 and dependent claims 14-18 and 51-56 due to the lack of disclosed structures. A lack of sufficient disclosure of structure under Section 112, ¶ 6 renders a claim indefinite. *See Blackboard, Inc. v. Desire2Learn, Inc*., 574 F.3d 1371, 1382 (Fed. Cir. 2009).

3. "associated"

"Associated" appears in the claim term "second information associated with the first information from a second application program" recited in claim 1. Patent Owner contends that "associated" should be construed as meaning "a pre-existing connection or relationship." Prelim. Resp. 8-9. Patent Owner claims this is the term's ordinary and customary meaning in accordance with the intrinsic evidence

that shows searching for information that is "related" to text. *Id*. (citing Ex. 1001, Abstract, 3:63-66, 5:66-6:2, 4:43-45, 4:57-58). Petitioner offers no construction for this term.

We are not persuaded by Patent Owner that "associated," as claimed, should be narrowly construed as requiring "a pre-existing" connection or relationship. Indeed, the '854 specification refers to related information that possibly may match the searched data or data that corresponds to part of a typed name. *See* Ex. 1001, 3:63-67, 4:43-58. Based on the record before us, we determine that "associated" is construed as "connected or related."

4. "application program"

Patent Owner argues that "application program," which appears in independent claims 1, 7, 13, 36, 43, and 50, should be construed as "an independently executable computer program designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." Prelim. Resp. 9. Patent Owner asserts that the term as used in the patent specification refers to software products, such as Microsoft Word and Excel. *Id*.

The term "application program" does not appear in the specification of the '854 patent. However, we are not persuaded by Patent Owner that "application program" is limited to programs "designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." The '854 patent expressly states that the invention is not limited to such programs. Ex. 1001, 9:64-10:10.

Based on the record before us, we determine that "application program" encompasses an independent executable program.

B. Asserted Unpatentability Based on LiveDoc/Drop Zones (Ex. 1005)

LiveDoc discloses a structure detection process running in the background on the visible text of a document entered by a user. LiveDoc's analyses are presented to the user through automatic visual highlighting of discovered structures. Ex. 1005, 55. Certain meaningful portions of a document are automatically highlighted in LiveDoc mode, and clicking on them causes certain actions to occur. *Id*.

Drop Zones is titled, "Drop Zones – An Extension to LiveDoc" and discloses a user interface for managing LiveDoc objects in the context of a set of typical user tasks. Pet. 18. A Drop Zones "assistant" takes features identified by LiveDoc, interprets the feature's meaning, and recommends appropriate actions, made visible when the user selects structures identified by LiveDoc, and drags and drops them to the assistants. Ex. 1005, 60.

1. Anticipation Based on LiveDoc/Drop Zones (Ex. 1005)

Petitioner contends that claims 1-18, 36-56, 93-95, 98, and 101 are anticipated by LiveDoc/Drop Zones. Pet. 19-27. Petitioner provides claims charts listing the limitations of claims 1-6, 36-42, 93-95 and corresponding citations to LiveDoc/Drop Zones. Pet. 17-25. Patent Owner argues that LiveDoc/Drop Zones fails to teach or suggest "responding to a user selection by inserting a second information into the document" as recited in claim 1 and similar limitations in independent claims 7, 13, 36, 43, and 50. Prelim. Resp. 27. Similarly, claims 93, 98, and 101 recite "inserting" information into the document. Patent Owner argues that highlighting and handling of the information in the LiveDoc system takes place outside of the document and an overlay is used to present information. *Id*. at 27-30. Patent Owner argues that the portion of the printed publication relied on by

Petitioner does not teach insertion into the document, but instead refers to future improvements to the LiveDoc program that would allow the overlay information to be placed "within the context of the document." Pet. 30-31 (quoting Ex. 1005 at 58).

Based on the record before us, we agree with Patent Owner that Petitioner has not persuasively shown that LiveDoc/Drop Zones describes inserting the highlighted information or the second information into the document as recited in claim 1 and related claims. *See* Pet. 20-21(claim chart referencing LiveDoc for insertion) 26-28 (discussing obviousness). The portion of LiveDoc that Petitioner relies on is under the subheading "Futures: Extensibility and Semantics," and describes placing the second information returned by the LiveDoc program "within the context of the document itself, rather than in a separate application," as an imagined future improvement to the existing LiveDoc functions. Ex. 1005 at 58. Moreover, Petitioner has not explained how placing within the context of the document encompasses the claimed "inserting" by a user. As Patent Owner argues (Prelim. Resp. 30), placing the returned information within the context of the document appears to refer to not relying on a separate or external application program to view the rendered information. Petitioner's contentions and analysis for the "inserting" limitation have not shown persuasively that LiveDoc/Drop Zones describes "responding to a user selection by inserting a second information into the document" as recited in independent claim 1 and related independent claims 7, 13, 36, 43, 50, or "inserting" into the document as recited in independent claims 93, 98, and 101.

With respect to means-plus-function limitations in claims 13, 50, 100, and 101 and dependent claims 14-18 and 51-56, we are unable to interpret independent claims due to the lack of disclosed structures as discussed above in Section II.A.2.

Thus, we also conclude that the information presented in the Petition and the Preliminary Response does not show there is a reasonable likelihood that Petitioner would prevail in its challenge of claims 13-18, 50-56, 100 and 101, which contain means-plus-function limitations.

Based on the record before us, we do not find that there is a reasonable likelihood that Petitioner will prevail in showing that claims 1-18, 36-56, 93-95, 98, and 101 are obvious over LiveDoc/Drop Zone.

2. Obviousness Based on LiveDoc/Drop Zones (Ex. 1005)

Petitioner contends that claims 1-18, 36-56, 93-95, 98, and 101 are obvious in view of LiveDoc/Drop Zones. Pet. 19-26.

For reasons discussed above, we agree with Patent Owner that Petitioner has not persuasively shown that LiveDoc/Drop Zones teaches inserting the highlighted information or the second information into the document as recited in claim 1 and related claims. *See* Pet. 20-21(claim chart referencing LiveDoc for insertion) 26-28 (discussing obviousness). Moreover, LiveDoc expressly states that "more sophisticated models of documents are needed" to pursue the imagined future improvements. Ex. 1005, 10. The evidence, thus, tends to show that one of ordinary skill in the art would not have been capable of modifying LiveDoc/Drop Zones consistent with the contemplation of presenting a rendering within the context of the document itself. *See id.*

Based on the record before us, we do not find that there is a reasonable likelihood that Petitioner will prevail in showing that claims 1-18, 36-56, 93-95, 98, and 101 are obvious over LiveDoc/Drop Zones.

3. Obviousness Based on LiveDoc/Drop Zones (Ex. 1005) and Moore (Ex. 1006)

Petitioner contends that claims 1-18, 36-56, 93-95, 98, and 101 are obvious in view of LiveDoc/Drop Zones. Pet. 27-28. Petitioner relies on Moore for additional claim limitations involving database application not previously argued as being disclosed in LiveDoc/Drop Zones. For the same reasons discussed above with respect to obviousness based on LiveDoc/Drop Zones, we do not find that there is a reasonable likelihood that Petitioner will prevail in showing that claims 1-18, 36-56, 93-95, 98, and 101 are obvious over LiveDoc/Drop Zones and Moore.

C. Asserted Unpatentability Based on Domini (Ex. 1007)

Petitioner contends that Domini anticipates claims 1-18, 36-38, 40-45, 49-52, 54-56, 93, 98, and 101 under 35 U.S.C. § 102(e). Pet. 31-39.

1. Domini (Ex. 1007)

Domini discloses identifying and correcting spelling errors in a document created by a word processing program. Ex. 1006, Abstract; 4:65-5:11. The user selects the "spelling and grammar" command to initialize the spell check program. *Id*. at 16:13-16. Without user intervention, the spell check program identifies misspelled words and presents them in red, bold typeface. *Id*. at 17:27-33; 4:12-16. The spell check program also displays a list of suggested corrections that may be selected and entered into the document by the user. *Id*. at 1:42-44; 12:1-5; 12:61-64.

2. Anticipation based on Domini (Ex. 1007)

Petitioner asserts that claims 1-18, 36-38, 40-45, 49-52, 54-56, 93, 98, and 101 are anticipated by Domini. Pet. 28-36. Petitioner's claim chart provides citation to Domini, which Petitioner contends discloses the corresponding claim

limitations in method claims 1-6, 36-42, and 93. *Id*. at 29-35. Petitioner argues that computer readable medium claims 7-12, 43-45, 49, and 98, and system claims 13-18, 50-52, 54-56, and 101, which include means-plus function limitations, are anticipated for the same reasons cited for their related method claims. *Id*. at 35. However, we determined above that the means-plus-function limitations of claims 13-18, 50-52, 54-56, and 101 are indefinite and that Petitioner would not likely prevail with respect to those claims. Thus, we address Petitioner's grounds based on Domini as they apply to claims 1-12, 36-38, 40-45, 49, 93, and 98.

Petitioner asserts that the spell checking and correction method in Domini, which identifies words, marks and alerts them to the user and allows a user to select from suggested corrections or add a word to the dictionary, correspond to the limitations of independent claims 1, 7, 36, 43, 93, and 98. Pet. 29-36.

Patent Owner contends that Domini fails to disclose a second application program as recited in independent claims 1, 7, 13, 36, 43, and 50. Prelim. Resp. 33-36. Although Domini discloses a preferred embodiment that uses program modules within a program (*see* Prelim. Resp. 35 (citing Ex. 1007, 5:1-8)), Domini also discloses that the system could run in a distributed computing environment with program modules located and executed locally and remotely. Ex. 1007, 5:15-27. Indeed, we are persuaded by Petitioner that Domini discloses multiple application programs in Figure 1, which identifies word processing program 37A and spell checker program 37B separately. Pet. 30 (citing Ex. 1006, Fig 1). Domini also refers expressly to these "application programs 37" as "different programs." Ex. 1006, 7:41-51.

Patent Owner also contends that "Domini fails to 'search' for 'related information' in a 'database' based upon text identified in a document as required at least by independent claims 93, 98, and 101." Pet. 36. Patent Owner asserts that

the spell checking modules in Domini present words based on what the user intended to type and not on a pre-existing relationship. Pet. 37. We disagree. As discussed above with the term "associated," we find that searching for related information as recited in the claims does not require a pre-existing relationship.

We are persuaded that Petitioner has shown that Domini discloses selecting words by comparing the word in the document to words in standard and custom dictionaries. *See* Pet. 28-32; *see* Ex. 1006, 17:27-57. Domini discloses that the word being checked "corresponds" to words in the dictionaries. Ex. 1006, 17:47-49. Thus, we disagree with Patent Owner that there is no relationship between the words in the document and the words in the dictionaries.

Patent Owner contends that Domini fails to disclose "a second computer program" as required by independent claims 93 and 98 because Petitioner Domini fails to teach a "'second computer program' distinct from a 'first computer program' and a database." Prelim. Resp. 38. Patent Owner asserts that Petitioner appears to rely on the spell checker with dictionaries to be both the "first computer program" and the "second computer program" as recited in claims 93 and 98. Petitioner's citations to Domini refer to the spell check program as corresponding to the first program and the spell check program searching a dictionary as the second program. Pet 34-35. We are not persuaded by Petitioner's argument and evidence that the spell checker with dictionaries is both the first and second programs of claim 93 and 98. Thus, Petitioner has not shown a reasonable likelihood that it will prevail as to claims 93 and 98.

Based on the record before us, Petitioner has shown a reasonable likelihood that it will prevail as to claims 1-12, 36-38, 40-45, and 49 as anticipated by Domini.

D. Asserted Unpatentability Based on Hachamovitch (Ex. 1008)

Petitioner contends that Hachamovitch anticipates claims 1-18, 36-56, 86, 87, 89, 93, 97, 98, 100, and 101 under 35 U.S.C. § 102(e). Pet. 40-45. In addition, Petitioner contends that claims 3-5, 9-11, 15-17, 38-41, 45-48, 53, 88, 90, and 91 are obvious in view of Hachamovitch. Pet. 45-49.

1. Hachamovitch (Ex. 1008)

Hachomovitch discloses a word completion utility that automatically can predict for data entry in a data file, such as a word processor or email application. Ex. 1008, 4:10-13. The word completion system is used in conjunction with an individual application program or operates independently across multiple application programs. *Id*. at 4:21-25. The user's partially typed word is compared to the name-completion pairs and if a match is found within the list a suggestion list will be presented to a user. *Id*. at 4:58-5:6. A suggested word is presented to the user in a pop-up user interface within a word processing application as shown in Figs. 2A and 2B below.

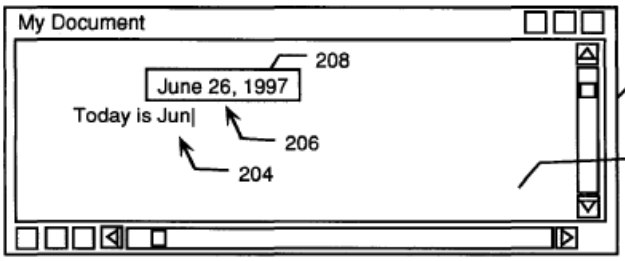


FIG. 2A

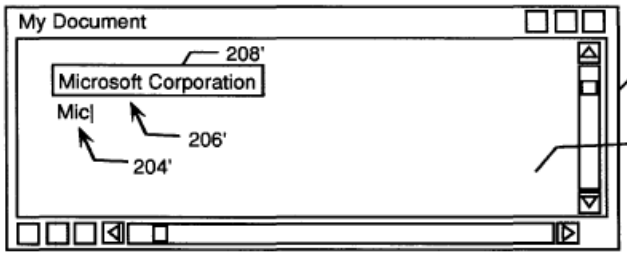


FIG. 2B

Figure 2A illustrates a word completion suggestion where the name completion pair is tied to a date or system parameter. *Id*. at 10:18-21; 10: 57-61. Figure 2B illustrates a word completion suggestion where the name-completion pair is tied to predefined properties, such as an initial capitalized letter. *Id*. at 11: 4-14. Once a word completion is displayed, the user may accept the word completion using an acceptance keystroke (e.g. tab or enter). Once accepted, the word completion utility replaces the partial data entry with the completion entry in the data file. *Id*. at 5:7-10; 7:4-5.

2. Anticipation Based on Hachamovitch (Ex. 1008)

Petitioner asserts that Hachamovitch discloses the limitations of claims 1-18, 36-56, 86, 87, 89, 93, 97, 98, 100, and 101. Pet. 16, 37-44. Petitioner provides claims charts indicating which portions of Hachamovitch disclose the limitations of claims 1-6, 36-42, 86, 87, 89, and 93. *Id*. at 37-44. Petitioner argues that the computer readable medium claims 7-12, 43-49, 97, and 98 are anticipated for the same reasons as those cited for method claims 1-6, 36-42, 86 and 93. *Id*. at 44.

Patent Owner argues that Miller fails to disclose "a second application program" as recited in independent claims 1, 7, 13, 36, 43 and 50. Petitioner contends that Hachamovitch teaches that the word completion utility can be deployed as a "stand-alone" "application-independent" utility. Pet. 38 (Ex. 1008, 7:65-8:5). Patent Owner contends, however, that Hachamovitch discloses that "application independence" refers to a utility that is accessible through different programs but does not contemplate that the utility is a "standalone executable program." Ex. 1008, 8:6-9. Hachamovitch states that "[t]o deploy the word completion system as an application-independent utility, an interface is defined within each application program through which the word completion utility may

communicate with each application program." Ex. 1008, 8:6-9. Patent Owner argues that Hachamovitch describes a utility that can be accessed via other application programs and does not operate independent of the application program. Prelim. Resp. 42-43.

We are persuaded by Petitioner's argument and evidence that Hachamovitch discloses a stand-alone utility that operates with various application programs. Pet. 37-43. Although the utility described in Hachamovitch operates through the application program, Hachamovitch also states that the utility can be part of the application program or operate independent of an application as a "stand-alone" utility. Based on the record before us, we are persuaded that the stand-alone utility operating through the application programs interface corresponds to an application program as recited in independent claims 1, 7, 13, 36, 43 and 50.

Patent Owner also contends that Hachamovitch fails to disclose "a second computer program" as recited in independent claims 86, 93, 97, 98, 100 and 101. Prelim. Resp. 43-45. Independent claims 86, 93, 97, 98, 100 and 101 require a first computer program to analyze the document and a second computer program to search the database and locate related information. Patent Owner argues that Petitioner equates the word completion program as the first program and refers to the database or the word completion program as the second program. *Id*. We agree with Patent Owner.

With respect to claim 86, Petitioner asserts that the auto-complete includes the first computer program, and the database searched by the auto-complete is the second computer program. Pet. 42 (claim 86c and 86d limitations). We are not persuaded by Petitioner's evidence that the database searched by the auto-complete utility is a first computer program as recited in the claims. Petitioner has not provided sufficient explanation or evidence that the database searched by the utility

is a second computer program that is executed or used to search the database as recited in the claims. Pet. 42.

Based on the foregoing, we find that Petitioner has not shown a reasonable likelihood that it will prevail as to independent claims 86, 93, 97, 98, 100 and 101 and dependent claims 87 and 89.

Patent Owner also contends that Hachamovitch fails to disclose "marking without user intervention the first information *to alert the user that the first information can be utilized in a second application program*" as recited in independent claims 1, 7 and 13. Prelim. Resp. 45. Because the marking in the word completion utility occurs after the user has entered partial data, Patent Owner asserts that the marking that occurs indicates that the suggested completion has already been retrieved. Prelim. Resp. 45-46. In sum, because the suggested completion terms are already displayed in Hachamovitch, the first information of the claim "has already been used" to search and provide a suggestion. *Id*. at 46. We disagree.

We are persuaded by Petitioner's evidence that Hachamovitch discloses marking the first information and indicates that the information can be used in a second application program, the utility. Pet. 37-38. Claim 1 does not require that the use of the first information be *after* the marking occurs. Based on the record before us, we find that Hachamovitch's displaying of suggested terms as the user enters data indicates that information "can be used" by the word completion utility by showing the results of a search. Thus, we are persuaded by Petitioner on the record before us that Hachamovitch discloses "marking without user intervention the first information to alert the user that the first information can be utilized in a second application program" as recited in independent claims 1, 7 and 13.

In addition to the findings above, we also determined in Section II.A.2 that the means-plus-function limitations of claims 13-18, 50-56, 100 and 101 lack sufficient structure and are indefinite. Furthermore, we found above that Petitioner would not likely prevail with respect to claims 86, 87, 89, 93, 97, 98, 100, and 101. Based on the foregoing, we do not find that there is a reasonable likelihood that Petitioner will prevail in showing that claims 13-18, 50-56, 86, 87, 89, 93, 97, 98, 100, and 101 are anticipated by Hachamovitch.

On the record before us, we find that there is a reasonable likelihood that Petitioner will prevail in showing that claims 1-12 and 36-49 are anticipated by Hachamovitch.

3. Obviousness over Hachamovitch (Ex. 1008)

Petitioner argues that claims 3-5, 9-11, 15-17, 38-41, 45-48, 53, 88, 90, and 91 are obvious in view of Hachamovitch. Pet. 16, 44-46. We find above in Section II.D.2 that there is not a reasonable likelihood that Petitioner will prevail in showing that claims 13-18, 50-56, 86, 87, 89, 93, 97, 98, 100, and 101 are disclosed by Hachamovitch. For those reasons discussed above, we are not persuaded by Petitioner's arguments that Hachamovitch teaches or suggests claims 15-17, 53, 88, 90, and 91.

With respect to claims 3-5, 9-11, 38-41, and 45-48, Patent Owner argues and provides testimony that it would have been obvious to initialize the second application program and search using the second application program for the second information as recited in claims 3, 9, 38 and 45 and their respective dependent claims. Pet. 44 (citing Ex. 1002 ¶ 161). We are persuaded by Petitioner's evidence and argument regarding initializing the second application program and using it for searching as recited in claims 3-5, 9-11, 38-41, and 45-48. Based on the record before us, we find that there is a reasonable likelihood that

Petitioner will prevail in showing that claims 3-5, 9-11, 38-41, and 45-48 are obvious in view of Hachamovitch.

E. Asserted Unpatentability Based on Luciw (Ex. 1009)

Petitioner contends that Luciw anticipates claims 1-18, 36-56, 86-88, 90, 92-94, 97, 98, 100, and 101 under 35 U.S.C. § 102(e). Pet. 16, 46-57. Petitioner also contends that claims 3-5, 9-11, 15-17, 38-41, 45-48, 53, 87, 89, 91, and 95 are obvious in view of Luciw. *Id*.

1. Luciw (Ex. 1009)

Luciw discloses a pen-based, handheld device that provides user assistance based on information entered into a document, such as a note area created by a notepad application. Ex. 1008, 2:19-22; 6:24-59. When the user writes certain information, such as a name, it is recognized automatically and converted to formal font form. *Id*. at Fig. 2; 3:8-10; 10:10-21; 11:43-45. Luciw allows the user to make a selection from a list of persons with the identified name; when the user makes a selection, information associated with the person, such as the person's full name, is inserted into the document. *Id*. at 11:60-12:6.

2. Anticipation based on Luciw (Ex. 1009)

Patent Owner argues that Luciw fails to disclose a "second application program" as recited in independent claims 1, 7, 13, 36, 43, and 50. Prelim. Resp. 46-49.

Petitioner provides claim charts indicating which portions of Luciw correspond to the identified claims. Pet. 47-54. However, Petitioner fails to identify which program or operation corresponds to the "second application program" of claim 1. Pet. 47-48 (claim 1c). In addition, Petitioner identifies the database of Luciw as the "second application program" for claim 86 and the

second computer program of claim 93, but fails to explain how the database in Luciw corresponds to a computer application program. *Id*. at 53 (claim 86c), 54 (claim 93c). We agree with Patent Owner that Luciw discloses a single application program and a computer database. Prelim. Resp. 47-48; *see* Ex. 1009, 6:55-59. Petitioner provides no citation or argument that shows the database in Luciw is a second program as recited in the claims.

Based on the record before us, we do not find that there is a reasonable likelihood that Petitioner will prevail in showing that claims 1-18, 36-56, 86-88, 90, 92-94, 97, 98, 100, and 101 are anticipated by Luciw or that claims 3-5, 9-11, 15-17, 38-41, 45-48, 53, 87, 89, 91, and 95 are obvious in view of Luciw.

III. CONCLUSION

For the foregoing reasons, we determine that the information presented in the petition establishes that there is a reasonable likelihood Petitioner would prevail in challenging claims 1-12 and 36-49 of the '854 patent on the grounds that:

(1) Claims 1-12, 36-38, 40-45, and 49 are unpatentable under 35 U.S.C. § 102(e) over Domini;

(2) Claims 1-12 and 36-49 are unpatentable under 35 U.S.C. § 102(e) over Hachamovitch; and

(3) Claims 3-5, 9-11, 38-41, and 45-48 are unpatentable under 35 U.S.C. § 103(a) in view of Hachamovitch.

The Board has not made a final determination on the patentability of any challenged claims.

IV. ORDER

For the reasons given, it is

ORDERED that *inter parties* review is instituted as to claims 1-12 and 36-49 of the ’854 patent on the grounds that:

(1) Claims 1-12, 36-38, 40-45, and 49 are unpatentable under 35 U.S.C. § 102(e) over Domini;

(2) Claims 1-12 and 36-49 are unpatentable under 35 U.S.C. § 102(e) over Hachamovitch; and

(3) Claims 3-5, 9-11, 38-41, and 45-48 are unpatentable under 35 U.S.C. § 103(a) in view of Hachamovitch;

FURTHER ORDERED that pursuant to 35 U.S.C. § 314(a), *inter partes* review of the ’854 patent is hereby instituted commencing on the entry date of this Order, and pursuant to 35 U.S.C. § 314(c) and 37 C.F.R. § 42.4, notice is hereby given of the institution of a trial;

FURTHER ORDERED that all other grounds are *denied*, and no ground other than those specifically granted above is authorized for the *inter partes* review as to the ’854 patent.

Petitioner:

David Fehrman
dfehrman@mofo.com

Mehran Arjomand
marjomand@mofo.com

Matthew Smith
smith@turnerboyd.com

Zhuanjia Gu
gu@turnerboyd.com

Patent Owner:

Robert Asher
rasher@sunsteinlaw.com

Bruce Sunstein
bsunstein@sunsteinlaw.com

Exhibit 6S

UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE PATENT TRIAL AND APPEAL BOARD

_____

Apple Inc., Google Inc., and Motorola Mobility LLC

Petitioners,

v.

Arendi S.A.R.L.

Patent Owner.

_____

Case No. IPR2014-00207

Patent No. 7,496,854

_____

Before HOWARD B. BLANKENSHIP, SALLY C. MEDLEY, and TREVOR M. JEFFERSON, Administrative Patent Judges.

PATENT OWNER ARENDI S.A.R.L.'S RESPONSE UNDER 37 C.F.R. § 42.120

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Cases

Statutes

Regulations

EXHIBIT LIST

Arendi Exhibit Number	Description
2001	American Heritage College dictionary 3rd edition 1997 definition of the term "dictionary." (previously filed with the Patent Owner's Preliminary Response)
2002	American Heritage College dictionary 3rd edition 1997 definition of the terms "independent", "executable", and "program"
2003	Declaration of John V. Levy, Ph.D.

I. INTRODUCTION

Patent Owner Arendi S.A.R.L. ("Arendi" or "Patent Owner") respectfully requests that the Board find that Petitioners Apple Inc., Google Inc., and Motorola Mobility LLC ("Petitioners") have failed to show by a preponderance of the evidence that the claims 1-12, 36-49, of U.S. Patent No. 7,496,854 (the "854 Patent") are either anticipated or made obvious in view of the cited prior art of U.S. Patent No. 6,085,206 ("Domini") and U.S. Patent No. 6,377,965 ("Hachamovitch").

The Board instituted the current *inter partes* review based on the following grounds:

(1) Claims 1-12, 36-38, 40-45, and 49 are unpatentable under 35 U.S.C. § 102(e) over Domini;

(2) Claims 1-12 and 36-49 are unpatentable under 35 U.S.C. § 102(e) over Hachamovitch; and

(3) Claims 3-5, 9-11, 38-41, and 45-48 are unpatentable under 35 U.S.C. § 103(a) in view of Hachamovitch;

However, for each proposed ground, at least one claim element is missing from the relied-upon reference.

II. OVERVIEW OF THE '854 PATENT

The '854 Patent is directed, among other things, to computer-implemented processes for automating a user's interaction between a first application, such as a word processing application or spreadsheet application, on the one hand, and a second application, such as contact management application having a database, on the other hand.

The invention as claimed provides a significant simplification of prior art methods. In the prior art, a user who has entered first information (e.g. a person's name) in a document must leave the first application program, (e.g. a word processor) and open and search using a second application program (e.g. contact management program) when the user wishes to locate second information related to the first information from the second application program and to use that information in the first application. The user of such prior art systems must search for the first information (e.g. a name) and the second information (e.g. an address) using the second program and then return to the first program and manually enter the second information into the document. This process requires a plurality of actions by the user in order to obtain related information to the information typed within the document. Thus, according to embodiments of the present invention, "the process of creating and updating records in an address database is

significantly simplified, since this may now be performed directly from the word processor." col. 9, lines 61-63

The '854 refers to multiple examples of application programs such as Microsoft Word™, Microsoft Excel™, NOTEPAD™, WORDPAD™, WORDPERFECT™, QUATROPRO™, AMIPRO™, Microsoft Outlook™, ACCESS™, ORACLE™, DBASE™, RBASE™, and CARDFILE™. col. 9, line 64 to col. 10, line 10.

In the '854 Patent, Exhibit 1001, Figs. 1 and 2 are flow charts showing for these interactions a number of scenarios, which are described from col. 4, line 25 to col. 5, line 57. Further details of the interactions are provided in discussion thereafter of the other figures of the '854 Patent and the discussion includes references back to relevant portions of the flow charts in Figs. 1 and 2. Fig. 1 is reproduced below.

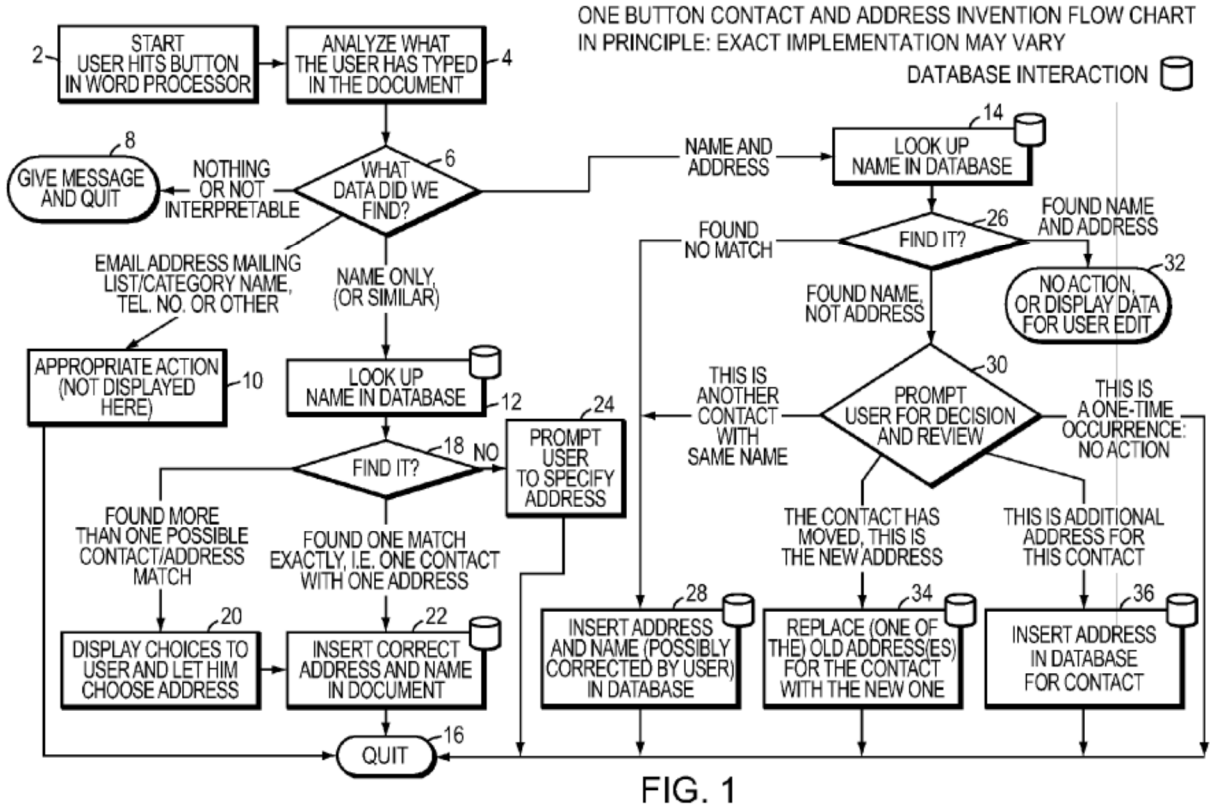


FIG. 1

In various scenarios, text in a document in the first application is analyzed (in step 2 of Fig. 1) to identify first information. Exhibit 1001, col. 4, line 25-49. The analysis takes place without user designation of a specific part of the document to be subject to the analyzing. *Id*.

Once first information has been identified, a number of different scenarios can follow, depending on the circumstances. In one scenario, if the first information includes a name, a search is initiated in the database associated with the second application for the name. *Id*., Fig. 1, steps 6, 12, and 14. If the contact information identified in the document included only a name, and if only a single entry is found in the database for the name and the entry includes a single address,

then the address is inserted into the document. *Id*., Fig. 1, steps 6, 12, 18, and 22; Fig. 4; col. 5, lines 63 to col. 6, line 3. Fig. 4, which is reproduced below, shows the document displayed in Microsoft Word after the address has been inserted.

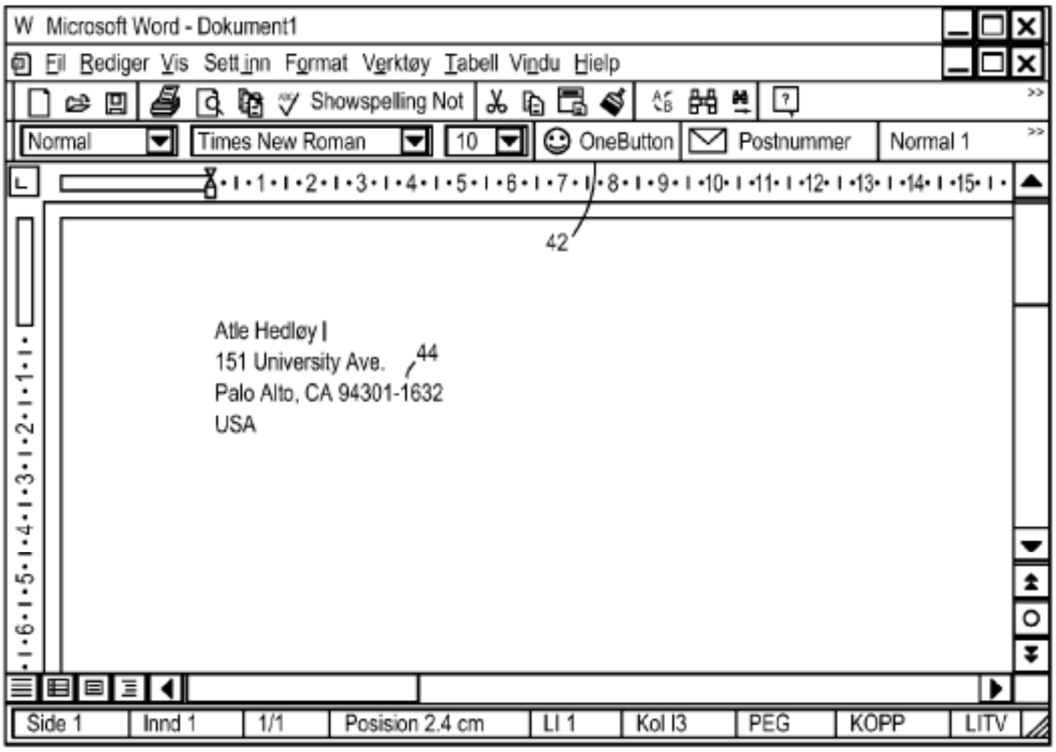


FIG. 4

Shown in Fig. 4 is the One Button 42, which, when pressed, launches the processes just recited, including analyzing the document to identify first information, the searching in the database, and inserting of the address. *Id*., Fig. 2, steps 2, 4; col. 4, lines 25-28; col. 5, lines 58 to col. 6, line 5.

On the other hand, if multiple addresses are found in searching the database for the identified name, these found addresses are displayed, and the user is

presented with a choice of which of the addresses to insert. *Id*., Fig. 1, steps 18, 20, and 22; Fig. 10; col. 7, line 25 to col. 8, line 7.

In another scenario, when the user clicks on the "One Button" after having typed into the document a name and an address, the document is analyzed as before (per Fig. 1, step 4) to identify the name and the address. Next, the database is searched for the identified name (per Fig. 1, step 14). If the name happens to be in the contact database but the address in the contact database for that name differs from the address typed by the user into the document (per Fig. 1, step 26), then the user is prompted to make a choice (per Fig. 1, step 30). The user is presented with

a screen shown in Fig. 9, which is reproduced below.

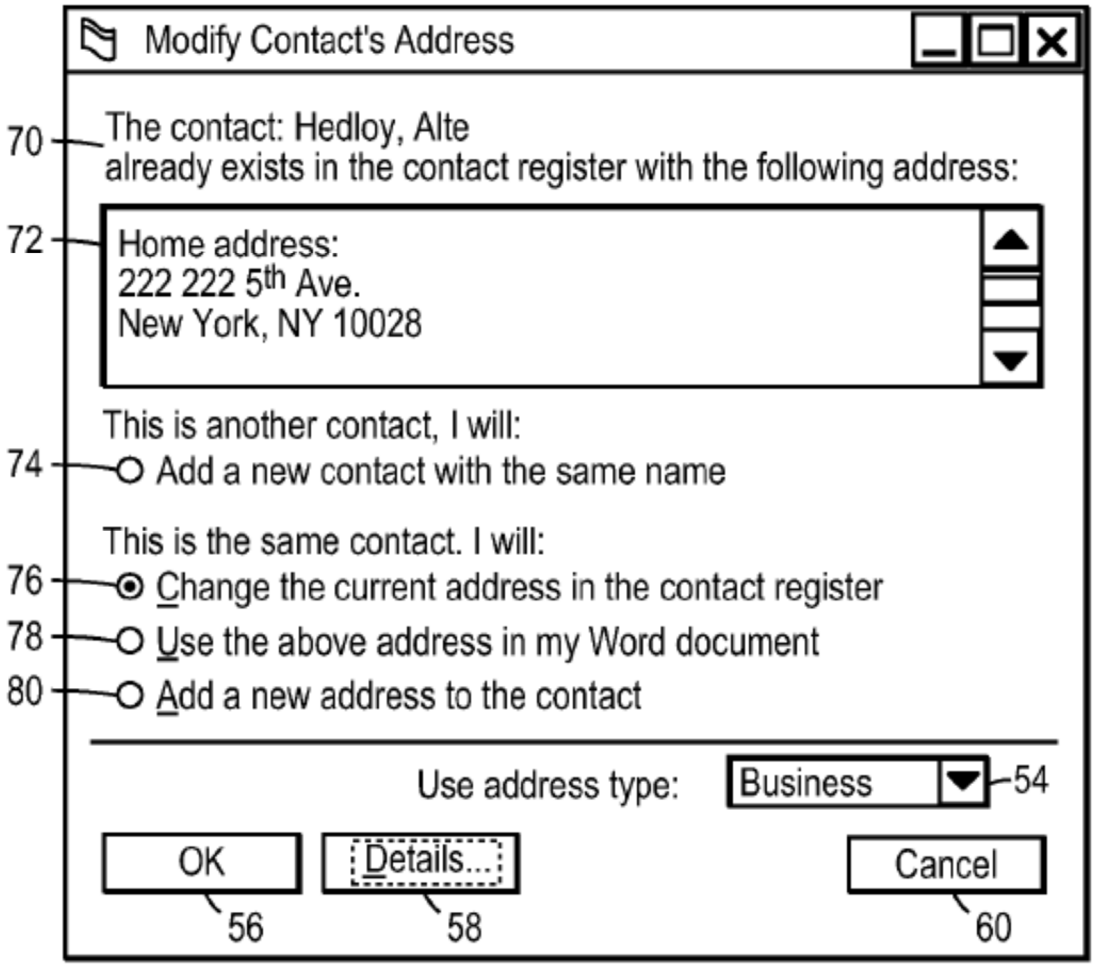


FIG. 9

Fig. 9 represents a screen presented to the user in which the user is given a series of choices that can be made in this specific context. *Id.*, col. 6, line 66 to col. 7, lines 23. The screen reproduces the name that is both in the document and in the contact database, and it also displays the address that is in the contact database for that name. Below this information, the screen offers a total of four choices in two categories. As shown in Fig. 9 and explained in the '854 Patent, the

user is enabled to select one of the four choices. *Id*. The first category is "This is another contact" and the choice under this category is to "Add a new contact with the same name". The second category is that "This is the same contact", and the user is given three other choices for the contact: (a) "Change the current address in the contact register"; (b) "Use the above address [reproduced from the contact database] in my Word document"; and (c) "Add a new address to the contact".

These same four choices are also illustrated in connection with item 30 of Fig. 1 of the '854 Patent, which shows logical flow followed in described embodiments of the invention. Item 30 is labeled "PROMPT USER FOR DECISION AND REVIEW", and there are four outcomes shown from this item: (1) "THIS ANOTHER CONTACT WITH THE SAME NAME"; (2) "THE CONTACT HAS MOVED, THIS IS THE NEW ADDRESS"; (3) "THIS IS A ONE-TIME OCCURRENCE: NO ACTION"; and (4) "THIS IS ADDITIONAL ADDRESS FOR THIS CONTACT". These choices are described in the '854 Patent, col. 4, line 62to col. 6, line 8.

It can be seen that the first of the four choices is to add a new contact, and two of the remaining choices are specific ways of updating an existing contact. (Another choice offered is to do neither of these and simply use the address in the Word document as typed.) Consequently, the screen of Fig. 9 presents to the user

a choice, among other things, between competing alternatives of storing a new contact or updating an existing contact.

III. CLAIM CONSTRUCTION

In an *inter partes* review, the Patent Trial and Appeal Board gives patent claims their "broadest reasonable construction in light of the specification of the patent." 37 C.F.R. § 42.100(b); *Phillips v. AWH Corp.*, 415 F.3d 1303, 1316 (Fed. Cir. 2005) (*en banc*). "To ascertain the scope and meaning of the asserted claims, we look to the words of the claims themselves, the specification, the prosecution history, and, lastly, any relevant extrinsic evidence. *Phillips*, 415 F. 3d at 1315-17, *In re Baxter Int'l*, 678 F. 3d. 1357, 1362 (Fed. Cir. 2012) (Standard on appeal from *ex parte* reexamination.). Extrinsic evidence is relevant only to the extent it is consistent with the specification and file history. *Phillips*, 415 F. 3d at 1316, 1319.

A disputed claim term should be given its plain and ordinary meaning, as the term would have been understood by a person of ordinary skill in the art at the time of the invention. The person of ordinary skill in the art is deemed to have read the claim terms in the context of the "intrinsic evidence" (claims, specification, and prosecution history), *Phillips*, 415 F.3d at 1315. The claims themselves provide substantial guidance in determining the meaning of particular claim terms. *Id*. at 1314. First, a term's context in the asserted claim can be very instructive. *Id.* Other claims can also aid in determining the claim's meaning because claim terms are

typically used consistently throughout the patent. *Id.* Differences among the claim terms can also assist in understanding a term's meaning. *Id.*

The claims "must be read in view of the specification of which they are a part." *Id.* at 1314–15. "[T]he specification 'is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.'"*Id.* at 1315(quoting *Markman v. Westview Instruments Inc*., 52 F.3d 967 (Fed. Cir. 1995) (*en banc*, *affd.* 517 U.S. 370 (1996). "The construction that stays true to the claim language and most naturally aligns with the patent's description of the invention will be, in the end, the correct construction." *Phillips*, 415 F. 3d at 1323

A. Application Program

At page 10 of the Decision by the PTAB for Institution of Inter Partes Review, the PTAB holds that the term "application program" means an "independent[ly] executable program".[1]

[1] In construing the term "application program", the PTAB rejected Patent Owner's construction as "an independently executable computer program designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." [citation], p. 10. The PTAB did not take issue with the requirement that the program be "independently executable. Rather the PTAB stated "we are not persuaded by Patent Owner" that

The American Heritage Collection Dictionary, 3rd Edition 1997 defines the term "independent" to mean "not dependent on…..[a] controlling group or system" and "free from the influence guidance, or control of another or others; self-reliant." This dictionary also defines the term "executable" to mean, "to run (a program or an instruction)" and defines the term "program" to mean, "a procedure for solving a problem that involves collection of data, processing, and presentation of results." Exhibit 2002, American Heritage Collection Dictionary, 3rd Edition 1997. Thus, the Patent Owner agrees with the PTAB's interpretation of an application program as an independently executable program.

a. Application Programs are Independently Executable

The Patent Owner provides additional support from the specification and the Patent Owner's expert for the claim construction of an "application program" as an independently executable program.

"application program" is limited to programs "designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." The PTAB retained the requirement that an "application program" is "independently executable". Patent Owner therefore infers that the absence of "-ly" in "independent" is a typographical error.

The background section of the specification provides guidance as to the meaning of "application program." Col 1, lines 34-37 of the '854 patent states:

> Typically, the information is retrieved by the user from an information management source external to the word processor, such as a database program, contact management program, etc.,

And col. 1, lines 45-46 states:

> This requires the user to learn how to use and have access to the database.

Both of the above passages from the background make it clear that the problem being solved by the inventor and claimed in '854 patent relates to obtaining information from an information management program (a second application program) separate from the word processor in which the user is viewing the document (1st program), and the information management program could be used (and learned) by the user independently from the word processor. Thus, these passages do not suggest that there is a subsidiary relationship between the application programs as would be found between an application program and a module or utility.

The specification also provides guidance as to the meaning of an "application program" in the form of examples, and the definition of "application program" can be construed by the commonly shared features of the computer programs articulated in the examples. See Exhibit 2003 the Declaration of Dr. John Levy ("Levy Decl."), ¶¶ 42-43.

The specification explicitly refers to a "Spreadsheet Application" at column 8, line 55. Other application programs are described in the specification as shall now be detailed.

The specification recites seven separate examples of the invention between col. 5 and col. 9. In six of the examples, Microsoft Word is referenced as the application in which a user enters first information. In the seventh example, first information is entered in a Microsoft Excel spreadsheet. Thus, the application programs include at least Microsoft Word and Microsoft Excel. In each of the seven examples, a database program, such as Microsoft Outlook is accessed. The database program searches a related database for the first information to locate associated second information in one or more database entries. Thus, in addition to Microsoft Word and Microsoft Excel, the inventor has also contemplated at least Microsoft Outlook as an "application program" as would be understood by one of ordinary skill in the art.

The passage running from col. 9, line 64 through col. 10, line 10 of the '854 patent further defines the intended scope of the invention according to the inventor. In this passage, the Patent Owner states that the invention is not limited to Microsoft Word documents and Excel Spreadsheets, but is applicable to all types of word processing documents. The Patent then lists a plurality of different word processing and spreadsheet applications including NOTEPAD™, WORDPAD™,

WORDPERFECT™, QUATROPRO™, AMIPRO™, etc. Based on the recitation of programs, as contemplated by the inventor, "application programs" include word processing programs and spreadsheet programs of the type articulated.

At col. 10, the specification states:

> Although the present invention is defined in terms of information management or is database programs, such as OUTLOOK ™, etc., the present invention is applicable to all types of information management or database programs such as ACCESS™, ORACLE™, DBASE™, RBASE™, CARDFILE ™, including "flat files," etc., as will be readily apparent to those skilled in the art.

In this passage, the inventor defines the scope of his invention as further relating to information management programs and database programs. Thus, the inventor contemplated that database programs and information management programs as types of application programs. The Patent Owner does not make any statements to suggest that a subsidiary program, called by and run under control of an application program, is itself an application program.

Given that the specification identifies word processors, spreadsheet programs, information management programs and database programs as examples of application programs, the definition of an "application program" can be construed from the commonly shared features. The common features of each of these examples are that they each are computer programs that can be executed

independently and none of the example programs runs synchronously under the control of a separate application program. Dr. Levy states in his declaration, that one of ordinary skill in the art reading the '854 patent would understand that the exemplary computer programs mentioned in the detailed description including "word processors, spreadsheet programs, information management programs and database programs" are application programs. Exhibit 2003, Levy Decl. ¶43.

b. Subsidiary Programs are not Independently Executable and therefore, are not Application Programs.

Dr. Levy contrasts application programs with subsidiary programs such as modules and utilities. Subsidiary programs extend the functionality of the controlling application program. In order for the functionality of a subsidiary program to be useful, an application program must first be launched and the subsidiary program's functionality activated within the application program. *Id.,* ¶ 44.

The Patent Owner also notes that in IPR2014-00452 directed to U.S. Patent No. 6,323,853 that shares the same specification as the '854 patent, the PTAB found that "application program" should mean "an independently executable computer program designed to assist in the performing of a specific task, such as word processing or spreadsheet processing." *Google Inc. and Motorola Mobility LLC v. Arendi S.A.R.L.,* IPR2014-00452, Paper 10. In its findings, the PTAB said: "The evidence cited by Patent Owner provides support for Patent Owner's claim

construction argument. Additionally, upon reviewing the specification of the '853 patent, we do not find any disclosure that provides an explicit definition of "application program" contradicting Patent Owner's proposed claim construction."

The Patent Owner therefore construes the term "application program" to mean "an independently executable computer program."

B. "The second information associated with the first information from the second application program"

The term "associated" indicates that there is "an association between" the first information and the second information such that the second information can be found as the result of a search using the first information. "Associated" is used in independent claim 1 in the limitation: "inserting a second information into the document, the second information associated with the first information from a second application program." "Associated" also appears in dependent claim 3 in the limitation "searching, using the second application program, for the second information associated with the first information." The Federal Circuit has made clear that terms should be construed consistently in the same patent. *Am. Permahedge, Inc. v. Barcana, Inc.,* 105 F.3d 1441, 1446 (Fed. Cir. 1997). Thus, the term "associated" must share the same meaning in claims 1 and 3. As stated in Philips, other claims are useful in determining a claim's meaning. *Phillips.*, 415 F.3d at 1315 . Thus, we look to claim 3 for clarification of the term "associated" in

the context of "the second information associated with the first information from the second application program."

Claim 3 places the term associated in the context of "searching." Thus, the term "associated" indicates that there is "an association between" the first and second information that can be found as the result of a search. Additionally, claim 3 requires: "retrieving the second information." One of ordinary skill in the art would understand that "searching" and "retrieving" apply to the searching of a data source and the retrieval of the second information from that data source. Exhibit 2003, Levy Decl. ¶ 48. The "association" between the first information and the second information is equivalent to the association of data in a database record. The specification itself supports an understanding that the word "associated" applies to shared information in a database record. Since "associated" in claim 3 is equivalent to the association of data in a database record, the same word "associated" must have the same meaning in other claims in the '854 patent including claim 1.

In the specification of the '854 patent, the term "associated" appears in Example 1 at col. 5, line 65 through col. 6, line 3.

> The user hits the button 42, for example, marked "OneButton" and the program according to the present invention retrieves the name 40 from the document, searches a database for the name 40, and inserts the

> retrieved address 44 associated with the name 40 into the document as shown in, for example, FIG. 4. (Emphasis Added).

As used in the specification, the term "associated" refers to the association between a name and an address for a given database entry.

The '854 patent discloses multiple embodiments of the invention including 7 enumerated examples, each of which is directed to the first and second information being contact related information where the contact information is associated in a database record. In all examples wherein finding and inserting the second information actually occurs in the '854 patent, there is a pre-existing relationship between first and second information. Only if second information is not found is there a lack of a pre-existing relationship, but likewise there is not second information available to complete the requirements of the claim by performing an action, such as inserting. Thus, there must be a pre-existing relationship for an action to be based upon the second information, such as the act of insertion.

The term "associated" must be construed in the context of the claim language, which is directed to a specific contemplated "possibility." The final limitation of claim 1 reads:

> responding to a user selection by inserting a second information into the document, the second information associated with the first information from a second application program.

In this contemplated possibility, in which second information is inserted into the document, the response to user selection is predicated on the second information being identified and "associated with the first information from a second application program" prior to the user selection. This "possibility" can be seen in Fig. 1 of the '854 patent when a search for first information occurs in a database (Step 12). In block 18 of Fig. 1, if the answer to the question "Find It?" is answered affirmatively, only then is the possibility of insertion in block 22 available. Thus, whenever insertion into the document is the result, an association between the first information and the second information exists. As a result, the association between the second information and the first information is necessarily pre-existing.

In the Decision instituting the IPR at page 10, the PTAB stated that "indeed, the '854 specification refers to related information that possibly may match the searched data or data that corresponds to part of a typed name" and at page 16 the PTAB states "we find that searching for related information as recited in the claims does not require a pre-existing relationship." The fact that "part of a typed name" may be first information does not negate the requirement that the relationship between the first and second information is "pre-existing." If a database stores the relationship between a name and an address, even if only a part of the name is used in the search, the relationship with the address is still one that was pre-existing in

the database. Therefore, there would be a pre-existing relationship between the "part of the typed name" and the second information (e.g. address) in the database.

The Patent Owner therefore submits that the claim language "the second information associated with the first information from the second application program" when read in context of the claims and the embodiments contemplated in the specification of the '854 patent requires that the association between the first and second information is a "pre-existing relationship", such as, the association between field entries for a database record in a database.

IV. OVERVIEW OF THE PRIOR ART

A. Overview of Domini

Domini is directed to a combined spell checking and grammar-checking module that operates within a word processing application. See, e.g., Ex. 1007 Abstract, col. 3, lines 1-12, col. 5, lines 1-8.

Fig. 3 reproduced below shows a dialog box for the spell checking and grammar-checking module after a user has selected the "spelling and grammar" command within the application program (e.g. word processing application). The preferred embodiment of the spell checking and grammar module of Domini operates within Microsoft Word. Exhibit 2003, Levy Decl. ¶ 49. According to the '854 patent, Microsoft Word is the application program.

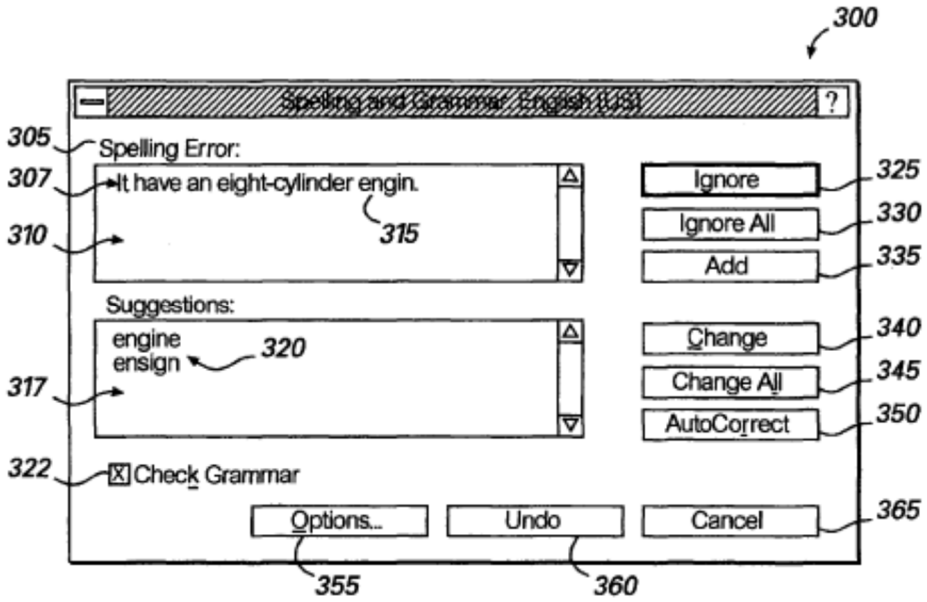


FIG.3

The preferred spell checking and grammar module employs both a Common Grammar Application Programming Interface (CGAPI) and a Common Speller Programming Interface (CSAPI). Ex. 1007, col. 8, lines 51-64 and col. 9, lines 6-19. With respect to the CGAPI and CSAPI, Domini states that the interfaces allow programmers to write applications that can utilize the grammar or spell checker "while being independent of the underlying grammar checker program module" or "spell checker program module". col. 8, lines 58-62 and col. 9, lines 11-16.The API as used in conjunction with the program module allows for the creation of a synchronous relationship between the application program (e.g. word processor) and the program module, such that the program module operates as a subsidiary of the application program. Exhibit 2003, Levy Decl. ¶¶54-57.

Domini also contemplates a spell checking and grammar module that operates in a distributed environment as well. Ex. 1007 Domini at col. 5, lines 13-26. In the distributed environment, program modules communicate synchronously with each other via network communications where the execution of some modules occurs remotely from the processor used directly by the user executing Word, but the module is still under the control of Word. There is no programming distinction between interfacing synchronously with a local module or synchronously with a remote module other than simply the destination of the module. Exhibit 2003 at ¶¶19 and 50-52 Instead of employing a local procedure call for the module to interface synchronously, the word processing application may use a remote procedure call for the module to interface synchronously. *Id.* at ¶ 51.

It is noteworthy that the paragraph in Domini discussing a distributed environment confirms that the Domini system works "in conjunction with other program modules." Exhibit 1007 at col. 5, lines 12-17. Nowhere does Domini contemplate the spell checker and grammar checker module operating as an independently executable program because it is always working in conjunction with a word processing program with which the user is editing the document.

In Domini, upon selection of the spelling and grammar command within the word processing application, a sentence is extracted from the word processing document and the spell checking program module is called within the process of the word processor, such that the compiled code of the module is run. *Id.* at col. 16, line 56-66. Exhibit 2003 at ¶ ¶20, 49. Each word is extracted from the sentence and the spell checking program module verifies whether the words appears in the one or more dictionaries of the spell checking program module. Exhibit 1007, Domini at col. 17, lines 19-42. If the word does not appear within any dictionary, an error is generated indicating that the word is potentially misspelled and the word processing application accesses a string buffer that contains one or more suggestions for the potentially misspelled word. The suggestions are displayed within a combined spelling and grammar checking dialog box. *Id.* at col. 18, line 4-20. A user may then select between the one or more suggestions to replace the word within the document. *Id.* at col. 18, lines 21-26.

In order to insert a correctly spelled word into the document, it is necessary for the module of Domini to make use of an interface with the word processing application to pass the suggestion to the main word processor module so that the main word processor module can insert the selected suggestion into the document. Exhibit 2003 at ¶20. Even if the module of Domini were running on a different processor from the main word processor module, in a distributed network

configuration, the module operates synchronously with the main word processor module and is not independently executable. *Id.* at ¶¶ 21, 50-52. The remote module receives a series of characters from the main word processor module through a client/server service such as a remote procedure call. *Id.* at ¶¶20, 46, 51. It is clear from the Domini specification that the main word processor module is suspended/blocked on the local processor while the remote module is executed and returns one or more parameters using a remote procedure call to the processor on which the main word processor module is being executed. The main word processing module process is then unblocked and the parameters are passed to the word processor. A graphical interface is presented to the user through the main word processor module. *Id.* at ¶¶ 51-52. The main word processor module performs the act of insertion. *Id.* at ¶¶ 49, 51-52.

B. Overview of Hachamovitch

Hachamovitch describes a word completion utility that is used in conjunction either with an individual application program or with a group of different application programs. Exhibit 1008 at col. 4, lines 10-28. The word completion utility is executed from within the application program. Exemplary embodiments of the word completion utility are incorporated into the MICROSOFT OFFICE 97 Suite. *Id.* at col. 9, lines 44-45. The word completion utility can predict word completions for data entries in an unstructured portion of a

data file, such as a word processing document or an e-mail within either a word processor application program or an e-mail application program. The suggested word completions may be based on a suggestion list that includes name-completion pairs. The partially typed word is compared to the name-completion pairs and if a match is found within the list, a suggestion list will be presented to a user. *Id.* at col. 4, line 58- col. 5, line 6. A word completion suggestion may be presented to the user in a pop-up user interface within a word processing application as shown in Fig. 2A. The suggestion is displayed to a user and the user may accept the word completion using an acceptance keystroke (e.g. tab or enter). Once accepted the word completion utility replaces the partial data entry with the completion entry in the data file using the pre-established interface between the word completion utility and the word processing application. *Id.* at col. 5, lines 7-10. Exhibit 2003, Levy Decl. ¶¶27-29.

As shown in Fig. 2A a user begins to type "Jun" and the word completion utility identifies this partial entry as being representative of the current date. Thus, the current date is presented in the pop-up user interface and the current date can be selected by the user and inserted into the data file by the word completion utility.

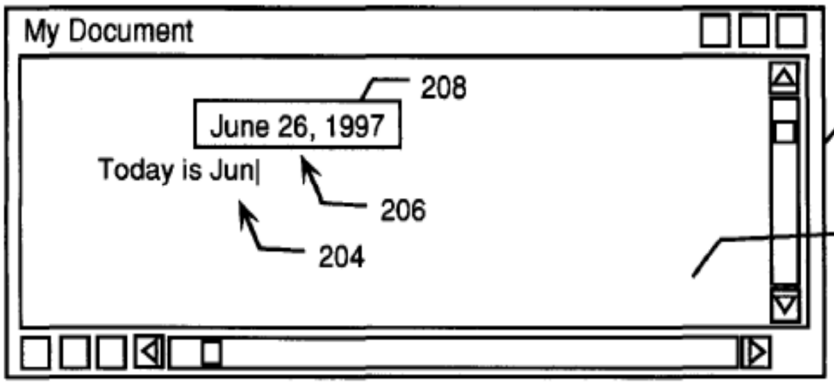


FIG. 2A

In addition to deploying the system on an individual application basis Hachamovitch also contemplates an embodiment in which the word completion utility resides within the operating system as a stand-alone utility on an application independent basis. Exhibit 1008 at col. 7, lines 62-67, col 4, lines 22-32. "Application independence is the ability of the same word completion system to work with several different application programs, such as a word processing program, an e-mail program, a spreadsheet program, and so forth." *Id.* at col. 4, lines 24-27. In order "to deploy the word completion system as an application-independent utility, an interface is defined within each application program through which the word completion utility may communication with each application program. This allows the word completion utility to monitor the entry of characters into the application program user interface, to determine the location within the user interface to display the word completion frame, and to determine when the user had invoked the word completion user interface." *Id.* at Col. 8, lines 6-17.

When the word completion utility of Hachamovitch operates on an application independent basis, the word completion utility remains under the control of the current "host" application program and operates synchronously with the application program. The utility provides no function unless an application program is executed and calls the utility through an interface, such as an application-programming interface. The word completion module remains under the control of and within the process of the word processor where the word processor and the module synchronously communicate. Exhibit 2003 ¶¶ 27-32. Accordingly, the word completion utility of Hachamovitch is not independently executable. *Id*.

Hachamovitch describes his system as a "utility," which according to Fig. 1 is distinct from an "application program." Thus, in Hachamovitch, a utility is computer code that adds functionality to one or more application programs, but never operates independently from its "host" application.

V. THE PRIOR ART DOES NOT ANTICIPATE OR RENDER ANY CLAIM OBVIOUS

A. Overview of reasons why the claims under review are patentable over the prior art of Domini and Hachamovitch

For each ground asserted in the present *inter partes* review [IPR2014-00207] the reference relied upon fails to teach at least one element of the claims to which the reference is applied.

Consequently, the grounds asserted fail to anticipate any claim under 35 U.S.C. §102(e) as alleged, and fail to rending any claim obvious under 35 U.S.C. §103(a) as alleged.

Each of the independent claims for review in the '854 patent require both a first application program and a second application program. The Petitioners look to Domini and Hachamovitch as anticipatory or making the claims obvious.

However, Domini fails to teach a second application program as required by the independent claims. Domini discloses a spell checker module that operates inside of a word processing application. Thus, Domini discloses only a single application program and therefore, fails to disclose a second application and fails to disclose second information from a second application as required by these claims. Domini discloses operation of its module in a distributed computing environment. Even though the software code of the application program and the module may be on different machines in a distributed computing environment, the code still operates in a synchronous manner, such that the spell-checking module is activated from within the application program and therefore, the spell-checking module is not an independently executable computer program. Thus, Domini fails to teach a second application program.

Similarly, the petitioners suggest that the word completion utility of Hachamovitch discloses both a first application program and a second application

program. However, Hachamovitch, like Domini, describes a subsidiary program that operates within a host application program, and therefore Hachamovitch does not teach the "second application program" limitation of these independent claims and their corresponding dependent claims. Hachamovitch mentions that one embodiment of the utility may be "application independent." As would be understood by one of ordinary skill in the art, application independence does not refer to a separate application program, but rather to the fact that the utility has an associated standard interface that can be linked with a plurality of application programs. Thus, the application independent embodiment of Hachamovitch functions materially the same as the utility running inside of the application program and therefore, does not teach or suggest a second application program.

In fact, even if Domini or Hachamovitch were created as an independent application programs, neither could perform the tasks they are created to perform, including insertion of a correctly spelled word or completed word into a document while the user is working on the document with another program such as a word processor.

Additionally, Domini fails to anticipate the claims because the independent claims require an association between first information and second information. The association must be similar to the association between data fields in a database record for which a search can be performed. Domini only creates an association

between first and second information when the user selects a correctly spelled word to insert into the document and thus, this association is not of a type that can be searched in a database (i.e. is not pre-existing). Finally, dependent claims 3-5, 9-11, 38-41 and 45-48 each require that "searching….for the second information associated with the first information" occurs in response to "a user selection", however, Petitioners' evidence suggests that user selection occurs after the performance of a search, and therefore, Petitioners do not establish that Domini anticipates these dependent claims.

B. Because Domini's spell checker is a module operating under the control of the first application program, Domini fails to disclose a "second application program", and therefore Domini fails to establish anticipation.

The '854 Patent discloses an embodiment including the retrieval of information (*e.g.*, a name or address) from one application program (e.g. Microsoft Outlook) while a user works simultaneously in another application program (e.g., Microsoft Word). See Exhibit 1001 col. 2, lines 14-23. The independent claims (1, 7, 36, 43) all require "a first application program" and "a second application program." As would be understood by one of ordinary skill in the art at the time of the invention, an "application program" is an "independently executable program" that is independent of and not under the control of another program. An

application program can be contrasted with a subsidiary program, such as a module, or a utility that adds functionality to a host application program and the module or utility operates synchronously with the host application program. Exhibit 2003, Levy Decl. at ¶18. Such subsidiary programs require the operation of a separate application program ("host") in order to be activated and to provide any functionality. *Id.* These modules do not operate independently and are not independently executable programs. *Id.* at ¶¶43-44.

When a subsidiary program resides on the same computer as the host application program, the module operates synchronously within the process of the host application program and the operating system does not block the application program in order for the module to function. *Id.* at ¶18. The module operates as a component of the word processing application program.

When a subsidiary program resides in a distributed computing environment on a server, the module is accessed by a remote procedure call in contrast to a local procedure call from the application program, and operation of the module still depends on the process of the application program, since the subsidiary program operates synchronously with the application program. *Id.* at ¶50-52. In this synchronous communication, the application program suspends/blocks operation while the subsidiary program is executed. When the application program receives results back from the subsidiary program, the application process is continued. In

this distributed environment, the subsidiary program functions exactly as if run on a local computer. *Id.* at ¶52.

The '854 Patent describes the interaction between two application programs wherein a user may operate within a first application program without the need for exiting and initiating a second application program to find information managed by the second application program that is related to information entered into a document within the first application program. Examples of application programs as specified within the specification of the '854 Patent encompass only independently executable computer programs: such as a word processing program, including Microsoft Word, NOTEPAD™, WORDPAD™, WORDPERFECT™, QUATROPRO™, AMIPRO™ a spread sheet program, such as Microsoft Excel, and a contact database program, such as Microsoft Outlook. ACCESS™, ORACLE™, DBASE™, RBASE™, CARDFILE™ Ex. 1001, col. 1, lines 39-42 and col. 9, line 64, to col. 10, line 10.

In contrast, the Domini reference discloses the retrieval of information (e.g., suggested spelling corrections) while a user works within a single program—i.e. a word processor. See, e.g., Ex. 1007 Abstract, col. 3, lines 1-12, col. 5, lines 1-8, col. 8, lines 51-57 , col. 9, lines 6-12 etc. Domini presents the suggested spelling corrections to the user and then allows for insertion of information into the document being drafted in the word processor.

Domini contemplates two different versions of the spell checker and grammar checker program module and mentions a less desirable non-integrated prior art version.

a. **Word processor and Domini's module on the same computer**

The first version operates on a personal computer, such that the word processor and the program module exist within the same personal computer. *Id.* at col. 5, lines 13-15. The spell checker program module of Domini is activated from within the word processor. *Id*. at col. 16, lines 12-16; Exhibit 2003, Levy Decl. at ¶¶18-20, 49. One of ordinary skill in the art reading Domini would understand Domini's module to be a subsidiary program that is activated by the controlling host application program and provides additional functionality to the application program, but is not itself an independently executable program. *Id.*

Thus, this first version of the spell checker program module of Domini is not an "application program" and therefore, this embodiment of Domini lacks the required "second application program" of the independent claims.

b. **Word processor and Domini's module in a distributed computing environment**

In the second embodiment, the spell checker and grammar checker program module of Domini is part of a distributed computing environment. Exhibit 1007 Domini at col. 5, lines 15-27. In such a configuration, the spell checker and grammar checker program module may be running on a different computer than the word processor, but is still under the control of the word processor. In such a configuration, the application program (word processor) synchronously may execute a remote procedure call to the server that contains the Domini module. Exhibit 2003 ¶51.The remote procedure call includes the required interface, such as, parameters passed from the application program to the module. The application program waits for the module to return with the proposed spellings. *Id.* at¶¶50-52 and the application program allows a user to insert a selected spelling into the document. *Id.* at ¶20. Operation of the module in this context still depends on the word processor and the spelling and grammar-checking module operating synchronously.

Therefore, the spelling and grammar-checking module of Domini in a distributed system does not operate independently and only responds to actions through the word processor application program. *Id.* at ¶50. Thus, insertion can only be achieved in Domini by the word processor itself, which is passed the selected correctly spelled word. *Id.* at ¶20. As with the first embodiment, this second embodiment of the spell checker and grammar checker module is a

subsidiary program to the word processor and is not itself an application program.[2] *Id.* at ¶¶18, 21, 44, and 50. Also in this environment, the Domini spell checking program module is useless without its host application program and cannot function independently. Therefore, this second embodiment in Domini of the spell checker and grammar checker module lacks the required second application program of the independent claims.

c. **Domini's "stand alone" spell checker cannot replace a misspelled word with a suggested correction**

The background section of Domini discloses a "stand alone" spell checker Exhibit 1007 col. 1, lines 56-66. A person of ordinary skill in the art would understand that a "standalone spell checker" was an independently executable computer program that does not interact with a word processor program in order to perform its spell checking function. Such a spell checker, being independently executable, would generate and output its own user display and receive inputs directly from the user. Such a separate user interface is cited by Domini as a

[2] In the Decision to institute the current IPR, the PTAB at page 15 states that "Domini also discloses that the system could run in a distributed computing environment with program modules located and executed locally and remotely." Paper 9. Thus, the PTAB misinterpreted Domini's distributed embodiment as an "independently executable computer program."

disadvantage of performing spell checking using an independently executable program (see Exhibit 1007 col. 1, line 56 to col. 2, line 26). Exhibit 2003, Levy Decl. at ¶24.

Domini describes these stand-alone spell checkers as not being integrated with the word processor. As such, Domini suggests to one of ordinary skill that a stand-alone spell checker would not be capable of inserting text into a word processor. Exhibit 2003 ¶¶24-25,35.

We have shown that the Domini module is not independently executable, and therefore is not a "second application program" as required by the independent claims. Moreover, nothing in Domini suggests that it could operate as an independently executable program. Because Domini fails to teach a second application program, Domini similarly fails to disclose or suggest using both a "first application program" and a "second application program" as required in independent claims 1, 7, 36, and 43. As a result, Ground 1 fails to establish anticipation of the independent claims listed above.

Additionally, since Domini fails to disclose "a second application", Domini also fails to disclose "second information" "associated with the first information from a second application program" as required by independent claim 1, 7, 36, and 43. Thus, Domini fails to teach the "associated information" limitation of the independent claims in addition to failing to teach a second application program.

For at least these reasons, Ground 1 fails to establish anticipation by Domini of the above listed independent claims or for dependent claims 2-6, 8-12, 37-42, and 45-49.

C. Because the independent claims require a pre-existing association between first information and second information, Domini fails to anticipate the claims.

Domini's spell checker and grammar checker module operates by looking at every term in the text (e.g., textually delineated grouping of characters) and comparing it to a dictionary list, and if the term is not found in the list, then the term is highlighted, and a listing of suggested words is presented to the user from which the user makes a selection. Exhibit 1007 col. 4, lines 13-16 The independent claims 1, 7, 36, and 43 each require an association between second information and first information and more specifically, claim 1 requires "the second information associated with the first information from a second application program." As explained above in the claim construction section III B., the association is of a type that the second information can be found as a result of a [successful] search for the first information. The association between the first and second information must be pre-existing, such as, the relationship between field entries of a database record. Independent claim 1 and dependent claim 3 both use the term "associated" and the Federal Circuit has found that terms within a patent should be construed

consistently. *Phillips,* 415 F.3d at 1314. In claim 3, the term "associated" appears in the context of the limitations:

searching, using the second application program, for the second information associated with the first information; and

retrieving the second information.

In this context, the association between the first and second information is such that the association can be searched and the second information can be retrieved as a result of the search. This type of association is one that includes a "pre-existing" relationship. If the association is not pre-existing, a search for second information based upon first information cannot be performed. Exhibit 2003, Levy Decl. at ¶48.

In Domini, there is a search for each delineated set of characters (i.e. a word in the document), but the list of suggested alternative spelled words comes only if the search is unsuccessful. Exhibit 1007 Fig. 7 no. 725-735. The suggestions that are presented to the user are retrieved from a string buffer ("SRB") and do not share a pre-existing relationship with the misspelled word, such as, the association between field entries for a database record in a database. An association is only made between a suggestion and the misspelled word when the user recognizes the correctly spelled word and selects the suggested word as a replacement for the misspelled word. Thus, the "association" in Domini between the misspelled word

(e.g. first information) and the selected correctly spelled word (e.g. second information) is not an "association" as contemplated by claim 1 in the '854 patent, since the association is not of the type that can be searched for in a database.

Thus, Domini fails to teach the limitation of:

responding to a user selection by inserting a second information into the document, the second information associated with the first information from a second application program.

Domini's embodiments of its word completion system either fail to have the type of association required by the claim limitation or fail to operate in response to user selection to cause insertion of second information "associated" with first information.

For at least these reasons, Petitioners fail to establish anticipation by Domini of the above listed independent claims or for dependent claims 2-6, 8-12, 37-42, and 45-49.

D. Because the Petitioners have failed to establish that Domini teaches "searching….for the second information associated with the first information" responsive to "a user selection"", Domini does not anticipate dependent claims 3-5, 9-11, 38-41 and 45-48.

Dependent claims 3, 9, 38, and 45 each state that the act of insertion further comprises searching for second information associated with first information. Each of the claims from which these claims depend requires that second information be inserted into a document in response to user selection. Thus, the claim language requires that searching occur as the result of user selection. Domini does not teach searching for second information associated with first information in response to user selection. More specifically claim 1 requires:

> responding to a user selection by inserting a second information into the document, the second information associated with the first information from a second application program.

In addition, Claim 3 requires that the step of inserting further comprises:

> searching, using the second application program, for the second information associated with the first information; and
> retrieving the second information.

The Petitioners suggest that the user selection is the selection of one of the suggested corrections that are retrieved from the SRB (spell return buffer). See Petition at page 30 element [1d]. However, the search that is suggested by the Petitioners is the search and retrieval of the suggested corrections from the SRB buffer. See Petition at pgs. 31-32 for element [3c] As stated above, the user

selection must precede the search, since the search occurs in response to user selection; however the user selection as articulated by the Petitioner occurs after the search has already occurred. Thus, the suggested search of the Petitioner does not meet the requirement of dependent claims 3, 9, 38, and 45.

Since the Petitioners have failed to establish that Domini teaches a search in response to user selection, the Petitioners have failed to show that Domini anticipates claims 3, 9, 38, and 45.

For at least these reasons, Ground 1 fails to establish anticipation by Domini of at least dependent claims 3, 9, 38, and 45 and the claims that depend from these claims including claim 4-5, 10-11, 39-41 and 46-48.

E. Because Hachamovitch describes a utility operating within a first application, Hachamovitch fails to disclose "a second application program", and therefore, Hachamovitch does not anticipate the independent claims 1, 7, 36, and 43 and Hachamovitch does not render obvious any of dependent claims 3-5, 9-11, 38-41, and 45-48, all of which depend on the independent claims.

As previously articulated with respect to the Domini reference, the independent claims 1, 7, 36, and 43 in the '854 Patent require "a first application program" and a separate "second application program" where the two application programs are independently executable.

Hachamovitch describes a word completion utility for use with a word processor where the word completion utility adds functionality to an application

program Ex. 1008 at col. 33-34. A utility is known to be a subsidiary program and cannot be used independently of a host application program. Ex. 1008 Fig.1, Exhibit 2003 ¶¶27-29.

Hachamovitch states that the word completion utility can be deployed within an individual application program (*Id.* at col. 7, lines 62-64) or the utility can be an application-independent utility (*Id.* At col. 7, lines 65-67).

> The word completion system may be deployed within an individual application program, particularly a word processing application program or an e-mail application program. Alternatively, the word completion system may be deployed within an operating system or as a stand-alone utility that may operate on an application-independent basis. col. 7, lines 62-67

When referring to the word completion utility as a stand-alone application-independent utility, Hachamovitch does not contemplate the word completion utility as a separately executable application program, but rather a utility that may be accessed by a plurality of application programs. Exhibit 2003, Levy Decl. at ¶30 In this context, the term stand-alone merely suggests a subsidiary program that has a standard interface. *Id.* Thus, the word completion utility stands alone in a common memory location where any requesting application program can make use of the utility using the standard interface. *Id.* "To deploy the word completion system as an application-independent utility, an interface is defined within each

application program through which the word completion utility may communicate with each application program." Exhibit 1008, col. 8, lines 6-9.

> This allows the word completion utility to monitor the entry of characters into the application program user interface, to determine the location within the user interface to display the word completion frame, and to determine when the user had invoked the word completion user interface.
>
> Col. 8, lines 10-14.

By having a standard interface between the utility and the application program, the utility operates synchronously with the application program and the application program knows to pass the keystrokes to the utility. Exhibit 2003, Levy Decl. at ¶¶30-31. The utility can receive information from the application program about where any graphical overlays should be placed on the window that displays the application program. The utility may generate the overlay itself, or the overlay may be generated by the application program. In either embodiment, the utility operates synchronously with the application program and is only activated from within the host application program.

Thus, an application-independent utility is simply a subsidiary program that can be synchronously accessed by a plurality of application programs and the utility does not operate independent of any application program. Exhibit 2003

¶¶31, 43. As a result, Hachamovitch, like Domini, fails to teach the required second application program.

One of ordinary skill in the art would not be led to implement the Hachamovitch word completion system as an independently executable program, since that would forego the ease of inserting word completion results into the document characteristic of synchronous operation. Exhibit 2003 at ¶¶34-35. Hachamovitch discloses the steps one would take to implement synchronous operation. There is no disclosure or suggestion of how one of ordinary skill in the art would implement an independently executable word completion program and still achieve insertion of results into the document being processed by another independently executable program. Exhibit 2003 ¶36. As a result, Hachamovitch fails to teach or suggest a second application program as required by the independent claims.

For at least this reason, Petitioners have failed to establish anticipation of independent claims 1, 7, 36, and 43 and corresponding dependent claims 2-6, 8-12, 37-42, and 44-49 by Hachamovitch, and Petitioners have failed to establish a *prima facie* case for obviousness in view of Hachamovitch of dependent claims 3-5, 9-11, 38-41, and 45-48.

VI. CONCLUSION

For the foregoing reasons, Petitioners have failed to show by a preponderance of the evidence that any of the claims being reviewed (claims 1-12, and 36-49), of U.S. Patent No. 7,496,854 are unpatentable over the cited prior art. Accordingly, the Board should find that claims 1-12 and 36-49 of the '854 Patent are not rendered unpatentable by Domini or Hachamovitch.

Dated: August 26, 2014

Respectfully submitted,

/Robert M. Asher, #30,445/

Robert M. Asher
Registration No. 30,445
Bruce D. Sunstein
Registration No. 27,234
John J. Stickevers
Registration No. 39,387
Sunstein Kann Murphy & Timbers LLP
125 Summer Street
Boston, MA 02110
Tel: (617) 443-9292
Fax: (617) 443-0004

CERTIFICATE OF SERVICE

It is certified that on August 26, 2014, copies of the PATENT OWNER ARENDI S.A.R.L.'S RESPONSE UNDER 37 C.F.R. § 42.120 and Exhibits 2002 and 2003 have been served on Petitioners as provided in 37 C.F.R. § 42.6(e) via electronic mail transmission addressed to the persons at the following addresses:

LEAD COUNSEL FOR PETITIONER APPLE DAVID L. FEHRMAN dfehrman@mofo.com Registration No. 28,600 MORRISON & FOERSTER LLP 707 Wilshire Blvd., Suite 6000 Los Angeles, CA 90017-3543 Tel: (213) 892-5200 Fax: (213) 892-5454	BACK-UP COUNSEL FOR PETITIONER APPLE MEHRAN ARJOMAND marjomand@mofo.com Registration No. 48,231 MORRISON & FOERSTER LLP 707 Wilshire Blvd., Suite 6000 Los Angeles, CA 90017-3543 Tel: (213) 892-5200 Fax: (213) 892-5454
LEAD COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC. MATTHEW A. SMITH smith@turnerboyd.com Registration No. 49,003 TURNER BOYD LLP 2570 W. El Camino Real, Suite 380 Mountain View, CA 94040 Tel: (650) 265-6109 Fax: (650) 521-5931	BACK-UP COUNSEL FOR PETITIONERS MOTOROLA MOBILITY, LLC and GOOGLE, INC. ZHUANJIA GU gu@turnerboyd.com Registration No. 51,758 JULIE S. TURNER turner@turnerboyd.com TURNER BOYD LLP 2570 W. El Camino Real, Suite 380 Mountain View, CA 94040 Tel: (650) 265-6109 Fax: (650) 521-5931 and kent@turnerboyd.com docketing@turnerboyd.com

Date: August 26, 2014

/Robert M. Asher, #30,445/
Robert M. Asher
Registration No. 30,445
Sunstein Kann Murphy & Timbers LLP
125 Summer Street
Boston, MA 02110
Tel: (617) 443-9292
Fax: (617) 443-0004

Exhibit 6T

Trials@uspto.gov
Tel: 571-272-7822

Paper 32
Entered: June 9, 2015

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC., GOOGLE INC., and MOTOROLA MOBILITY LLC,
Petitioner,

v.

ARRENDI S.A.R.L.,
Patent Owner.

Case IPR2014-00207
Patent 7,496,854 B2

Before HOWARD B. BLANKENSHIP, SALLY C. MEDLEY, and TREVOR M. JEFFERSON, *Administrative Patent Judges.*

JEFFERSON, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
35 U.S.C. § 318(a) and 37 C.F.R. § 42.73

I. INTRODUCTION

A. Background

Apple Inc., Google Inc., and Motorola Mobility LLC (collectively "Petitioner") filed a Petition (Paper 3, "Pet.") to institute an *inter partes* review of claims 1–18, 36–56, 86–95, 97, 98, 100, and 101 of U.S. Patent 7,496,854 B2 (Ex. 1001, "the '854 patent"). Pet 1; *see* 35 U.S.C. § 311. Arendi S.A.R.L. ("Patent Owner") filed a Preliminary Response (Paper 7, "Prelim. Resp."). Pursuant to 35 U.S.C. § 314, in our Decision to Institute (Paper 9, "Dec."), we instituted this trial as to claims 1–12 and 36–49. Dec. 23–24.

After the Decision to Institute, Patent Owner filed a Patent Owner Response (Paper 16, "PO Resp.") and Petitioner filed a Reply to the Patent Owner Response (Paper 20, "Reply"). An oral hearing (Paper 31, "Tr.") was held on February 4, 2015.

B. Related Matters

Patent Owner has sued Petitioner for infringement of the '854 patent in *Arendi S.A.R.L. v. Apple Inc.*, No. 1:12-cv-01596-LPS (D. Del.); *Arendi S.A.R.L. v. Google Inc.*, No. 1:13-cv-00919 (D. Del.); and *Arendi S.A.R.L. v. Motorola Mobility LLC*, Case No. 1:12-cv-01601-LPS (D. Del.). Pet. 1; Paper 6, 2–3. The '854 patent is also the subject of a petition in IPR2014-00206, also filed by Petitioner. *Id*. at 3–4. We instituted *inter partes* review as to claims 19, 20, 22–26, 28–30, 57, 58, 60–74, 76–78, 85, and 96 of the '854 patent in *Apple Inc. v. Arendi S.A.R.L.*, Case IPR2014-00206, slip op. at 22 (PTAB June 11, 2014) (Paper 9).

C. References Relied Upon

Petitioner relies upon the following prior art references:

Reference	Exhibit
U.S. Patent No. 6,085,206 ("Domini")	Ex. 1007
U.S. Patent No. 6,377,965 ("Hachamovitch")	Ex. 1008

D. The Alleged Grounds of Unpatentability

We instituted this trial based on the ground of unpatentability set forth in the table below. Dec. 14–16, 17–22, 23.

Reference	Basis	Claims Challenged
Domini	35 U.S.C. § 102(e)	1–12, 36–38, 40–45, and 49
Hachamovitch	35 U.S.C. § 102(e)	1–12 and 36–49
Hachamovitch	35 U.S.C. § 103(a)	3–5, 9–11, 38–41, and 45–48

E. The '854 Patent

The '854 patent, titled "Method, System and Computer Readable Medium for Addressing Handling From a Computer Program," relates to computer implemented processes for providing a computer program, such as a word processing program or spreadsheet program, that is coupled to an information management source, such as a database program or contact management program. Ex. 1001, 1:19–50.

Figures 3 and 4 of the '854 patent are reproduced below.

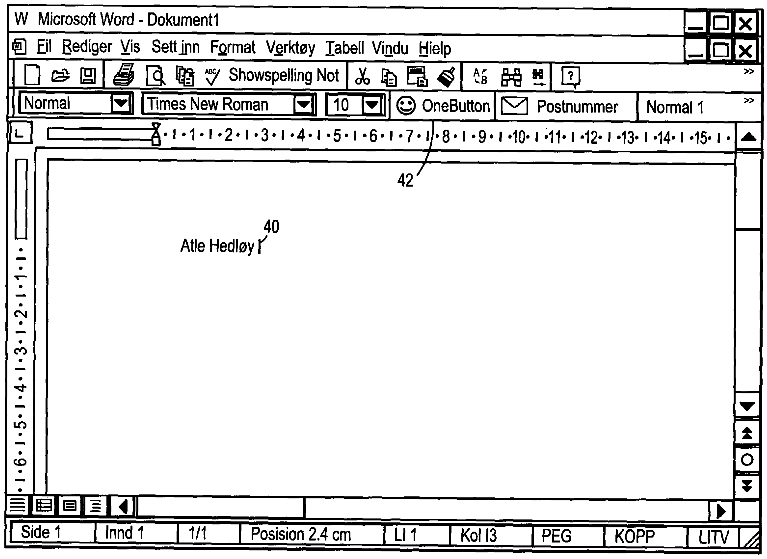


FIG. 3

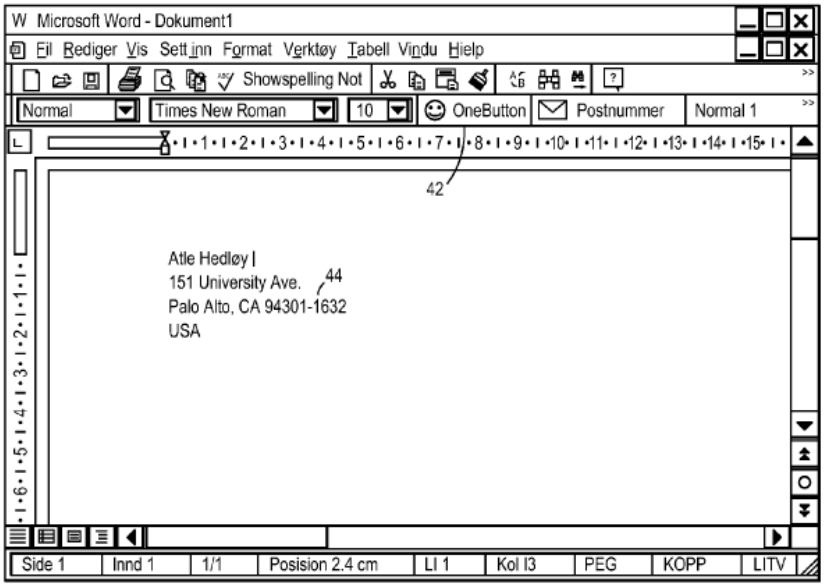


FIG. 4

Figure 3 illustrates the inputting of a name to be searched into a document. Figure 4 illustrates a retrieved address that is inserted into a document. Ex. 1001, 2:51–57. The user types a name into the document. When the user clicks on OneButton 42, the claimed process is launched, retrieving name 40 from the document,

searching a database for name 40, and inserting the retrieved address associated with the name 40 into the document as shown in Figure 4. *Id*. at 5:60–6:5.

Figure 2 of the '845 patent, illustrating a flow chart of a method for address handling within a computer program, is reproduced below.

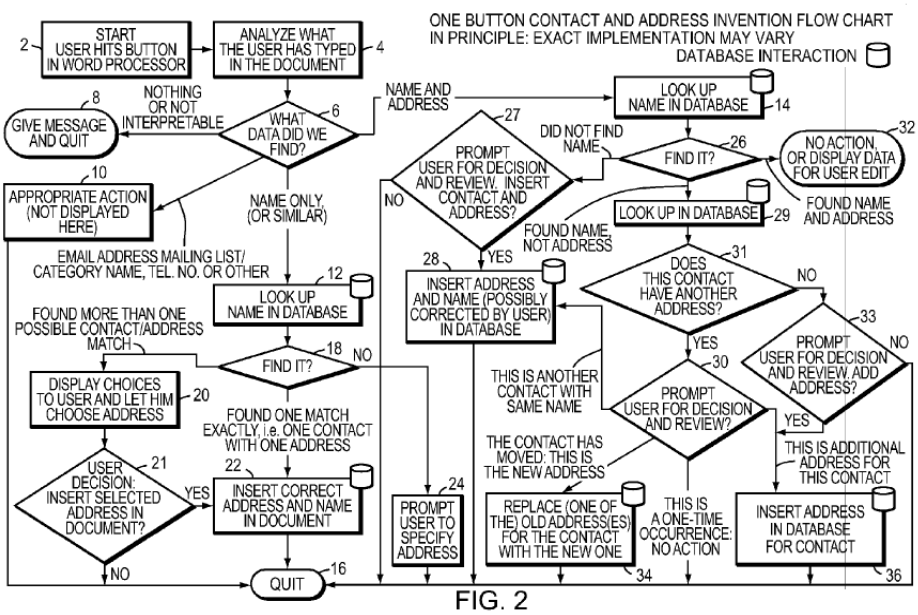


FIG. 2

Figure 2 depicts a flow chart of the address handling process initiated by the user clicking on OneButton 42 of Figure 4. At step 4, text typed by the user in a document is analyzed for contact information. At step 6, if the identified contact information includes a name, a search occurs in the database at step 12. When the database finds a name with more than one possible matching address, the user is prompted for a decision, and that selected information is added to the document at step 22. *Id*. at 5:10–22, 6:4–5.

Independent claim 1, reproduced below, is illustrative of the claimed subject matter:

> 1. A method for information handling within a document created using a first application program comprising the steps of:

> entering a first information in the first application program;
>
> marking without user intervention the first information to alert the user that the first information can be utilized in a second application program; and
>
> responding to a user selection by inserting a second information into the document, the second information associated with the first information from a second application program.

II. ANALYSIS

A. Claim Construction

The Board interprets claims of an unexpired patent using the broadest reasonable construction in light of the specification of the patent in which they appear. *See* 37 C.F.R. § 42.100(b); *In re Cuozzo Speed Techs., LLC*, 778 F.3d 1271, 1279–81 (Fed. Cir. 2015). Claim terms generally are given their ordinary and customary meaning, as would be understood by one of ordinary skill in the art in the context of the entire disclosure. *See In re Translogic Tech., Inc.*, 504 F.3d 1249, 1257 (Fed. Cir. 2007).

1. "marking . . . the first information to alert the user"

The claim phrase "marking without user intervention the first information to alert the user" appears in independent claims 1 and 7. In the Decision to Institute, we preliminarily construed "marking without user intervention the first information to alert the user" to mean that the first information is detected and has some form of marking applied to it without user intervention claim. Dec. 7–8. We also determined that "marking" included the acts of highlighting, designating, or displaying the information in a separate screen or window to draw a user's attention. *Id.*

The parties do not dispute this preliminary construction. Based on the record before us, we determine that "marking" includes highlighting, designating, or displaying the information in a separate screen or window to draw a user's attention.

2. "associated"

"Associated" appears in the claim phrase "second information associated with the first information from a second application program" recited in each of the challenged independent claims 1, 7, 36, and 43. In the Decision to Institute, we determined preliminarily that "associated" is construed as "connected or related." Dec. 9–10.

Patent Owner contends that because "associated" in dependent claim 3 describes searching for the second information "associated" with the first information, this indicates that the "association" is equivalent to the association of a data in a database record. PO Resp. 16–17. Patent Owner argues that the context of the claims and embodiments of the '854 patent "requires that the association between the first and second information is a 'pre-existing relationship,' such as, the association between field entries for a database record in a database." PO Resp. 20. Patent Owner further contends that the '854 patent embodiments refer to finding and inserting the second information, such that "there must be a pre-existing relationship for an action to be based upon the second information, such as the act of insertion." PO Resp. 17–18 (citing Ex. 1001, 5:65–6:3).

We are not persuaded by Patent Owner's argument that a "pre-existing" relationship is required for "second information associated with the first information from a second application program." The '854 specification refers to related information that may match the searched data or data that corresponds to

part of a typed name. Ex. 1001, 3:63–67, 4:43–58. Indeed, the '854 patent written description states that there may be "more than one *possible* contact/address match" to the first information and that "the program displays menu choices to the user to let him choose an appropriate answer" to insert. Ex. 1001, 4:46–49 (emphasis added). Thus, Patent Owner has not demonstrated that a pre-existing relationship is described in the '854 specification.

Patent Owner's arguments limiting the term "associated" to the examples in the '854 specification referring to databases also is not commensurate in scope with the breadth of the claims or the broadest reasonable interpretation. We must be careful not to read a particular embodiment appearing in the written description into the claim if the claim language is broader than the embodiment. *See In re Van Geuns*, 988 F.2d 1181, 1184 (Fed. Cir. 1993) (stating that "limitations are not to be read into the claims from the specification"). In the present case, Patent Owner has not shown persuasively that a person of ordinary skill would have understood the "second information associated with the first information from a second application program" as being limited to a "pre-existing relationship" between two pieces of information based on the claims, embodiments, and examples in the '854 specification. To the contrary, the '854 specification describes that a program operation based on a name or initials (the first information) could return more than one possible matching second information for insertion. Ex. 1001, 4:43–58.

In sum, under a broadest reasonable interpretation, we do not find that "associated" as recited in "second information associated with the first information from a second application program" is limited to a pre-existing relationship. We determine that "associated" is construed as "connected or related" and that that "second information associated with the first information from a second

application program" includes second information that is related to or connected with the first information from a second application program.

3. "application program"

In the Decision to Institute, we determined that "application program" encompasses an independent executable program. Dec. 10. In so determining, we rejected Patent Owner's narrow proposed construction of application program as "an independently executable computer program designed to assist in the performance of a specific task, such as word processing or spreadsheet processing or contact management or e-mail or calendaring." Prelim. Resp. 9; Dec. 10.

Patent Owner contends that it agrees with our interpretation (PO Resp. 11) of an "application program" as an independently executable program, but attempts to interpret "independently executable program"[1] in a manner that excludes programs with certain attributes. *See* PO Resp. 10–16. Based on the background section of the '854 patent that refers to retrieval of information from sources external to a word processor, such as a database or contact management program, Patent Owner asserts that the claimed invention is limited to obtaining information from an information management program that can be used separately and independently from the word processor. PO Resp. 12 (citing Ex. 1001, 1:34–37, 1:45–46).

[1] Patent Owner suggests that the Decision to Institute's use of the term "independent" rather than "independently" in construing "'application program' to encompass an independent executable program" (Dec. 11) was a typographical error. PO Resp. 10–11. Patent Owner does not explain how "independent" differs from "independently" and defines the term "independent" as part of its analysis. *Id*. at 10–11. For purposes of this Decision, we address Patent Owner's contentions as if they apply to both "independent" and "independently."

Patent Owner's interpretation of "application program" is based on the commonly shared features of the example programs from the specification. PO Resp. 14 ("Given that the specification identifies word processors, spreadsheet programs, information management programs and database programs as examples of application programs, the definition of an 'application program' can be construed from the commonly shared features."). Patent Owner relies on the Declaration of Dr. John Levy (Ex. 2003), paragraphs 42–43, to support its interpretation that subsidiary programs, which extend the functionality of the controlling application, are not "independently executable computer programs" as recited in the claims. PO Resp. 12.

We disagree with Patent Owner's narrow interpretation. The term "application program" does not appear in the specification of the '854 patent. However, we are not persuaded that the term is limited by the commonly shared features of the examples in the '854 patent specification. *See Van Geuns*, 988 F.2d at 1184 ("[L]imitations are not to be read into the claims from the specification."). Patent Owner has not provided sufficient evidence to limit "application program" to programs that are not under the control of another program or run synchronously under the control of a separate application program (PO Resp. 13–14). We do not find Dr. Levy's testimony persuasive that the broadest reasonable interpretation of "application program" by one of ordinary skill in the art at the time of the invention is defined by "commonly shared features" of examples of computer programs in the patent specification. *See* PO Resp. 11–16 (citing Ex. 2003 ¶¶ 42–44). Construing "application program" as Patent Owner suggests improperly limits the claim term to the embodiments and examples in the '854 patent specification and imports negative limitations unsupported by the intrinsic evidence.

Patent Owner has not shown that the broadest reasonable construction of "application program" excludes subsidiary programs. *See* PO Resp. 11–16; Reply 6–10. On the full record, we determine that "application program" is construed as an independent executable program.

B. Unpatentability Based on Domini (Ex. 1007)

Petitioner contends that Domini anticipates claims 1–12, 36–38, 40–45, and 49 under 35 U.S.C. § 102(e). Pet. 31–39.

1. Domini (Ex. 1007)

Domini discloses identifying and correcting spelling and grammar errors in a document created by a word processing program. Ex. 1007, Abstract, 4:65–5:11. Figure 1, below, shows an embodiment of the invention. *Id*. at 4:39–41.

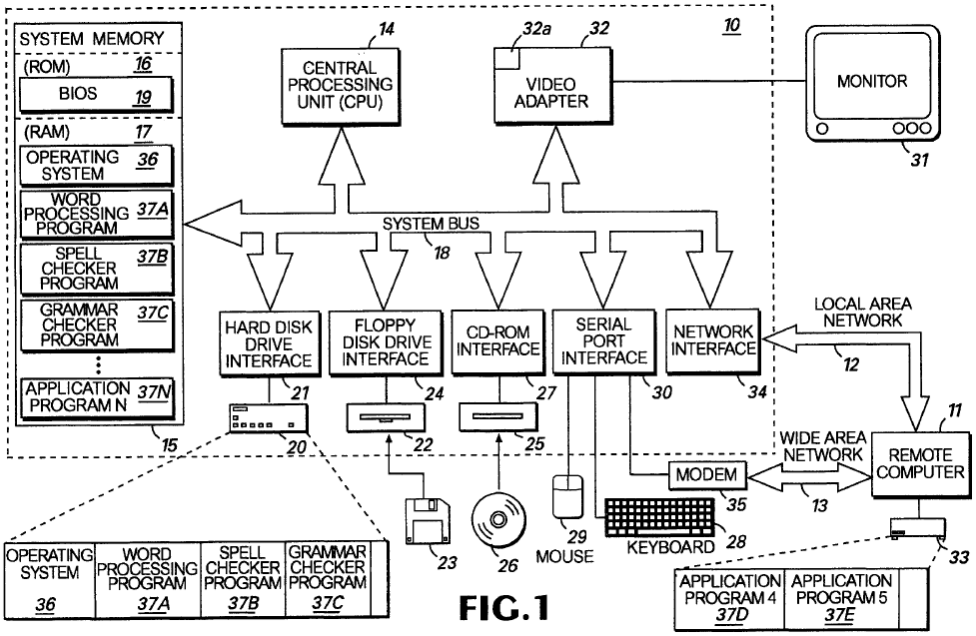


FIG.1

Figure 1 depicts personal computer 10 connected by networks 12 and 13 to remote computer 11. *Id*. at 7:13–16. Domini discloses that "[t]hose skilled in the art will understand that program modules such as an operating system 36, application

programs 37, and data are provided to the personal computer 10." *Id*. at 7:41–43. Thus, personal computer 10 and remote computer 11 contain program modules, such as operating system 36, application programs 37. *Id*. at 6:33–42, 7:41–44. Domini states further that:

> [t]he *application programs* 37 may include a number of different programs such as a word processing program 37a, a *spell checker program 37b*, and a grammar checker program 37c. In the preferred personal computer 10, the local hard disk drive 20 is used to store data and programs, including the operating system and programs.

Id. at 7:46–52 (emphasis added).

In the spelling and grammar programs disclosed in Domini, the user selects the "Spelling and Grammar" command to initialize the spell check program. *Id*. at 16:13–16. Without user intervention, the spell check program identifies misspelled words and presents them in red, bold typeface. *Id*. at 17:27–33, 4:12–16. The spell check program also displays a list of suggested corrections that may be selected and entered into the document by the user. *Id*. at 1:42–44, 12:1–5, 12:61–64.

2. Anticipation based on Domini (Ex. 1007)

a. "application program"

Patent Owner argues that the spell checker program in Domini is a "module" operating under the control of the first application program and fails to disclose the "second application program" as recited in claims 1–12, 36–38, 40–45, and 49. PO Resp. 30–35.

Patent Owner's arguments and analysis are based on its proposed claim interpretation that excludes subsidiary programs from "application program" as recited in the challenged claims. PO Resp. 30–31 (contrasting subsidiary programs with application programs and stating that it "would be understood by one of

ordinary skill in the art at the time of the invention, an 'application program' is an 'independently executable program' that is independent of and not under the control of another program." (citing Ex. 2003 ¶¶ 42–48)). As discussed above, we do not construe "application program" to exclude subsidiary programs based on characteristics of the example programs in the '854 patent specification. *See* Section II.A.3, *supra*. Accordingly, we are not persuaded by Patent Owner's arguments and testimony of Dr. Levy that the program modules in Domini that are explicitly identified as "application programs" do not meet the claim limitation for "application program." PO Resp. 30–35. Dr. Levy's narrow interpretation of application program reads limitations into the claim (PO Resp. 30), that are not supported by the '854 specification. *See* Reply 8–9 (citing Ex. 1001, 7:25–8:7).

Patent Owner's argument that one of ordinary skill in the art would have understood spell checker program 37b to be a program module (PO Resp. 30–31) and not an application program—as it is expressly described—is not supported by a plain reading of the Domini disclosure. Patent Owner's arguments and evidence fail to address the term "application program" as it is used in Domini and show that it differs from the "application program" as recited in the challenged claims. Indeed, Dr. Levy has not provided testimony that the term "application program" as used in Domini (Ex. 1007, 7:41–52, Fig. 1) differs from or is inconsistent with "application program" as recited in the challenged claims. *See* Tr. 35:13–20 (stating that Patent Owner's expert found Domini consistent with Patent Owner's construction).

With respect to application programs in Domini, Patent Owner concedes that Domini discloses stand-alone spell checkers (PO Resp. 35–36 (citing Ex. 1007, 1:56–2:26; Ex. 2003 ¶ 24)), but argues that "a stand-alone spell checker would not be capable of inserting text into a word processor" (PO Resp. 36 (citing Ex. 2003

¶¶ 24–25, 35)). We agree with Petitioner (Reply 11), however, that Domini discloses incorporating changes into a document by replacing words in the word processing document. Ex. 1007, 12:59–13:31, 14:42–67.

We disagree with Patent Owner's narrow interpretation of the term "application program" that excludes spell checker program 37b explicitly disclosed in Domini (Ex. 1007, 7:46–52). PO Resp. 25–29. We find that spell checker program 37b and other application programs in Domini disclose the "application program" recited in the challenged claims. Based on the complete record, we find that Domini discloses an "application program." Based on the complete record and in light of Patent Owner's and Petitioner's arguments and evidence, we find, by a preponderance of the evidence, that Domini discloses an "application program" as recited in the challenged claims.

b. "the second information associated with the first information from the second application program"

Patent Owner's argument that Domini fails to disclose "the second information associated with the first information from the second application program" is based on its claim construction that "associated' requires a pre-existing relationship between the first and second information. PO Resp. 37–39. Because we determined previously that "associated" is construed as "connected or related" (Section II.A.2, *supra*), we are not persuaded by Patent Owner's argument. Patent Owner's argument that Domini does not disclose a pre-existing relationship, "such as the relationship between field entries of a database record" (PO Resp. 37) is premised on an overly narrow interpretation of the claim term that is not commensurate in scope with the challenged claims.

c. searching in response to a user selection

Patent Owner contends that dependent claims 3, 9, 38, and 45, require that searching for second information is initiated or prompted by a user selection. PO Resp. 39–40. Claim 1 of the '854 patent recites "responding to a user selection by inserting a second information into the document." Claim 3, which depends from claim 1, requires that "the step of inserting further comprises: searching using the second application, for the second information associated with the first information and retrieving the second information." Patent Owner argues that Petitioner has failed to show that Domini discloses searching as a result of a user selection, because the suggested substitute terms for insertion are already obtained when the user makes the selection. PO Resp. 40–41 (citing Pet. 31–32).

Petitioner responds that Patent Owner's interpretation of the claim limitation is improper as it excludes disclosed embodiments of the '854 invention. Reply 13 (citing Ex. 1001, Fig. 1, 4:46–50). Petitioner argues that claims 3, 9, 38, and 45 "simply state that *insertion* must be done in response to user selection" and "[t]here is no required ordering between searching and user selection to insert." Reply 13–14.

Based on the record, we are not persuaded the Petitioner has shown by a preponderance of the evidence that the searching steps of claims 3, 9, 38, and 45, occur in response to a user selection. Dependent claims 3, 9, 38, and 45, and the claims that depend therefrom, provide additional limitations on the insertion step that takes place in response to a user's selection. The evidence cited by Petitioner refers only to insertion of the already retrieved suggestions. *See* Pet. 31–36 (claim chart showing claims 3, 9 and 38 and applying same analysis to claim 45). Accordingly, we find that Petitioner has not shown by a preponderance of the

evidence that dependent claims 3–5, 9–11, 38, 40, 41, and 45 are anticipated by Domini.

d. Conclusion

Based on the complete record and the evidence and argument presented by Petitioner and Patent Owner, we find, by a preponderance of the evidence, that Domini anticipates claims 1, 2, 6–8, 12, 36, 37, 42–44, and 49. Petitioner has not shown by a preponderance of the evidence that Domini anticipates claims 3–5, 9–11, 38, 40, 41, and 45.

C. Unpatentability Based on Hachamovitch (Ex. 1008)

Petitioner contends that Hachamovitch anticipates claims 1–12 and 36–49 under 35 U.S.C. § 102(e). Pet. 40–45. In addition, Petitioner contends that claims 3–5, 9–11, 38–41, and 45–48 are obvious under 35 U.S.C. § 103(a) in view of Hachamovitch. Pet. 45–49.

1. Hachamovitch (Ex. 1008)

Hachamovitch discloses a word completion utility that automatically predicts word completion for data entry in a data file, such as a word processor or email application. Ex. 1008, 4:10–13. The word completion system is used in conjunction with an individual application program or operates independently across multiple application programs. *Id*. at 4:21–25. The user's partially typed word is compared to the name-completion pairs and if a match is found within the list a suggestion list will be presented to a user. *Id*. at 4:58–5:6. A suggested word is presented to the user in a pop-up user interface within a word processing application as shown in Figs. 2A and 2B below.

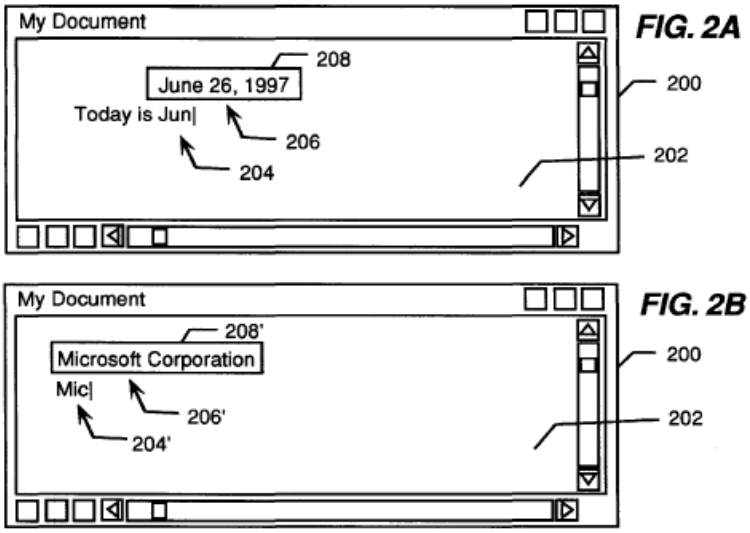


Figure 2A illustrates a word completion suggestion where the name completion pair is tied to a date or system parameter. *Id*. at 10:18-21; 10: 57-61. Figure 2B illustrates a word completion suggestion where the name-completion pair is tied to predefined properties, such as an initial capitalized letter. *Id*. at 11: 4-14. Once a word completion is displayed, the user may accept the word completion using an acceptance keystroke (e.g. tab or enter). Once accepted, the word completion utility replaces the partial data entry with the completion entry in the data file. *Id*. at 5:7-10; 7:4-5.

2. Anticipation Based on Hachamovitch (Ex. 1008)

Petitioner argues that Hachamovitch fails to disclose or teach a "second application program" as recited in independent claims 1, 7, 36, and 43, because the word completion utility is a subsidiary program that cannot be used independently of the host application. PO Resp. 41–42 (citing Ex. 1008, Fig. 1; Ex. 2003 ¶¶ 27–29). Patent Owner's arguments are based on its proposed claim interpretation that excludes subsidiary programs from the "second application program" recited in the challenged claims. PO Resp. 41–44. As discussed above, we disagree that the

construction of application program excludes subsidiary programs that are deployed within other applications or synchronously accessed by multiple applications. *See* Section II.A.3, *supra*.

Based on the record, we find that Petitioner has shown that Hachamovitch discloses that the word completion utility can be deployed as a "stand-alone" "application-independent" utility. Pet. 38 (Ex. 1008, 7:65–8:5). Hachamovitch states that "[t]o deploy the word completion system as an application-independent utility, an interface is defined within each application program through which the word completion utility may communicate with each application program." Ex. 1008, 8:6–9. Although the utility described in Hachamovitch operates through the application program, the reference expressly states that the utility can be part of the application program or operates independent of the applications as a "stand-alone" utility. Ex. 1008, 7:65–8:5. We find that the preponderance of the evidence indicates that "stand-alone" utility in Hachamovitch is an "application program" as recited in the claims.

We have reviewed Petitioner and Patent Owner's argument and evidence. We find that Petitioner has shown by a preponderance of the evidence that Hachamovitch anticipates claims 1–12 and 36–49. In addition, Petitioner has shown by a preponderance of the evidence that claims 3–5, 9–11, 38–41, and 45–48 are obvious in view of Hachamovitch.

D. Patent Owner's Motion to Exclude Evidence

Patent Owner moves to exclude the deposition transcript of Petitioner's expert, Dr. Daniel A. Menascé, Exhibit 1014. Paper 23. Dr. Menascé was deposed by Patent Owner's counsel on August 7, 2014 for this proceeding and for proceedings IPR2014-00206 and IPR2014-00207. Ex. 1014, 1. Patent Owner

contends the Menascé transcript is irrelevant and improper supplemental information that is not submitted in compliance with 37 C.F.R. § 42.123. Paper 22, 9–12.

With respect to Exhibit 1014, Petitioner argues and we agree that 37 C.F.R. § 42.53(f)(7) states that deposition testimony must be filed by its proponent as an exhibit. Paper 26, 3–4. Consistent with Petitioner's position, the rule recently has been clarified. *See* Amendments to the Rules of Practice for Trials Before the Patent Trial and Appeal Board, 80 Fed. Reg. 28,561, 28,563 (May 19, 2015) ("To clarify that either party is permitted to file testimony as an exhibit, the Office amends 37 CFR 42.53(f)(7) to delete the phrase 'by proponent' in the second sentence."). Because either party is permitted to file testimony as an exhibit, Petitioner's filing of the exhibit is proper. Accordingly, we *deny* Patent Owner's motion to exclude Exhibit 1014, the deposition transcript of Petitioner's expert, Dr. Daniel A. Menascé.

Patent Owner also seeks to exclude "Exhibit 1015," Visual Studio 2012 ("VS2012"), filed as an attachment to the deposition transcript of John V. Levy, Ph.D. (Ex. 1013).[2] Paper 23, 4–8. Patent Owner implicitly acknowledges that Petitioner's Reply to Patent Owner's Response does not expressly discuss or rely on VS2012. Paper 23, 5. Because we do not consider or rely on VS2012, or the portion of Dr. Levy's testimony discussing VS2012, in reaching our determinations in this Decision, Patent Owner's motion to exclude "Exhibit 1015" of Exhibit 1013 is *dismissed* as moot.

[2] Although the attachment is labelled "Exhibit 1015," VS2012 was not entered into the file as Exhibit 1015.

III. CONCLUSION

Petitioner has demonstrated, by a preponderance of the evidence, that claims 1–12 and 36–49 are unpatentable based on the following grounds of unpatentability:

(1) Claims 1, 2, 6–8 12, 36, 37, 42–44, and 49 under 35 U.S.C. § 102(e) as anticipated by Domini;

(2) Claims 1–12 and 36–49 under 35 U.S.C. § 102(e) as anticipated by Hachamovitch;

(3) Claims 3–5, 9–11, 38–41, and 45–48 under 35 U.S.C. § 103(a) for obviousness over Hachamovitch.

IV. ORDER

For the reasons given, it is

ORDERED that, based on a preponderance of the evidence, claims 1–12 and 36–49 of U.S. Patent No. 7,496,854 B2 are held unpatentable; and

FURTHER ORDERED that Patent Owner's motion to exclude "Exhibit 1015" is *dismissed*, and the motion to exclude Exhibit 1014, the deposition transcript of Dr. Daniel A. Menascé, is *denied*; and

FURTHER ORDERED that, because this is a Final Written Decision, parties to this proceeding seeking judicial review of our Decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

For Petitioner:

David L. Fehrman
dfehrman@mofo.com

Mehran Arjomand
marjomand@mofo.com

Matthew A. Smith
smith@turnerboyd.com

Zhuanjia Glu
gu@turnerboyd.com

For Patent Owner:

Robert Asher
rasher@sunsteinlaw.com

Bruce Sunstein
bsunstein@sunsteinlaw.com

Exhibit 6U

RECEIVED
CENTRAL FAX CENTER
DEC 30 2004

PATENTS
103176-0001C1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re The Application of:
Atle Hedloy

Serial No.: 09/923,134

Filed: August 6, 2001

For: METHOD, SYSTEM AND COMPUTER READABLE MEDIUM FOR ADDRESSING HANDLING FROM A COMPUTER PROGRAM

Examiner: Luu, Sy D

Art Unit: 2174

Cesari and McKenna, LLP
88 Black Falcon Avenue
Boston, MA 02210
December 30, 2004

CERTIFICATE OF TRANSMISSION

I hereby certify that the following paper is being facsimile transmitted to the Patent and Trademark Office on December 30, 2004.

Melissa Altman

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

SUPPLEMENTAL AMENDMENT

Please enter the following amendments in the above-identified patent application:

PATENTS
103176-0001C1

IN THE CLAIMS:

1. (Currently Amended): A method for information handling within a document created using a first application program comprising the steps of:

entering a first information in the first application program;

~~marking~~ analyzing without user intervention the first information for information to ~~alert the user that the first information can~~ be utilized in a second application program; and

responding to a user selection by inserting into the document a second information from the second application program ~~into the document~~, the second information associated with the first information ~~from a second application program~~.

2. (Currently Amended): The method of claim 1, wherein the user selection further comprises an activation of a device selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu, and a voice command device.

3. (Currently Amended): The method of claim 1, wherein the step of inserting the second information into the document further comprises the steps of:

~~initializing the second application program;~~

searching, using the second application program, for the second information associated with the first information; and

retrieving the second information.

4. (Original) The method of claim 3, wherein when the second application program includes second information associated with the first information, performing the further step of displaying the second information.

5. (Original): The method of claim 4, further comprising the step of:

completing at least one of the first and second information in the document.

6. (Original): The method of claim 1, wherein the first information comprises a name.

7. (Currently Amended): A computer readable medium~~, including program instructions, for performing the method of claim 1.~~ comprising program instructions for:

entering a first information in a document in a first application program;

analyzing without user information the first information for information to be utilized in a second application program; and

responding to a user selection by inserting into the document a second information from a second application program, the second information associated with the first information.

8. (Currently Amended): Electromagnetic signals propagating over a computer network, the electromagnetic signals carrying information for~~: practicing the method of claim 1.~~

entering a first information in a document in a first application program;

analyzing without user information the first information for information to be utilized in a second application program; and

responding to a user selection by inserting into the document a second information from a second application program, the second information associated with the first information.

9. (Currently Amended): ~~A computer readable medium, including program instructions, for performing the method of claim 2.~~ The computer readable medium of claim 7 wherein the user selection further comprises an activation of a device selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu, and a voice command device.

10. (Currently Amended): ~~A computer readable medium, including program instructions, for performing the method of claim 3.~~ The computer readable medium of claim 7 wherein inserting the second information into the document further comprises:

searching, using the second application program, for the second information associated with the first information; and

retrieving the second information.

11. (Currently Amended): ~~A computer readable medium, including program instructions, for performing the method of claim 4.~~ The computer readable medium of claim 7, wherein when

the second application program includes second information associated with the first information, performing the further step of displaying the second information.

12. (Currently Amended): ~~A computer readable medium, including program instructions, for performing the method of claim 5.~~ The computer readable medium of claim 11, further comprising completing at least one of the first and second information in the document.

13. (Currently Amended): ~~A computer readable medium, including program instructions, for performing the method of claim 6.~~ The computer readable medium of claim 7, wherein the first information comprises a name.

14. (Currently Amended): A method for information handling within a document created by a first application program comprising the steps of:

entering a first information in the first application program;

~~marking~~ analyzing without user intervention the first information for information to ~~alert the user that the first information can~~ be utilized in a second application program; and

responding to a user selection by performing an operation related to a second information, the second information associated with the first information from the second application program.

15. (Previously Presented): The method of claim 14, wherein the first information is a name, and the operation performed is selected from a group consisting of an electronic mail, a telex, a facsimile or a letter addressed to the name indicated by the first information.

16. (Previously Presented): The method of claim 14, wherein the operation performed is entering additional data into a database.

17. (Previously Presented): The method of claim 16, wherein the additional data is entered by a user.

18. (Previously Presented): The method of claim 16, wherein the additional data is located within the document.

19. (New): A method for information handling within a document created using a first application program, comprising:

analyzing, without user intervention, information entered in said document by a user for search information to be used to search for corresponding information stored in a second application program; and

commanding a function item to retrieve from said second application program information stored therein corresponding to said search information.

20. (New): The method of claim 19 in which said function item includes a device selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu selection, and a voice command device.

21. (New): The method of claim 19 in which the step of commanding said function item comprises performing a single click on said function item in a window or program on a computer screen.

22. (New): The method of claim 19 in which the step of commanding said function item comprises making a single selection in a menu in a program.

23. (New): The method of claim 19 which includes the step of adding to said search information in said first application program information retrieved from said second application program.

24. (New): The method of claim 23 in which said information added into said first application program is selected from the group consisting of a name and an address.

25. (New): The method of claim 19 which includes the step of displaying said information retrieved from said second application program.

26. (New): A computer readable medium comprising program instructions for

analyzing, without user intervention, information entered in said document by a user for search information to be used to search for corresponding information stored in a second application program; and

responding to a user's command of a function item by retrieving from said second application program information stored therein corresponding to said search information.

27. (New): A computer readable medium according to claim 26 in which said function item includes a device selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu selection, and a voice command device.

28. (New): A computer readable medium according to claim 26 in which the step of commanding said function item comprises performing a single click on said function item in a window or program on a computer screen.

29. (New): A computer readable medium according to claim 26 in which the step of commanding said function item comprises making a single selection in a menu in a program.

30. (New): A computer readable medium according to claim 26 which includes the step of adding to said search information in said first application program information retrieved from said second application program.

31. (New): A computer readable medium according to claim 30 in which said information added into said first application program is selected from the group consisting of a name and an address.

32. (New): A computer readable medium according to claim 26 which includes the step of displaying said information corresponding to said search information.

33. (New): Electromagnetic signals propagating over a computer network, the signals carrying information for:

analyzing, without user intervention, information entered in a document by a user of a first application program for search information to be used to search for corresponding information stored in a second application program; and

responding to a user's command of a function item by retrieving from said second application program information stored therein corresponding to said search information.

34. (New): Electromagnetic signals propagating over a computer network according to claim 33 which includes information for displaying said information corresponding to said search information.

PATENTS
103176-0001C1

REMARKS

On May 19, 2004, the Examiner issued a Notice of Allowance in this matter. However, an information disclosure statement filed by the Applicant on March 17, 2004 was lost by the USPTO and never delivered to the Examiner. After inquiring with the Examiner as to the March 17, 2004 IDS, the undersigned filed a Notice of Inquiry on June 17, 2004. The USPTO was not able to locate the IDS. To avoid abandonment of this application, Applicant filed a Request for Continued Examination and resubmitted the March 17, 2004 IDS on August 18, 2004.

By this Supplemental Amendment, the Applicant has amended claims 1-3 and 7-14 to better claim the invention. Applicant has added claims 19-34 in order to more fully define the invention. Independent claim 19 tracks the overall description of the invention set forth at page 3, lines 7 – 21 of the specification. Thus, "the first application program" of claim 1 corresponds to (but is not limited to) a program such as a word processor (see page 3, line 13). The "second application program" of claim 19 corresponds to (but is not limited to) a program such as a database program (see page 3, line 14). "[C]ommanding a function item" is accomplished, e.g., by singly clicking on it (page 3, lines 10 – 12) or by making a single selection in a menu (page 3, lines 11 – 12). Commanding the function item initiates a program associated with the item for retrieving from the second application program information related to the user-entered information. (page 3, lines 13 – 21). Analysis of the user-entered data is performed without user intervention to extract certain characterizing information to be used in the search. (page 7, lines 13 – 18). In contrast, as noted in our prior Responses in

this Application, Pandit, U.S. Patent No. 5,859,636, the principal reference cited by the Examiner but subsequently found to be distinguishable, requires the user to specifically mark the text to be used in retrieving related information form the second application program See, e.g., col. 2, l. 7-8 of Pandit which requires the user to designate the specific text to be used in the retrieval by "shading, underlining or pointing to and clicking on the text."

Independent claims 26 and 33 embody the limitations of claim 19 in the format of a computer-readable medium and electromagnetic signals, respectively.

Please charge any additional fee occasioned by this paper to our Deposit Account No. 03-1237.

Respectfully submitted,

Duane H. Dreger
Reg. No. 48,836
CESARI AND MCKENNA, LLP
88 Black Falcon Avenue
Boston, MA 02210-2414
(617) 951-2500

Exhibit 6V

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Hedloy	Attorney Docket:	3324/103
Serial No.:	11/745,186	Art Unit:	2166
Filing Date:	May 7, 2007	Examiner:	Pham
Invention:	METHOD, SYSTEM AND COMPUTER READABLE MEDIUM FOR ADDRESSING HANDLING FROM AN OPERATING SYSTEM		

Mail Stop After Final
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Response E

Dear Sir/Madam:

In response to the Office Action dated June 15, 2010, the Applicant submits the following amendment and remarks.

Table of Contents begins on page 2.

Amendments to the Claims are reflected in the listing of claims which begin on page 3 of this paper.

Remarks begin on page 15 of this paper.

TABLE OF CONTENTS

AMENDMENTS TO THE CLAIMS

Claims 1-106. (Cancelled)

107. (Previously Presented) A computer implemented method for information handling, comprising:

displaying information electronically, using a computer program;

electronically analyzing the information to identify a portion of that information as contact information and to determine what type of contact information the portion is, without user designation of a specific part of the electronically displayed information to be subject to the analyzing;

electronically searching in an information source for the contact information in order to find whether the contact information is included in the information source; and

when the information source includes the contact information, if second information in the information source is associated with the contact information, causing electronic display of at least a portion of the second information.

108. (Previously Presented) A method according to claim 107, further comprising:

during the displaying, receiving an execute command from an input device that initiates at least one process of this method.

109. (Previously Presented) A method according to claim 107, further comprising:

in a computer process, performing an action depending on the type of contact information the portion is.

110. (Currently Amended) A method according to claim 109 ~~110~~, wherein the portion is a name and the action includes insertion of an address into the displayed information.

111. (Previously Presented) At least one non-transitory computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, comprising:

displaying information electronically, using a computer program;

electronically analyzing the information to identify a portion of that information as contact information and to determine what type of contact information the portion is, without user designation of a specific part of the electronically displayed information to be subject to the analyzing;

electronically searching in an information source for the contact information in order to find whether the contact information is included in the information source; and

when the information source includes the contact information, if second information in the information source is associated with the contact information, causing electronic display of at least a portion of the second information.

112. (Previously Presented) At least one non-transitory computer readable medium according to claim 111, wherein the instructions establish processes further comprising:

during the displaying, receiving an execute command from an input device that initiates at least one process for which instructions are stored in the computer readable medium.

113. (Previously Presented) At least one non-transitory computer readable medium according to claim 111, wherein the instructions establish processes further comprising:

in a computer process, performing an action depending on the type of contact information the portion is.

114. (Previously Presented) At least one non-transitory computer readable medium according to claim 113, wherein the portion is a name and the action includes insertion of an address into the displayed information.

115. (Previously Presented). An apparatus for information handling, comprising:

a processor; and

a memory storing instructions executable by the processor to perform processes that include:

displaying information electronically, using a computer program;

electronically analyzing the information to identify a portion of that information as contact information and to determine what type of contact

information the portion is, without user designation of a specific part of the electronically displayed information to be subject to the analyzing;

electronically searching in an information source for the contact information in order to find whether the contact information is included in the information source; and

when the information source includes the contact information, if second information in the information source is associated with the contact information, causing electronic display of at least a portion of the second information.

116. (Previously Presented) An apparatus according to claim 115, wherein the memory further stores instructions executable by the processor to perform processes that include:

during the displaying, receiving an execute command from an input device that initiates at least one process for which instructions are stored in the memory.

117. (Previously Presented) An apparatus according to claim 115, wherein the memory further stores instructions executable by the processor to perform processes that include:

in a computer process, performing an action depending on the type of contact information the portion is.

118. (Previously Presented) An apparatus according to claim 117, wherein the portion is a name and the action includes insertion of an address into the displayed information.

119. (Previously Presented) A computer implemented method for information handling, comprising:

analyzing in a computer process information electronically displayed to identify a portion of that information as contact information, without user designation of a specific part of the electronically displayed information to be subject to the analyzing, wherein the contact information is at least one of a name, a title, an address, a telephone number, and an email address;

electronically searching in an information source for the contact information in order to find whether the contact information is included in that information source; and

when the information source includes the contact information, if second information in the information source is associated with that contact information, electronically displaying at least a portion of the second information, wherein the second information is at least one of a name, a title, an address, a telephone number, and an email address.

120. (Previously Presented) A method according to claim 119, further comprising:

receiving an execute command from an input device that initiates at least one process of this method.

121. (Previously Presented) A method according to claim 119, wherein the method is implemented in a client running a program, the client selected from a group consisting of a computer, a cell phone, a palm top device, and a personal organizer.

122. (Previously Presented) A method according to claim 121, wherein the contact information is a name, the second information is an address, and the client is a computer.

123. (Previously Presented) A method according to claim 121, wherein the contact information is a telephone number.

124. (Previously Presented) A method according to claim 121, wherein the contact information is a telephone number, the second information is a name, and the client is a cell phone.

125. (Previously Presented) At least one non-transitory computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, comprising:

analyzing in a computer process information electronically displayed to identify a portion of that information as contact information, without user designation of a specific part of the electronically displayed information to be subject to the analyzing, wherein the contact information is at least one of a name, a title, an address, a telephone number, and an email address;

electronically searching in an information source for the contact information in order to find whether the contact information is included in that information source; and

when the information source includes the contact information, if second information in the information source is associated with that contact information, electronically displaying at least a portion of the second information, wherein the second information is at least one of a name, a title, an address, a telephone number, and an email address.

126. (Previously Presented) At least one non-transitory computer readable medium according to claim 125, wherein the instructions establish processes further comprising:

receiving an execute command from an input device that initiates at least one process for which instructions are stored in the computer readable medium.

127. (Previously Presented) At least one non-transitory computer readable medium according to claim 125, wherein the at least one non-transitory computer readable medium is embodied in a client running a program, the client selected from a group consisting of a computer, a cell phone, a palm top device, and a personal organizer.

128. (Previously Presented) At least one non-transitory computer readable medium according to claim 127, wherein the contact information is a name, the second information is an address, and the client is a computer.

129. (Previously Presented) At least one non-transitory computer readable medium according to claim 127, wherein the contact information is a telephone number.

130. (Previously Presented) At least one non-transitory computer readable medium according to claim 127, wherein the contact information is a telephone number, the second information is a name, and the client is a cell phone.

131. (Previously Presented) An apparatus for information handling, comprising:

a processor; and

a memory storing instructions executable by the processor to perform processes that include:

analyzing in a computer process information electronically displayed to identify a portion of that information as contact information, without user designation of a specific part of the electronically displayed information to be subject to the analyzing, wherein the contact information is at least one of a name, a title, an address, a telephone number, and an email address;

electronically searching in an information source for the contact information in order to find whether the contact information is included in that information source; and

when the information source includes the contact information, if second information in the information source is associated with that contact information, electronically displaying at least a portion of the second information, wherein the

second information is at least one of a name, a title, an address, a telephone number, and an email address.

132. (Previously Presented) An apparatus according to claim 131, wherein the memory further stores instructions executable by the processor to perform processes that include:

receiving an execute command from an input device that initiates at least one process for which instructions are stored in the computer readable medium.

133. (Previously Presented) An apparatus according to claim 131, wherein the apparatus is selected from a group consisting of a computer, a cell phone, a palm top device, and a personal organizer.

134. (Previously Presented) An apparatus according to claim 133, wherein the contact information is a name, the second information is an address, and the apparatus is a computer.

135. (Previously Presented) An apparatus according to claim 133, wherein the contact information is a telephone number.

136. (Previously Presented) An apparatus according to claim 133, wherein the contact information is a telephone number, the second information is a name, and the apparatus is a cell phone.

137. (Currently Amended) A computerized method for information handling, the method comprising:

displaying information in a document electronically using a computer program;

electronically analyzing the information to identify a portion of that information as contact information including at least one of a name without an address and a name with an address;

providing an input device configured to allow ~~the~~ a user to use the input device to command the ~~system~~ program to perform at least one of:

i) inserting address information from an information source and associated with the name into the document, and

ii) storing at least part of the contact information in the information source;

during the displaying, receiving an execute command from the input device, wherein accessing and manipulating the input device are the only user actions required to cause initiation and completion of the analyzing; and

if the contact information is identified as including a name without an address, electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes the name, if address information in the information source is associated with the name, causing insertion of the address information into the document;

if the contact information is identified as including a name with an address, i) electronically prompting the user with an option to save electronically in the information source at least some of the ~~contract~~ contact information, and ii) electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes at least one contact with the name, prompting the user to make a decision whether to store the name and address as a new contact or to update one of the at least one contact.

138. (Currently Amended) At least one non-transitory computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, comprising:

displaying information in a document electronically using a computer program;

electronically analyzing the information to identify a portion of that information as contact information including at least one of a name without an address and a name with an address;

providing an input device configured to allow ~~the~~ a user to use the input device to command the ~~system~~ program to perform at least one of:

i) inserting address information from an information source and associated with the name into the document, and

ii) storing at least part of the contact information in the information source;

during the displaying, receiving an execute command from the input device, wherein accessing and manipulating the input device are the only user actions required to cause initiation and completion of the analyzing; and

if the contact information is identified as including a name without an address, electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes the name, if address information in the information source is associated with the name, causing insertion of the address information into the document;

if the contact information is identified as including a name with an address, i) electronically prompting the user with an option to save electronically in the information source at least some of the ~~contract~~ contact information, and ii) electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes at least one contact with the name, prompting the user to make a decision whether to store the name and address as a new contact or to update one of the at least one contact.

REMARKS

The Applicant thanks Examiner Pham for his analysis of the pending claims and for his time during the examiner interview on July 20, 2010, in which the Hachamovitch reference was discussed with Bruce Sunstein and Jakub Michna, attorneys for the applicant, and Atle Hedloy, the inventor and applicant herein. No agreement was reached concerning the pending claims, and the present response is submitted to show that the rejection of the pending claims on the basis of Hachamovitch is improper. For the reasons set forth below, the rejection is traversed, and reconsideration and allowance of the claims are respectfully requested.

Claims 107-138 are currently pending. Claims 110, 137, and 138 have been amended. The amendments to the claims are to correct obvious defects giving rise to the objection in the outstanding office action.

The following remarks will first, in section I, discuss the pending claims, and then, in section II, demonstrate that Hachamovitch fails to disclose or suggest the subject matter defined by the pending claims.

I. The pending claims require two distinct processes: "analyzing" for "contact information" and then "searching" for that "contact information" in an information source.

Claims 107-110 all require "analyzing" to "identify" certain information as "contact information" and to "determine" the "type" of contact information that has been identified:

> electronically **analyzing** the information to **identify** a portion of that information as **contact information** and to **determine** what **type** of

> contact information the portion is, without user designation of a specific part of the electronically displayed information to be subject to the analyzing;

Claim 107, 2d subparagraph (emphasis added).

Thereafter, claims 107-110 all require "searching" for that same "contact information" in an information source to determine "whether" that very same contact information first "identified" and "determined" to be of a particular type in the "analyzing" step, is also included in that "information source":

> electronically **searching** in an **information source** for **the** contact information in order to find **whether** **the** contact information is included in the information source.

Claim 107, 3d subparagraph (emphasis added).

Claims 107-110 thus require two distinct processes: "analyzing" for "contact information" and then "searching" for that "contact information" in an information source. Claims 111-118 have similar limitations, but are written in computer readable media and apparatus format.

Claims 119-124 are similar to claims 107-110 in requiring both analyzing to identify contact information and thereafter searching for that contact information in an information source, but claims 119-124 are more specific in requiring the contact information to be at least one of a "name," "title", "address," "telephone number," and "e-mail address":

> **analyzing** in a computer process information electronically displayed to **identify** a portion of that information as **contact information**, without user designation of a specific part of the electronically displayed information to be subject to the analyzing, wherein the **contact information** is at least one of a **name**, a **title**, an **address**, a **telephone number**, and an **email address**;

Claim 119, 1st subparagraph (emphasis added).

Again, the claims require two distinct processes: "analyzing" for "contact information" and then "searching" for that "contact information" in an information source. Claims 125-136 have similar limitations, but are written in computer readable media and apparatus format.

Claims 137 (a method claim) and 138 (a corresponding media claim) are more specific yet, requiring the process of "analyzing" to "identify" "contact information including at least one of a name without an address and a name with an address" and then performing a series of processes depending on whether there is or is not an associated address, including processes, where applicable, that enable updating the "information source" with a "new contact." Moreover, after the process of analyzing to identify contact information, these claims, like all of the other claims, require searching in an information source for the contact information identified in the analyzing process.

II. Hachamovitch (1) fails to teach the claim requirement of analyzing to identify contact information, and (2) fails to teach the claim requirement of searching for the contact information thus identified.

We have shown that every pending claim herein requires (1) analyzing to identify contact information and then (2) searching in an information source to find the contact information identified in the analyzing process. Hachamovitch teaches neither of these processes, and for that reason does not anticipate or render obvious the subject matter defined by the pending claims.

Nothing in Hachamovitch remotely resembles the requirement of the claims for analyzing to identify contact information to be used in a search. Hachamovitch is rather

directed to a word completion system that searches a suggestion list each time a sufficiently long character string has been entered after a delimiter. (See Hachamovitch Abstract, first sentence; and text describing Fig. 5, including col. 14, line 55 to col. 16, line 7.) Hachamovitch searches the suggestion list to find an entry matching the character string entered by the user, and if a match occurs, offers the user the option of inserting a completion entry on the list. Although the Hachamovitch system can address context restraints under limited circumstances, these constraints operate only to limit results of the search for a match,(*e.g.*, ruling out "Very truly yours" in the heading of a document). The constraints have no effect on the search itself, which is performed invariably after a sufficient number of keystrokes has been entered by the user.[1]

The step in Hachamovitch of searching its suggestion list for a match with user-entered keystrokes simply cannot be equated with the claim requirement of analyzing information to identify contact information, because in Hachamovitch's system, once the minimum number of characters has been entered in a string of keystrokes, the search is performed, without any prior "analysis" of the string of keystrokes to identify them as "contact information." If the minimum number of characters is set at 3, then once 3

---

[1] These features relate to context, capitalization, and similar properties. (Col. 7, lines 31-40.) The capitalization limitation, for example, inhibits display of a completion suggestion that is capitalized if the key stokes at issue are not likewise capitalized (See step 512 of Fig. 5 and related text.) Importantly, the handling of context, capitalization, and similar properties in Hachamovitch in no way makes any distinction between "contact information" as required by the claims, and any other type of "name entry" or "completion entry" on the Hachamovitch suggestion list. Thus, 'Sincerely yours,' 'Very truly yours,' 'Cordially yours,' and 'Microsoft' are all capitalized, but that fact does not indentify Microsoft as "contact information," much less contact information of a particular "type." Moreover, the capitalization and context limitations are imposed to inhibit display of a completion suggestion only after the search in process 506 has already been completed. This fact is illustrated in the logical flow of the system in Fig. 5. Once the minimum number of characters is detected in process 504, the system runs the comparison, in process 506, that constitutes a search of the name entries in the suggestion list. Then there are tests for unambiguous match in process 508 and in process 510 for sufficient additional characters in the completion entry beyond those in the string entered by the user. Next, as the final step before displaying the completion suggestion in process 514, there is the test process 512 for matching of capitalization and context of the entered string with the requirements specified for the name entry in the suggestion list.

characters have been entered by the user after the delimiter, the search is performed for an entry on the suggestion list beginning with those same 3 characters, whether or not those characters having anything to do with "contact information."

While the Examiner is permitted to give the term "contact information" as used in the claims pending herein its broadest reasonable interpretation, the interpretation must also be *consistent with the specification* and *consistent with the ordinary usage* of the term as understood by one of ordinary skill in the art. (See MPEP § 2111). Among other things, the specification refers to names, addresses, telephone numbers, fax numbers, and e-mail addresses as information related to contacts. (See application, page 7, lines 2-22; page 8, lines 7-12.) Such information is related to a contact, such as an individual, a group, a company, or association, and the like. This is the ordinary meaning of "contact information" as used in the specification and as understood by one of ordinary skill in the art.

Each item on the Hachamovitch suggestion list is a "name entry" paired with a "completion entry." (See text describing Fig. 3, including col. 11, lines 38-50.) Hachamovitch sometimes calls the "completion entry" a "completion suggestion". (Col. 11, lines 38-50.) The term "name entry" in Hachamovitch has no specific meaning in relation to contact information, because the name entry defines simply the key-stroke sequence against which the entered character string is compared. We know this because, as seen in Fig. 3, the suggestion list includes both names, which are a form of "contract information," and non-names, such as "Very truly yours," which having nothing what so ever to do with "contact information." Thus, the fact a keystroke sequence has correspondence on the Hachamovitch suggestion list tells nothing about the character of

that keystroke sequence or the corresponding suggestion list entries except, at most, that it is or is not capitalized. There is no indentifying of "contact information" or searching of previously identified "contact information" as the claims require.

Thus, in performing the search in process 506, of Fig. 5, Hachamovitch makes no discrimination among different types of information, but seeks only to find a match between the string of keystrokes entered since the last delimiter and an entry in the suggestion list. Nothing in Hachamovitch meets the claimed requirement of analyzing to identify contact information to be searched. Hachamovitch fails to meet this limitation.

Furthermore, every word in the claim must have a meaning, and the word "contact" in the phrase "contact information" must also have meaning which makes "contact information" different from just "information." (MPEP § 2143.03 ("All words in a claim must be considered in judging the patentability of that claim against the prior art.")) Thus "contact information" must have to do with "contacts" and cannot mean, for example, every combination of characters longer than a specific length (*e.g.*, 3), nor phrases such as "Very truly yours" or "Save the Whales Symposium," which also appear as entries in the suggestion list in Fig. 3 of Hachamovitch.

It might be argued that the claimed process of "analyzing" to "identify" "contact information" can be found in the consideration of capitalization and context by the Hachamovitch system. (See footnote 1 above.) As discussed in footnote 1, however, capitalization and context are addressed by Hachamovitch only after searching has been completed and are used only to inhibit display of completion entries. But, as discussed in section I, the claims require that after such "analyzing" to "identify" "contact information", there must follow the process of searching in an information source to find

the contact information identified in the analyzing process. Because there is no searching in Hamachovitch after its consideration of capitalization and context, such consideration fails to meet the claim requirement of "analyzing" to "identify" "contact information". In other words, the claims require a searching process that targets the contact information identified by the prior analyzing process. Hachamovitch does not target a search to contact information identified by an analyzing process. Once the minimum number of characters has been entered, the Hachamovitch system searches for any entry in the suggestion list beginning with characters matching the entered string, whether it is contact information or not.

In addition, the way the Hachamovitch system uses constraint information is indifferent to whether contact information happens to be involved, and thus does not invoke a process of analyzing to identify contact information as required by the claims. At most, Hachamovitch identifies a context in which a string typed by the user occurs, but it is the user who has defined the context by typing in a region having a paragraph style label assigned via a Wizard invoked by the user. (See col. 5, lines 18-36.) The Hachomovitch system operates in the same manner when the user types in the salutation field as when the user types in the addressee field. There is no analyzing to identify contact information, as required by the claims. Similarly, capitalization in Hachamovitch makes no distinction between "contact information" as required by the claims, and any other type of "name entry" or "completion entry" on the Hachamovitch suggestion list. Thus, 'Sincerely yours,' 'Very truly yours,' 'Cordially yours,' and 'Microsoft' are all capitalized, but that fact does not indentify Microsoft as "contact information," much less

contact information of a particular "type." Nor does Hachamovitch disclose or suggest any reason to make any such identification.

Let us recap: the pending claims require (1) "analyzing" to "identify" "contact information" and (2) "searching" in an information source for that same contact information. Hachamovitch fails to teach these processes. Although Hachamovitch does search a suggestion list to see if there are entries on it corresponding to keystrokes entered by the user, if this step is considered to correspond to the claim requirement (2) of "searching" in an information source for that same contact information, then where is process (1) required by the pending claims—"analyzing" to "identify" that same "contact information"? Determining that three keystrokes have been entered may be a form of "analyzing," but it falls far short of the claim requirement of analyzing to "identify" those keystrokes as "contact information" before there is any "search" for that same "contact information" in an information source.

Thus, Applicant respectfully submits that this counting of keystrokes fails to meet the claim limitation of "analyzing" to "identify" "contact information." In fact, in claims 137 and 138 the "analyzing" process requires analyzing specifically to identify a "name" and that the name is with an address or without an address—a required process going far beyond counting keystrokes in Hachamovitch.

If, by some stretch of imagination, the searching step in Hachamovitch were argued to disclose searching for "contact information," Hachamovitch would still fail to disclose searching for "the" contact information that the claims require be identified by a previous step of first "analyzing" to "identify" that "contact information."

On the other hand, if, by some equally large stretch of imagination, Hachamovitch's search of a suggestion list to see if there are entries on it corresponding to user-entered keystrokes were argued to correspond to the claim requirement of "analyzing" to "identify" "contact information," then there are at least two problems with the argument. First, again, the keystroke matching of Hachamovitch does not identify contact information, since the match of entries on the suggestion list in Hachamovitch can as well single out "Very truly yours" and "Save the Whales Symposium" as anything else. The matching process does not result in identification of contact information, but rather only of a match of keystrokes to the suggestion list.

Although contact information may be included in the Hachamovitch suggestion list, and may be matched to the keystrokes, it is never *identified* as contact information as required by the claims. Only when the *user* sees and recognizes a matched suggestion list entry, such as Microsoft, as a company name is there any identification of that entry as "contact information." But all the claims require that the "analyzing" be done "electronically" —not by user intervention.

Second, the claims also require, after analyzing to identify contact information, "searching" an information source for the "contact information" previously identified in the analyzing process. But if the Hachamovitch step of seeking a keystroke match in suggestion list entries corresponds to the analyzing process required by the claims, there is no subsequent "searching" as the pending claims require--the match has already been found in the process of "analyzing."

Hachamovitch cannot be construed to include both limitations of the pending claims—the Hachamovitch step of seeking a keystroke match in suggestion list entries

cannot be BOTH the "analyzing" and "searching" processes required by the claims pending herein. In fact, for the reasons expressed above, the Hachamovitch step of seeking a keystroke match in suggestion list entries meets neither claim limitation (1), "analyzing" to "identify" "contact information," nor claim limitation (2), "searching" in an information source for that same contact information.

In summary, for at least the reasons mentioned above, Hachamovitch fails to meet limitations of the pending claims, and thus neither anticipates nor renders obvious the subject matter defined by the pending claims.

III. Conclusion.

Applicant believes that all of the rejections have been addressed and a notice of allowance is respectfully solicited. If any fees are required, please charge deposit account number 19-4972. To further expedite prosecution, the Examiner may call Bruce Sunstein or Jakub Michna at 617-443-9292 if he has any further questions.

Respectfully submitted,

/Bruce D. Sunstein, #27,234/

Bruce D. Sunstein
Registration No. 27,234

/Jakub M. Michna, #61,033/

Jakub M. Michna
Registration No. 61,033

Attorneys for Applicant

SUNSTEIN KANN MURPHY
& TIMBERS LLP
125 Summer Street
Boston MA 02110-1618
Tel: 617 443 9292
Fax: 617 443 0004
03324/00103 1298521.4

Exhibit 6W

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Hedloy	Attorney Docket:	3324/103
Serial No.:	11/745,186	Art Unit:	2166
Filing Date:	May 7, 2007	Examiner:	Saeed, Usmaan
Invention:	METHOD, SYSTEM AND COMPUTER READABLE MEDIUM FOR ADDRESSING HANDLING FROM AN OPERATING SYSTEM		

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Response C

Dear Sir/Madam:

In response to the Final Office Action dated March 26, 2009, the Applicant submits the following amendment and remarks.

Amendments to the Claims are reflected in the listing of claims which begin on page 2 of this paper.

Remarks begin on page 17 of this paper.

Amendments to the Claims

Listing of Claims:

Claims 1-28. (Cancelled)

29. (Currently Amended) A computerized method for information handling, comprising:

analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search;

retrieving the search criteria;

displaying the document using a first computer program;

on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria; and

when at least one of the local and remote information sources includes second information associated with the search criteria, causing display of at least some of the second information.

30. (Previously Presented) The method of claim 29 wherein the first computer program is the same as the second computer program.

31. (Previously Presented) The method of claim 29 further comprising storing the search criteria in the local information source when no second information associated with the search criteria is found in the local and remote information sources.

32. (Previously Presented) The method of claim 29 further comprising making changes, by the user, to the second information directly in the local information source.

33. (Previously Presented) The method of claim 29 wherein the search criteria is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address.

34. (Previously Presented) The method of claim 29 wherein the second information is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address associated with the search criteria.

35. (Previously Presented) The method of claim 29 wherein the input device is selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu and a voice command device.

36. (Previously Presented) The method of claim 29 wherein the input device is located on a device selected from a group consisting of a computer, a cell phone and a palm top device.

37. (Currently Amended) A system for information handling, comprising:

means for analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search;

means for retrieving the search criteria;

means for displaying the document using a first computer program;

means for, on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria; and

means for causing display of at least some of the second information when at least one of the local and remote information sources includes second information associated with the search criteria.

38. (Previously Presented) The system of claim 37 further comprising means for storing the search criteria in the local information source when no second information associated with the search criteria is found in the local and remote information sources.

39. (Previously Presented) The system of claim 37 further comprising means for making changes, by the user, to the second information directly in the local information source.

40. (Previously Presented) The system of claim 37 wherein the search criteria is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address.

41. (Previously Presented) The system of claim 37 wherein the second information is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address associated with the search criteria.

42. (Cancelled)

43. (Cancelled)

44. (Currently Amended) At least one computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, comprising:

analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search;

retrieving the search criteria;

displaying the document using a first computer program;

on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second

computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria; and

when at least one of the local and remote information sources includes second information associated with the search criteria, causing display of at least some of the second information.

45. (Previously Presented) The at least one computer readable medium of claim 44 wherein the first computer program is the same as the second computer program.

46. (Previously Presented) The at least one computer readable medium of claim 44, wherein the instructions establish processes further comprising storing the search criteria in the local information source when no second information associated with the search criteria is found in the local and remote information sources.

47. (Previously Presented) The at least one computer readable medium of claim 44, wherein the instructions establish processes further comprising making changes, by the user, to the second information directly in the local information source.

48. (Previously Presented) The at least one computer readable medium of claim 44 wherein the search criteria is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address.

49. (Previously Presented) The at least one computer readable medium of claim 44 wherein the second information is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address associated with the search criteria.

50. (Previously Presented) The at least one computer readable medium of claim 44 wherein the input device is selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu and a voice command device.

51. (Previously Presented) The at least one computer readable medium of claim 44 wherein the input device is located on a device selected from a group consisting of a computer, a cell phone and a palm top device.

52. (Currently Amended) A computerized method for information handling, comprising:

analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search;

retrieving the search criteria;

displaying the document using a first computer program;

on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria; and

when at least one of the local and remote information sources includes second information associated with the search criteria, performing an operation related to at least some of the second information;

wherein the first computer program runs on a device selected from a group consisting of a computer, cell phone, or palm top device;

wherein the input device is selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu and a voice command device.

53. (Cancelled).

54. (Currently Amended) At least one computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, comprising:

analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search;

retrieving the search criteria;

displaying the document using a first computer program;

on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria; and

when at least one of the local and remote information sources includes second information associated with the search criteria, performing an operation related to at least some of the second information;

wherein the first computer program runs on a device selected from a group consisting of a computer, cell phone, or palm top device;

wherein the input device is selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu and a voice command device.

55. (Cancelled).

56. (Currently Amended) A system for information handling, comprising:

means for analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search;

means for retrieving the search criteria;

means for displaying the document using a first computer program;

means for, on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria; and

means for performing an operation related to at least some of the second information when at least one of the local and remote information sources includes second information associated with the search criteria;

wherein the first computer program runs on a device selected from a group consisting of a computer, cell phone, or palm top device;

wherein the input device is selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu and a voice command device.

57. (Cancelled).

58. (Previously Presented) The method of claim 29 further comprising adding the second information to the search criteria in the document.

59. (Previously Presented) The method of claim 29 wherein displaying includes displaying the second information in the first computer program.

60. (Previously Presented) The method of claim 29, further comprising providing a prompt configured to enable the second computer program to include the search criteria in at least one of the local and remote information sources.

61. (Previously Presented) The method of claim 29, further comprising in response to the search not being successful, providing a prompt configured to enable updating at least one of the local and remote information sources to include the search criteria.

62. (Previously Presented) The method of claim 29, further comprising when the second information from the remote information source is different from the second information from the local data source, updating the local information source.

63. (Previously Presented) The method of claim 29 wherein using the input device to initiate searching precedes analyzing the document.

64. (Previously Presented) The method of claim 29 wherein analyzing the document is completed after using the input device and before searching is initiated.

65. (Previously Presented) The method of claim 29 wherein the execute command is the only command from a user necessary as a condition to cause the display of at least

some of the second information.

66. (Previously Presented) The method of claim 29 wherein the input device is a menu, and the entry of the execute command includes a user's selection of the menu and click on a menu choice from the menu.

67. (Previously Presented) The method of claim 29 further comprising, when searching results in a plurality of distinct instances of second information, causing display of such instances to enable user selection of one of them for use in performing the display.

68. (Previously Presented) The method of claim 52 wherein the operation comprises adding the second information to the search criteria in the document.

69. (Previously Presented) The method of claim 52 wherein performing the operation comprises causing the display of at least some of the second information in the first computer program.

70. (Previously Presented) The method of claim 52 wherein the operation comprises providing a prompt configured to enable the second computer program to include the search criteria in at least one of the local and remote information sources.

71. (Previously Presented) The method of claim 52 wherein the operation comprises in response to the search not being successful, providing a prompt configured to enable updating at least one of the local and remote information sources to include the search criteria.

72. (Previously Presented) The method of claim 52 further comprising, when the second information from the remote information source is different from the second information from the local data source, updating the local information source.

73. (Previously Presented) The method of claim 52 wherein the operation comprises when searching results in a plurality of distinct instances of second information, causing display of such instances to enable user selection of at least one of them for use in performing the operation.

74. (Currently Amended) A computerized method for information handling, comprising:

analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search, wherein the search criteria is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address;

retrieving the search criteria;

displaying the document using a first computer program;

on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria, wherein the second information is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address associated with the search criteria; and

wherein the input device is selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu and a voice command device;

wherein the first computer program is executed on a device selected from a group consisting of a computer, cell phone, or palm top device;

performing at least one of: (a) comparing the second information from the local information source with second information from the remote information source when second information is found in both the local and the remote information sources and (b) causing display of at least some of the second information when at least one of the local and remote information sources includes second information associated with the search criteria.

75. (Previously Presented) The method of claim 74 further comprising when searching results in more than one distinct instances of second information, causing display of such instances to enable user selection of at least one of the instances for use in performing the operation.

76. (Previously Presented) The method of claim 74 further comprising performing an action associated with at least one of the second information from the local and remote information sources.

77. (Previously Presented) The system of claim 37 further comprising means for adding the second information to the search criteria in the document.

78. (Previously Presented) The system of claim 37 wherein the means for displaying includes displaying the second information in the first computer program.

79. (Previously Presented) The system of claim 37 further comprising means for providing a prompt configured to enable the second computer program to include the search criteria in at least one of the local and remote information sources.

80. (Previously Presented) The system of claim 37 further comprising when the search is not successful, means for providing a prompt configured to enable updating at least one of the local and remote information sources to include the search criteria.

81. (Previously Presented) The system of claim 37 further comprising when the second information from the remote information source is different from the second information from the local data source, updating the local information source.

82. (Previously Presented) The at least one computer readable medium of claim 44 wherein the operation comprises adding the second information to the search criteria in the document.

83. (Previously Presented) The at least one computer readable medium of claim 44 wherein performing the displaying includes displaying the second information in the second computer program.

84. (Previously Presented) The at least one computer readable medium of claim 44, wherein the instructions establish processes further comprising providing a prompt configured to enable the second computer program to include the search criteria in at least one of the local and remote information sources.

85. (Previously Presented) The at least one computer readable medium of claim 44, wherein the instructions establish processes further comprising, when the search is not successful, providing a prompt configured to enable updating at least one of the local and remote information sources to include the first information.

86. (Previously Presented) The at least one computer readable medium of claim 44, wherein the instructions establish processes further comprising, when the second information from the remote information source is different from the second information from the local data source, updating the local information source.

87. (Previously Presented) The at least one computer readable medium of claim 44 wherein using the input device to initiate searching precedes analyzing the document.

88. (Previously Presented) The at least one computer readable medium of claim 44 wherein analyzing the document is completed after using the input device and before searching is initiated.

89. (Previously Presented) The at least one computer readable medium of claim 44 wherein the user command is the only command from a user necessary as a condition to cause display of at least some of the second information.

90. (Previously Presented) The at least one computer readable medium of claim 44

wherein the input device is a menu, and the entry of the user command includes a user's selection of the menu and click on a menu choice from the menu.

91. (Previously Presented) The at least one computer readable medium of claim 44, wherein the instructions establish processes further comprising, when searching results in a plurality of distinct instances of second information, causing display of such instances to enable user selection of at least one of them for use in performing the display.

92. (Currently Amended) A system for information handling, comprising:

means for analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search, wherein the search criteria is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address;

means for retrieving the search criteria;

means for displaying the document using a first computer program;

means for, on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria; and

wherein the input device is selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu and a voice command device;

wherein the second information is selected from a group consisting of a person name, a company name, a title, an address, a telephone number and an email address associated with the search criteria;

means for performing at least one of: (a) comparing the second information from the local information source with second information from the remote information source when second information is found in both the local and the remote information sources and (b) causing display of at least some of the second information when at least one of the local and remote information sources includes second information associated with the search criteria.

93. (Currently Amended) At least one computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, comprising:

analyzing in a computer process, without user designation, information in a document to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search;

retrieving the search criteria;

displaying the document using a first computer program;

on receipt, by ~~a~~ the first computer program displaying the document, of an execute command from an input device, searching for the search criteria using a second computer program, in at least one of a local and a remote information source, in order to find second information associated with the search criteria; and

performing at least one of:

(a) comparing the second information from the local information source with second information from the remote information source when second information is found in both the local and the remote information sources; and

(b) performing an operation related to the second information, the second information associated with the search criteria from the second computer program;

wherein the first computer program runs on a device selected from a group consisting of a computer, cell phone, or palm top device;

wherein the input device is selected from a group consisting of a touch screen, a keyboard button, a screen button, an icon, a menu and a voice command device.

94. (Previously Presented) The at least one computer readable medium of claim 93, wherein the instructions establish processes further comprising adding the second information to the search criteria in the document.

95. (Previously Presented) The at least one computer readable medium of claim 93 wherein performing the operation includes displaying the second information in the first computer program.

96. (Previously Presented) The at least one computer readable medium of claim 93, wherein the instructions establish processes further comprising providing a prompt configured to enable the second computer program to include additional information in at least one of the local and remote information sources.

97. (Previously Presented) The at least one computer readable medium of claim 93, wherein the instructions establish processes further comprising providing a prompt configured to enable the second computer program to include the search criteria in at least one of the local and remote information sources.

98. (Previously Presented) The at least one computer readable medium of claim 93, wherein the instructions establish processes further comprising, when the search is not successful, providing a prompt configured to enable updating at least one of the local and remote information sources to include the search criteria.

99. (Previously Presented) The at least one computer readable medium of claim 93, wherein the instructions establish processes further comprising, when the second information from the remote information source is different from the second information from the local data source, updating the local information source.

100. (Previously Presented) The system of claim 37 wherein the first computer program is the same as the second computer program.

101. (Previously Presented) The method according to claim 29, wherein searching using the second computer program includes searching in both the local and the remote information source.

102. (Previously Presented) The system according to claim 37, wherein searching using the second computer program includes searching in both the local and the remote information source.

103. (Previously Presented) The at least one computer readable medium according to claim 44, wherein searching using the second computer program includes searching in both the local and the remote information source.

104. (New) The method of claim 29 wherein the first computer program is different from the second computer program.

105. (New) The system of claim 37 wherein the first computer program is different from the second computer program.

106. (New) The at least one computer readable medium of claim 44 wherein the first computer program is different from the second computer program.

REMARKS

The Applicant thanks Examiner Abel Jalil for her time spent during the examiner interview and her analysis of the pending claims. Claims 29-41, 44-52, 54, 56 and 58-106 are pending in the case. Claims 29, 37, 44, 52, 54, 56, 74, 92, and 93 are amended and claims 104-106 are added as new. No new matter has been added with these amendments. The Applicant addresses the 35 U.S.C. §112 and §103 rejections below.

Interview Summary

A telephonic interview was held on May 6, 2009 between Applicant's representatives, Bruce Sunstein and Jakub Michna, and Examiner Neveen Abel Jalil. Applicant's representatives explained that the cited combination of Conrad and Turney does not disclose a computer program that displays a document before the initiation of a search, as required by claim 29. Examiner Abel Jalil suggested that the Applicant highlight the point that the first program displays the document before receipt of an execute command (for initiation of the search). Although no agreement was reached regarding the patentability of the claim, the Applicant submits with this response an amendment that implements the Examiner's suggestion.

I. The Claims Require Receipt of an Execute Command, By a Computer Program that Displays a Document, Before the Initiation of a Search.

Claim 29 is directed to a computerized method for information handling. The claim requires four processes:

First, the claim requires "analyzing, without user designation, information in a document." The document is analyzed "to identify, from the document, at least one part wherein at least a portion thereof will be used as search criteria in a subsequent search."

Second, the claim requires "retrieving the search criteria."

Third, as amended, the claim requires "displaying the document using a first computer program." While Applicant believes the original claim language required the first computer program to display the document, the claim is amended to highlight the point that the document is actually displayed and that it is the first computer program that displays the document. Thus, the scope of the claims has not been changed; only emphasis has been added.

Fourth, "on receipt … of an execute command from an input device" a search is initiated for the search criteria using a second computer program. The search is performed "in at least one of a local and a remote information source, in order to find second information associated with the search criteria." The execute command is received "by the first computer program *displaying the document*." Thus, the claim language requires that displaying the document happens *before* receipt of an execute command and initiation of the search.

Fifth, "when at least one of the local and remote information sources includes second information associated with the search criteria," the claim causes display of at least some of the second information.

II. Under a Correct Interpretation of the Claims, the Office Action's Rejection of the Claims as Indefinite is Improper.

As explained above, claim 29 requires: "on receipt, by the first computer program displaying the document, *of* an execute command from an input device, searching for the search criteria using a second computer program…" In other words, "on receipt … of an execute command from an input device," a search is initiated for search criteria using a second computer program. The prepositional phrase set off by commas simply adds that "the first computer program" receives the execute command. The office action alleges

that this limitation in claim 29 is grammatically improper and confusing. Applicant notes that the office action misquotes this claim limitation. Although when it is misquoted, the limitation may be grammatically improper and confusing, when it is reproduced correctly, as it appears in the claim, the limitation is grammatically correct and unambiguous.

Dependent claims 30, 45, and 100 add the limitation that "the first computer program is the same as the second computer program." This limitation appears in *dependent* claims and simply points out that in certain embodiments the first computer program is the same as the second computer program. Yet, in other embodiments, outside the scope of dependent claims 30, 45, and 100, but within the scope of new claims 104-106, the first computer program may be different from the second computer program. The office action asks: "why make the distinction between the first computer program and the second computer program in the first place?" As just explained, the distinction is made to clarify that the first computer program and the second computer program may be the same program in some embodiments, but different programs in other embodiments. There is nothing ambiguous about this claim language and structure.

III. The Obviousness Rejection Fails Because:

a. The Combination of Conrad and Turney Does Not Disclose a Computer Program that Displays a Document Before the Initiation of a Search; and

b. Conrad and Turney Disclose Searching For Documents Meeting User-Specified Criteria Whereas the Claims Require Starting with a Document and Finding Second Information Based on Search Criteria Identified From the Document.

The office action rejects claims 29-41, 44-52, 54, 56 and 58-103 as obvious over the combination of the Conrad patent and a newly cited publication by Turney. This combination, however, does not account for all of the limitations of the claims.

Claim 29, among other claims, requires a first computer program that displays a document before the initiation of a search. Neither Conrad nor Turney discloses these limitations, and in fact both of these references teach finding documents meeting user-specified criteria, whereas the claimed subject matter requires, on receipt of an execute command by a computer program already displaying a document, to initiate a search for search criteria already identified from analyzing the document. The office action concedes that Conrad does not specifically teach display of an actual document. Indeed, Conrad does not display a document. Figure 22 in Conrad simply shows the result of the search using user specified criteria (*i.e.*, a list of document names) and, notably, only data about the documents are displayed, not the documents themselves. *See also col. 10, lines 50-60.* Displaying *the name* of the document does not meet the requirements of claim 29, which requires displaying the document itself. Furthermore, claim 29 requires displaying the document *before* the search is performed. In direct contradistinction, Conrad's list of document names is displayed *after* a query for the documents is performed. *See abstract.* And Conrad even then, as discussed, does not display the documents, but rather, displays a list of document names.

The newly cited Turney reference does not resolve the deficiencies of Conrad. Turney is directed to a process for automatically extracting key phrases from documents. *See abstract*. Yet Turney also does not meet the limitations of claim 29 because it fails to disclose displaying a document. Instead, Turney displays an automatically generated *summary* of a document that includes highlighted key phrases. *See Section 2.2 and Figure 2*. Consequently, Turney fails to meet the limitations of claim 29 because the little it does display—namely, the summary—is displayed only *after* a query is

performed, not before the query is performed as required by the claim.

Conrad and Turney fall short for a second compelling reason. The cited prior art and the clamed method take fundamentally different approaches to searching for information. Both Conrad and Turney teach finding documents meeting user-specified criteria; they exemplify typical prior art approaches to searching. The user enters criteria, the application finds results, in this case documents meeting the criteria, and displays the result of the search. The claims pending herein require something beyond this. In particular, claim 29 requires "analyzing … information in a document to identify … search criteria …" and then, "searching for the search criteria … in order to find second information associated with the search criteria." Thus, whereas claim 29 starts *with* a document and finds second information associated with search criteria in the document, the methodology of Conrad and Turney starts *without* a document and first initiates a search for documents. Furthermore, whereas the search criteria of claim 29 are identified from the document, Conrad and Turney take the opposite approach by initiating a search for documents with *user-specified* criteria. Accordingly, because of these fundamentally different approaches and because Conrad and Turney do not disclose or suggest a program that displays a document before the initiation of a search, claim 29 is patentable over the cited prior art.

Independent claims 37, 44, 52, 54, 56, 74, 92, and 93 each require a program that displays a document before the initiation of a search, and therefore, these claims are patentable for the reason stated above with reference to claim 29. The dependent claims are also allowable for similar reasons.

Applicant believes that all of the rejections have been addressed and a notice of allowance is requested. If additional fees are required, please charge deposit account number 19-4972. To further expedite prosecution, the Examiner may call Jakub Michna at 617-443-9292 if he has any further questions.

Respectfully submitted,

/Jakub M. Michna, #61,033/

Jakub M. Michna
Attorney for Applicant
Registration No. 61,033

BROMBERG & SUNSTEIN LLP
125 Summer Street
Boston MA 02110-1618
Tel: 617 443 9292
Fax: 617 443 0004
03324/00103 1080972.2

Exhibit 6X

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Hedloy	Attorney Docket:	3324/103
Serial No.:	11/745,186	Art Unit:	2166
Filing Date:	May 7, 2007	Examiner:	Pham
Invention:	METHOD, SYSTEM AND COMPUTER READABLE MEDIUM FOR ADDRESSING HANDLING FROM AN OPERATING SYSTEM		

Response G

Dear Sir/Madam:

In response to the Office Action dated March 16, 2011, the Applicant submits the following amendment and remarks.

Amendments to the Claims are reflected in the listing of claims which begin on page 2 of this paper.

Remarks begin on page 21 of this paper.

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

What is claimed is:

Claims 1-118. (Cancelled).

119. (Currently Amended) A computer implemented method for information handling, the method comprising:

providing access to a contact database that can also be separately accessed and edited by a user and wherein the contact database includes at least three fields for storing contact information associated with each of one or more contacts, each of the at least three fields within the contact database being specific to a particular type of contact information selected from the group consisting of name, title, address, telephone number, and email address;

analyzing in a computer process information electronically displayed to identify a portion of that information as first contact information, without user designation of a specific part of the electronically displayed information to be subject to the analyzing, wherein the first contact information is at least one of a name, a title, an address, a telephone number, and an email address;

after identifying the first contact information, performing at least one action from a set of potential actions, using the first contact information previously identified as a result of the analyzing, wherein the set of potential actions includes:

~~allowing the user to command a program to perform at least one action selected from the group consisting of:~~

~~(i) displaying second information from an information source and associated with the contact information, and~~

~~(ii) initiating electronic communication using the contact information, wherein the program is capable of performing both action (i) and action (ii);~~

~~when the program performs action (i),~~ (i) initiating an electronic search ~~electronically searching~~ in ~~an information source~~ the contact database for the first contact information in order to find whether the first contact information is included in the contact database ~~information source~~;

when ~~the information source~~ a contact in the contact database includes the first contact information, if second contact information in the ~~information source~~ contact database is associated with that contact ~~information~~, electronically displaying at least a portion of the second contact information, wherein the second contact information is at least one of a name, a title, an address, a telephone number, and an email address; and

~~when the program performs action (ii),~~ (ii) initiating electronic communication using the first contact information;

wherein the computer implemented method is configured to perform both action (i) and action (ii) using the first contact information previously identified as a result of the analyzing; and

providing for the user an input device configured so that a single execute command from the input device is sufficient to cause the performing.

120. (Cancelled)

121. (Currently Amended) A method according to claim 119, wherein the computer implemented method is embodied in a client and the client is selected from a group consisting of a computer, a cell phone, a palm top device, and a personal organizer.

122. (Currently Amended) A method according to claim 121, wherein the first contact information is a name, the second contact information is an address, and the client is a computer.

123. (Currently Amended) A method according to claim 121, wherein the first contact information is a telephone number.

124. (Currently Amended) A method according to claim 121, wherein the first contact information is a telephone number, the second contact information is a name, and the client is a cell phone.

125. (Currently Amended) At least one non-transitory computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, the processes comprising:

providing access to a contact database that can also be separately accessed and edited by a user and wherein the contact database includes at least three fields for storing contact information associated with each of one or more contacts, each of the at least three fields within the contact database being specific to a particular type of contact information selected from the group consisting of name, title, address, telephone number, and email address;

analyzing in a computer process information electronically displayed to identify a portion of that information as first contact information, without user designation of a specific part of the electronically displayed information to be subject to the analyzing, wherein the first contact information is at least one of a name, a title, an address, a telephone number, and an email address;

after identifying the first contact information, performing at least one action from a set of potential actions, using the first contact information previously identified as a result of the analyzing, wherein the set of potential actions includes:

~~allowing the user to command a program to perform at least one action selected from the group consisting of:~~

~~(i) displaying second information from an information source and associated with the contact information, and~~

~~(ii) initiating electronic communication using the contact information,~~

~~wherein the program is capable of performing both action (i) and action (ii);~~

~~when the program performs action (i),~~ (i) initiating an electronic search ~~electronically searching~~ in ~~an information source~~ the contact database for the first contact information in order to find whether the first contact information is included in the contact database ~~information source~~;

when ~~the information source~~ a contact in the contact database includes the first contact information, if second contact information in the ~~information source~~ contact database is associated with that contact ~~information~~, electronically displaying at least a portion of the second contact information, wherein the second contact information is at least one of a name, a title, an address, a telephone number, and an email address; and

~~when the program performs action (ii),~~ (ii) initiating electronic communication using the first contact information;

wherein the processes are configured to perform both action (i) and action (ii) using the first contact information previously identified as a result of the analyzing; and

providing for the user an input device configured so that a single execute command from the input device is sufficient to cause the performing.

126. (Cancelled).

127. (Currently Amended) At least one non-transitory computer readable medium according to claim 125, wherein the at least one non-transitory computer readable medium

is embodied in a client[[,]] and the client selected from a group consisting of a computer, a cell phone, a palm top device, and a personal organizer.

128. (Currently Amended) At least one non-transitory computer readable medium according to claim 127, wherein the first contact information is a name, the second contact information is an address, and the client is a computer.

129. (Currently Amended) At least one non-transitory computer readable medium according to claim 127, wherein the first contact information is a telephone number.

130. (Currently Amended) At least one non-transitory computer readable medium according to claim 127, wherein the first contact information is a telephone number, the second contact information is a name, and the client is a cell phone.

131. (Currently Amended) An apparatus for information handling, the apparatus comprising:

a processor; and

a memory storing instructions executable by the processor to perform processes that include:

providing access to a contact database that can also be separately accessed and edited by a user and wherein the contact database includes at least three fields for storing contact information associated with each of one or more contacts, each of the at least three fields within the contact database being specific to a particular type of contact information selected from the group consisting of name, title, address, telephone number, and email address;

analyzing in a computer process information electronically displayed to identify a portion of that information as first contact information, without user designation of a specific part of the electronically displayed information to be subject to the analyzing, wherein the first contact information is at least one of a name, a title, an address, a telephone number, and an email address;

after identifying the first contact information, performing at least one action from a set of potential actions, using the first contact information previously identified as a result of the analyzing, wherein the set of potential actions includes:

~~allowing the user to command a program to perform at least one action selected from the group consisting of:~~

~~(i) displaying second information from an information source and associated with the contact information, and~~

~~(ii) initiating electronic communication using the contact information,~~

~~wherein the program is capable of performing both action (i) and action (ii);~~

~~when the program performs action (i),~~ (i) initiating an electronic search ~~electronically searching~~ in ~~an information source~~ the contact database for the first contact information in order to find whether the first contact information is included in the contact database ~~information source~~;

when ~~the information source~~ a contact in the contact database includes the first contact information, if second contact information in the ~~information source~~ contact database is associated with that contact ~~information~~, electronically displaying at least a portion of the second contact information, wherein the second contact information is at least one of a name, a title, an address, a telephone number, and an email address; and

~~when the program performs action (ii),~~ (ii) initiating electronic communication using the first contact information;

wherein the processes are configured to perform both action (i) and action (ii) using the first contact information previously identified as a result of the analyzing; and

providing for the user an input device configured so that a single execute command from the input device is sufficient to cause the performing.

132. (Cancelled).

133. (Previously Presented) An apparatus according to claim 131, wherein the apparatus is selected from a group consisting of a computer, a cell phone, a palm top device, and a personal organizer.

134. (Currently Amended) An apparatus according to claim 133, wherein the first contact information is a name, the second contact information is an address, and the apparatus is a computer.

135. (Currently Amended) An apparatus according to claim 133, wherein the first contact information is a telephone number.

136. (Currently Amended) An apparatus according to claim 133, wherein the first contact information is a telephone number, the second contact information is a name, and the apparatus is a cell phone.

137. (Currently Amended) A computerized method for information handling, the method comprising:

displaying information in a document electronically using a computer program;

electronically analyzing the information to identify a portion of that information as contact information including at least one of a name without an address and a name with an address;

providing an input device configured to allow a user to use the input device to command the program to perform at least one of:

(i) inserting address information from an information source and associated with the name into the document, and

(ii) storing at least part of the contact information in the information source;

wherein the program is ~~capable of performing~~ configured to perform both actions (i) and action (ii);

during the displaying, receiving an execute command from the input device, wherein accessing and manipulating the input device are the only user actions required to cause initiation and completion of the analyzing;

when the contact information is identified as including a name without an address, electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes the name, if address information in the information source is associated with the name, causing insertion of the address information into the document; and

when the contact information is identified as including a name with an address, (a) electronically prompting the user with an option to save electronically in the information source at least some of the contact information, and (b) electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes at least one contact with the name, prompting the user to make a decision whether to store the name and address as a new contact or to update one of the at least one contact.

138. (Currently Amended) At least one non-transitory computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, comprising:

displaying information in a document electronically using a computer program;

electronically analyzing the information to identify a portion of that information as contact information including at least one of a name without an address and a name with an address;

providing an input device configured to allow a user to use the input device to command the program to perform at least one of:

(i) inserting address information from an information source and associated with the name into the document, and

(ii) storing at least part of the contact information in the information source;

wherein the program is ~~capable of performing~~ configured to perform both action (i) and action (ii);

during the displaying, receiving an execute command from the input device, wherein accessing and manipulating the input device are the only user actions required to cause initiation and completion of the analyzing;

when the contact information is identified as including a name without an address, electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes the name, if address information in the information source is associated with the name, causing insertion of the address information into the document; and

when the contact information is identified as including a name with an address, (a) electronically prompting the user with an option to save electronically in the information source at least some of the contact information, and (b) electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes at least one contact with the name, prompting the user to make a decision whether to store the name and address as a new contact or to update one of the at least one contact.

139. (Currently Amended) A computerized method for information handling, the method comprising:

displaying information in a document electronically using a computer program;

electronically analyzing the information to identify a portion of that information as contact information including at least a name;

providing an input device configured to allow a user to use the input device to command the program to perform at least one action selected from the group consisting of:

(i) inserting address information from an information source and associated with the name into the document, and

(ii) storing at least part of the contact information in the information source;

wherein the program is ~~capable of performing~~ configured to perform both action (i) and action (ii);

during the displaying, receiving an execute command from the input device, wherein accessing and manipulating the input device are the only user actions required to cause initiation and completion of the analyzing;

when the program performs action (i), electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes the name, if address information in the information source is associated with the name, causing insertion of the address information into the document; and

when the program performs action (ii), electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes at least one contact with the name, prompting the user to make a decision whether to store the name as a new contact or to update one of the at least one contact.

140. (Previously Presented) A computerized method for information handling according to claim 139, further comprising:

when the program performs action (i) and the information source includes more than one address associated with the name, prompting the user to choose one of the addresses to use for insertion into the document.

141. (Currently Amended) At least one non-transitory computer readable medium encoded with instructions which when loaded on at least one computer, establish processes for information handling, comprising:

displaying information in a document electronically using a computer program;

electronically analyzing the information to identify a portion of that information as contact information including at least a name;

providing an input device configured to allow a user to use the input device to command the program to perform at least one action selected from the group consisting of:

(i) inserting address information from an information source and associated with the name into the document, and

(ii) storing at least part of the contact information in the information source;

wherein the program is ~~capable of performing~~ configured to perform both action (i) and action (ii);

during the displaying, receiving an execute command from the input device, wherein accessing and manipulating the input device are the only user actions required to cause initiation and completion of the analyzing;

when the program performs action (i), electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes the name, if address information in the information source is associated with the name, causing insertion of the address information into the document; and

when the program performs action (ii), electronically searching for the name in the information source, in order to find whether the name is included in the information source; and

when the information source includes at least one contact with the name, prompting the user to make a decision whether to store the name as a new contact or to update one of the at least one contact.

142. (Previously Presented) At least one non-transitory computer readable medium according to 141, wherein the instructions further establish processes wherein:

when the program performs action (i) and the information source includes more than one address associated with the name, prompting the user to choose one of the addresses to use for insertion into the document.

143. (New) A method according to claim 119, wherein the input device is a menu and the single execute command includes the user's selection of a menu choice from the menu.

144. (New) At least one non-transitory computer readable medium according to claim 125, wherein the input device is a menu and the single execute command includes the user's selection of a menu choice from the menu.

145. (New) An apparatus according to claim 131, wherein the input device is a menu and the single execute command includes the user's selection of a menu choice from the menu.

146. (New) A method according to claim 119, wherein the input device is a button within a window.

147. (New) At least one non-transitory computer readable medium according to claim 125, wherein the input device is a button within a window.

148. (New) An apparatus according to claim 131, wherein the input device is a button within a window.

149. (New) A method according to claim 119, wherein when the first contact information is an e-mail address, initiating electronic communication using the first contact information comprises creating an e-mail using the e-mail address.

150. (New) At least one non-transitory computer readable medium according to claim 125, wherein when the first contact information is an e-mail address, initiating electronic communication using the first contact information comprises creating an e-mail using the e-mail address.

151. (New) An apparatus according to claim 131, wherein when the first contact information is an e-mail address, initiating electronic communication using the first contact information comprises creating an e-mail using the e-mail address.

152. (New) A method according to claim 119, wherein the set of potential actions further

includes:

(iii) allowing the user to cause addition of at least some of the first contact information into the contact database.

153. (New) At least one non-transitory computer readable medium according to claim 125, wherein the set of potential actions further includes:

(iii) allowing the user to cause addition of at least some of the first contact information into the contact database.

154. (New) An apparatus according to claim 131, wherein the set of potential actions further includes:

(iii) allowing the user to cause addition of at least some of the first contact information into the contact database.

REMARKS

The Applicant thanks Examiners Pham and Alam for their analysis of the pending claims and for their time during the examiner interview on April 5, 2011, in which the Miller, Hachamovitch and Spencer references were discussed with Bruce Sunstein and Jakub Michna, Attorneys for the Applicant. Attorneys for the Applicant proposed amendments to the claims and explained to the Examiners that the claims distinguished over the Miller and the Hachamovitch references. Attorneys for the Applicant also proposed a draft declaration to swear behind the Spencer reference. Examiner Pham stated that the declaration would likely disqualify the Spencer reference as prior art and would overcome the rejections. (A signed declaration is provided herewith.) The amendments to the claims and remarks below expand on the points discussed during the interview. Reconsideration and allowance of the claims are respectfully requested.

Claims 119, 121-125, 127-131, and 133-154 are currently pending in the application. Claims 119, 121-125, 127-131, and 134-141 have been amended and claims 143-154 are new. No new matter has been added with these amendments.

Amendments and Support

In particular, independent claims 119, 125, 131, 137, 138, 139, and 141 have been amended by replacing the phrase "capable of performing" with "configured to perform." The amendment is made to clarify that the referenced methods, processes, and programs are *configured to perform* both action (i) and action (ii) after the analyzing is performed.[1] This amendment does not require that action (i) and action (ii) be performed

[1] As used herein and in the claims, the term "processes" is a systematic series of actions directed to some end.

simultaneously or one after the other, but merely emphasizes that there must be an actual ability to perform action (i) and action (ii) after the analyzing is performed.

Independent claims 119, 125, and 131 are also amended to require "after identifying the contact information, performing at least one action from a set of potential actions, using the first contact information previously identified as a result of the analyzing, wherein the set of potential actions includes:" actions "(i)" and "(ii)", which are now specifically numbered. Support for this amendment can be found in the application at, for example, Figure 1, numerals 4, 6, 10, and 12. This amendment requires that performing at least one of action (i) and action (ii) happens *after* the analyzing identifies first contact information and it also emphasizes that analyzing is distinct from performing action (i) and/or action (ii). Among other benefits, this claim limitation is important because a system that embodies the claim limitation avoids using irrelevant information for performing searching and electronic communication. Instead, information that has already been identified as contact information is used for the searching and electronic communication.

Furthermore, independent claims 119, 125, and 131 are amended to require "providing for the user an input device configured so that a single execute command from the input device is sufficient to cause the performing." Support for this amendment can be found in the application at, for example, page 6, lines 7-12; page 7, line 23 – page 8, line 2; and Figs. 1 and 2.

Additionally, independent claims 119, 125, and 131 are amended by replacing the phrase "information source" with the phrase "a contact database." As amended, the claims require "providing access to a contact database that can also be separately

accessed and edited by a user." The specification provides various examples of such contact databases (*e.g.*, "OUTLOOK" and "Symantec Act!"). Present Application at page 17, lines 2-4. *See also* Present Application at page 22, lines 1-5. The present application also explains that typically such contact databases are directly accessible and editable by the user. *See, e.g.,* Present Application at page 1, line 22 – page 2, line 10.

Furthermore, the claims require that the "contact database includes at least three fields for storing contact information associated with each of one or more contacts, each of the at least three fields within the contact database being specific to a particular type of contact information selected from the group consisting of a name, a title, an address, a telephone number, and an email address." Fig. 7 shows one example of such a contact database. Further support for this amendment can be found in the application at, for example, Figs. 10, 11, 12; page 6, lines 17-23; page 14, lines 5-8; and page 15, lines 8-10.

As amended, the claims now also require initiating a search "in order to find whether the first contact information is included *in the contact database*" and "when *a contact in the contact database* includes the first contact information, if second contact information in the contact database *is associated with that contact*, electronically displaying at least a portion of the second contact information." Support for this amendment can be found in the application at, for example, page 8, lines 20-22; page 13, lines 11-21; and Figs. 1 and 2, numerals 18, 22.

Dependent claims 121-124, 128-130, and 134-136 are amended to correct antecedent basis in light of the amendments made to independent claims 119, 125, and 131.

New claims 143-145 require that "the input device is a menu and the single execute command includes the user's selection of a menu choice from the menu." Support for these claims can be found in the application at, for example, page 6, lines 7-12 and page 22, lines 10-14.

New claims 146-148 require that the input device "is a button within a window." Support for these claims can be found in the application at, for example, Figure 3, numeral 42; the Abstract; and page 10, line 42 – page 11, line 1.

New claims 149-151 require that "when the first contact information is an e-mail address, initiating electronic communication using the first contact information comprises creating an e-mail using the e-mail address." Support for these claims can be found in the application at, for example, page 7, lines 2-5, 18-22 and page 8, lines 13-15.

New claims 152-154 require "allowing the user to cause addition of at least some of the first contact information into the contact database." Support for these claims can be found in the application at, for example, Fig. 1, numerals 28, 34, 36 and page 6, line 23 – page 7, line 2.

The pending claims structurally fall into two similar sets. Both sets require: (1) "analyzing" to identify "contact information" and (2) "searching" for that "contact information" thus identified, (3) providing an input device, and (4) performing an operation using the contact information and/or second information associated with the contact information. Moreover, the analyzing must precede the searching. The first set includes independent claims 119, 125, and 131 and the second set includes independent claims 137, 138, 139, and 141. Although the office action rejects the claims using different prior art, a rejection based on prior art must nevertheless address these recurring

themes in both sets of independent claims. The rejections of the claims are addressed below.

Independent Claims 119, 125, and 131

As amended, claim 119 (along with corresponding medium and apparatus claims 125 and 131 respectively) are directed to information handling. The claims require "providing access to a contact database that can also be separately accessed and edited by a user." Furthermore, the contact database "includes at least three fields for storing contact information associated with each of one or more contacts, each of the at least three fields within the contact database being specific to a particular type of contact information."

The claims further require "analyzing in a computer process information electronically displayed to identify a portion of that information as first contact information." After the analyzing to identify first contact information, the claims require "performing at least one action from a set of potential actions, using the first contact information previously identified as a result of the analyzing."

The first potential action includes "(i) initiating an electronic search in a contact database for the first contact information in order to find whether the first contact information is included in the contact database" and "when a contact in the contact database includes the first contact information, if second contact information in the contact database is associated with that contact, electronically displaying at least a portion of the second contact information." The second potential action includes "(ii) initiating electronic communication using the first contact information."

The claims have been amended to emphasize that performing at least one of action (i) and action (ii) happens *after* the analyzing identifies first contact information and that analyzing is distinct from performing action (i) or action (ii). Furthermore, as amended, the claims require "providing for the user an input device configured so that a single execute command from the input device is sufficient to cause the performing."

I. The Cited Prior Art references Fail to Disclose or Suggest Features Required by Independent Claims 119, 125, and 131

Independent claims 119, 125, 131 and several dependent claims are rejected as anticipated by the Miller reference. Dependent claims 122, 128, and 134 are rejected as obvious over the combination of the Miller reference and the Hachamovitch reference, while dependent claims 124, 130, and 136 are rejected as obvious over the combination of the Miller reference and a newly cited patent to Giordano (U.S. Patent No. 6,870,828). Alone or in combination, however, these references do not meet the limitations of independent claims 119, 125, and 131.

a. The Miller Reference Fails to Disclose or Suggest a Search for First Contact Information in a Contact Database, as Required by the Claims

As explained above, the claims first require "analyzing in a computer process information electronically displayed to identify a portion of that information as first contact information." Then, after the analyzing identifies a portion of the displayed information as first contact information, the claims require an electronic search for the first contact information in a contact database. There is nothing in the Miller reference that discloses or suggests a separate search in a contact database. The office action argues that the passage reproduced below discloses an electronic search for first contact information:

> Referring now to FIG. 10, a flowchart illustrating the preferred method 820 for scanning and detecting patterns in a document is shown. Method 820 starts by retrieving 1010 data to be analyzed. After the data is retrieved, several pattern analysis processes may be performed on the data. As illustrated in block 1020, [1] *a parsing process retrieves 1030 grammars, detects 1040 structures in the data based on the retrieved grammars,* and links 1050 actions associated with each grammar to each structure detected by that grammar. As illustrated in block 1060, [2] *a fast string search function retrieves 1070 the contents of string library 420, detects 1080 the strings in the data identical to those in the string library 420*, and links 1090 actions associated with the library string to the detected string. As illustrated in block 1100, additional pattern analysis processes, such as a neural net scan, can be performed 1100 to detect in the data other patterns, such as pictures, graphs, sound, etc. Method 820 then ends. Alternatively, the pattern analysis processes can be performed in parallel using a multiprocessor multitasking system, or using a uniprocessor multithreaded multitasking system where a thread is allocated to execute each pattern detection scheme.

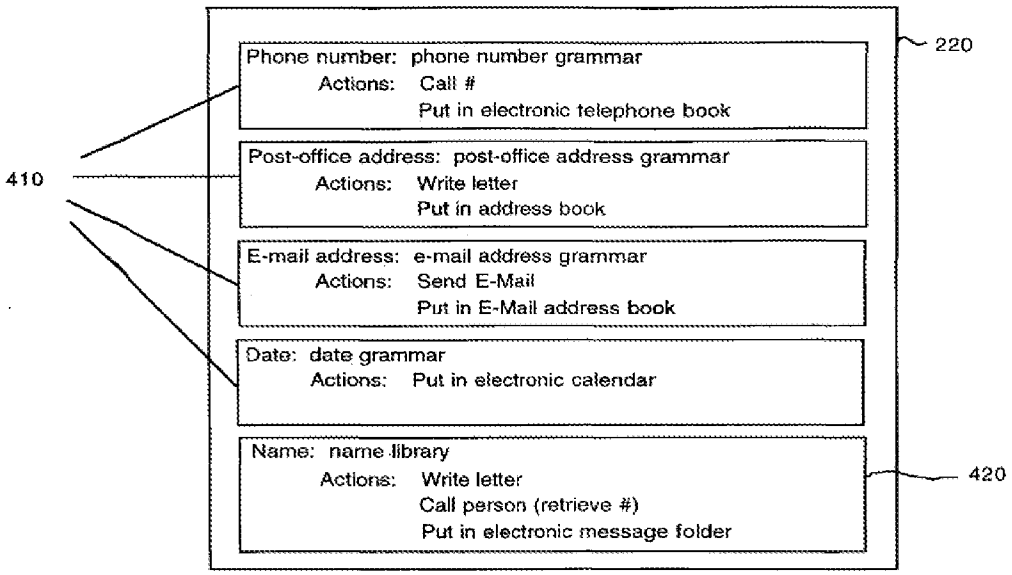
Miller, col. 6, line 34-55 (emphasis added). At most, the passage discloses two methods of detecting contact information. The first method uses grammars to detect structures within the data and the second method uses a "fast string search function" to detect strings in the data that are identical to strings in a string library.

Nowhere does the above passage disclose a search for first contact information in a contact database, which is distinct from analyzing, as required by the claims. The fast string search function of the Miller reference does not qualify as a search for first contact information in a contact database for at least two reasons. Firstly, the claims require a search for first contact information that was *already identified* by the analyzing. The Miller reference takes the opposite approach by using the fast string search function to *detect* information *for the first time.* Secondly, the string library disclosed by the Miller reference is not a contact database, let alone the specific contact database required by the claims. The claims require "a contact database that can also be separately accessed and edited by a user." Furthermore, the contact database "includes at least three fields for

storing contact information associated with each of one or more contacts, each of the at least three fields within the contact database being specific to a particular type of contact information." Such a contact database is neither disclosed nor suggested by the Miller reference and for this reason alone the claims are patentable over the Miller reference.

b. The Miller Reference Fails to Disclose or Suggest Second Contact Information, as Required by the Claims

Additionally, the claims are patentable over the Miller reference because the reference does not disclose or suggest second contact information. The claims require searching for first contact information in a contact database in order to find whether the first contact information is included in the contact database. When a contact in the contact database includes the first contact information and if *second* contact information is associated with that contact, then at least a portion of the *second* contact information is electronically displayed. The *second* contact information is "at least one of a name, a title, an address, a telephone number, and an email address." No such *second* contact information is disclosed or suggested in the Miller reference. Indeed, the concept of second contact information is entirely missing from the Miller reference. The office action argues that Figure 4 reproduced below discloses second contact information:

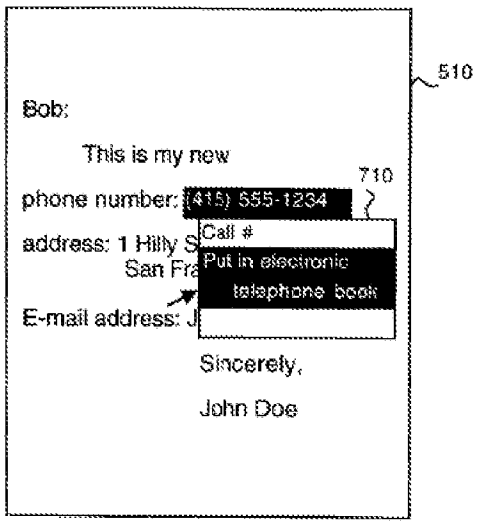


Miller, Fig. 4. Figure 4 shows grammars and strings (*e.g.*, "Phone Number", "Post-office Address", and "E-mail Address") that are associated with "actions" (*e.g.*, "Call #", "Write letter", and "Send E-Mail") in an analyzer server (220). *See* Miller, col. 5, lines 6-18. These associated "actions" do not qualify as contact information, as defined by the claims. An "action" is an activity, whereas second contact information is a type of information. The claims are even more specific and require that the second contact information is "at least one of a name, a title, an address, a telephone number, and an email address." For this additional reason, the claims are patentable over the Miller reference.

c. The Miller Reference Fails to Disclose or Suggest Electronically Displaying Second Contact Information, as Required by the Claims

The claims are also patentable over the Miller reference because the reference fails to disclose or suggest displaying second contact information. If second contact information in the contact database is associated with the contact, the claims require electronically displaying at least a portion of the second contact information. Because the

concept of second contact information is entirely missing from the Miller reference, the Miller reference cannot disclose displaying at least a portion of the second contact information. The office action argues that Figure 7 reproduced below discloses displaying second contact information:



Miller, Fig. 7. To the contrary, Figure 7 shows a pop-up menu that displays "actions" linked to a grammar. More particularly, the Figure shows the actions "Call #" and "Put in electronic telephone book" as linked to the telephone number "(415) 555-1234." Miller, col. 5, lines 38-50. As explained in Section (b) above, these actions do not constitute second contact information. The only contact information shown in Figure 7 is the telephone number "(415) 555-1234," but this number also does not qualify as *second* contact information because this number is identified within the document.[2] Whereas the claims require the second contact information to be located within the contact database: "if second contact information *in the contact database* is associated with that contact,

[2] At most, the telephone number might qualify as "first contact information" according to the claims.

electronically displaying at least a portion of the second contact information." For this supplementary reason, the claims are patentable over the Miller reference.

d. The Hachamovitch Reference and Giordano Reference Do Not Alleviate the Deficiencies of the Miller Reference

Among other things, the Hachamovitch reference fails to disclose or suggest the contact database required by the claims. The Hachamovitch reference is directed to an auto-completion system. As the user types a data entry into a document, the system searches for possible entry completions corresponding to the partial data entry in a word completion list. Hachamovitch, Abstract. The word completion suggestion list includes a "name" field and a "completion" field. Hachamovitch, Fig. 3. This listing, however, does not constitute a contact database, as required by the claims. In particular, the Hachamovitch reference does not disclose a contact database that "includes at least three fields for storing contact information associated with each of one or more contacts, each of the at least three fields within the contact database being specific to a particular type of contact information." The Hachamovitch reference fails to disclose or suggest the contact database required by the claims.

The Hachamovitch reference also suffers from other deficiencies. In particular, the Hachamovitch reference fails to teach the two distinct processes of the claims: (1) "analyzing" to identify "first contact information" and (2) then "searching" for that "first contact information" thus identified.[3] These arguments were presented in the Applicant's Response E of August 10, 2010 and the arguments in that response apply equally to

[3] The term "distinct processes" does not require separate computer programs, tasks, run files, etc. for each process, but simply that the identification of first contact information happens separately from (and before) the search or initiation of the electronic communication.

independent claims 119, 125, and 131. That response is attached hereto and the arguments therein are incorporated herein by reference as Exhibit A.

The Giordano reference also fails to disclose or suggest a search for first contact information in a contact database, as required by the claims. The Giordano reference is directed to a method that iconifies a telephone number appearing in a web page. Giordano, Abstract. In one embodiment, a user can store the iconified number in an address book. Giordano, col. 4, lines 57-67. Yet, there is nothing in the reference that discloses or suggests a *search* in the contact database for the first contact information, as required by the claims. Furthermore, there is nothing in the reference that discloses or suggests second contact information and displaying second contact information, as further required by the claims.

For at least these reasons, independent claims 119, 125, and 131 are patentable over the Miller, Hachamovitch, and Giordano references, either alone or in combination. Dependent claims 121-124, 127-130, 133-136, and 143-154 are patentable for similar reasons.

Independent Claims 137, 138, 139, and 141

Independent claims 137, 138, 139, and 141 are rejected as obvious in view of the Hachamovitch reference and a newly cited patent to Spencer (U.S. Patent No. 6,349,299).

The Hachamovitch reference fails to teach two distinct processes of the claims 137, 138, 139, and 141. In particular, the Hachamovitch reference fails to disclose or suggest (1) "analyzing" to identify "contact information" and (2) then "searching" for that "contact information" thus identified. These arguments were presented in the Applicant's Response E of August 10, 2010 (attached hereto as Exhibit A) and the

arguments in that response apply equally to independent claims 137, 138, 139, and 141. Those arguments are incorporated herein by reference.

Nonetheless, to advance prosecution and to expedite issuance of a notice of allowance, the Applicant submits herewith a Declaration Under C.F.R. 1.131 to remove the Spencer patent as a prior art reference. The Spencer patent claims priority to a provisional application filed on December 24, 1998. The Applicant completed his invention before this date and, therefore, the Spencer patent is not prior art and cannot be the basis of a rejection.

For at least these reasons, independent claims 137, 138, 139, and 141 are patentable over the Hachamovitch and Spencer references. Dependent claims 140 and 142 are patentable for similar reasons.

Applicant believes that all of the rejections have been addressed and a notice of allowance is respectfully solicited. If any fees are required for consideration of this amendment, please charge account number 19-4972. To further expedite prosecution, the Examiner may call Bruce Sunstein or Jakub Michna at 617-443-9292 if he has any further questions.

Respectfully submitted,

/Bruce D. Sunstein, #27,234/

Bruce D. Sunstein
Registration No. 27,234

/Jakub M. Michna, #61,033/

Jakub M. Michna
Registration No. 61,033

Attorneys for Applicant

SUNSTEIN KANN MURPHY
& TIMBERS LLP
125 Summer Street
Boston MA 02110-1618
Tel: 617 443 9292
Fax: 617 443 0004
03324/00103 1425912.1

Exhibit 6Y

Practitioner's Docket No. 3324/102 ***PATENT***

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Atle Hedloy
Application No.: 12/182,048 Group No.: 2176
Filed: 07/29/2008 Examiner: Tran, Quoc A.
For: Method, System and Computer Readable Medium for Addressing Handling from a Computer Program

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

AMENDMENT TRANSMITTAL

1. Transmitted herewith is an amendment for this application.

STATUS

2. Applicant is a small entity.

EXTENSION OF TERM

3. The proceedings herein are for a patent application and the provisions of 37 C.F.R. 1.136 apply. Applicant believes that no extension of term is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition for extension of time.

FEE FOR CLAIMS

4. The fee for claims (37 C.F.R. 1.16(b)-(d)) has been calculated as shown below:

	(Col. 1)		(Col. 2)		(Col. 3)		SMALL ENTITY			
	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NO. PREVIOUSLY PAID FOR		PRESENT EXTRA		RATE		ADDIT. FEE	
TOTAL	22	–	20	=	2	x	$ 26.00	=	$	52.00
INDEP.	2	–	3	=	0	x	$ 110.00	=	$	0.00
FIRST PRESENTATION OF MULTIPLE DEP. CLAIM						+	$ 0.00	=	$	0.00
							TOTAL ADDIT. FEE		$	52.00

Total additional fee for claims required $52.00

REPLACEMENT DRAWINGS

5. Attached are 14 sheets of replacement drawings.

TERMINAL DISCLAIMER

6. Attached is a terminal disclaimer.

Fee: $70.00

INFORMATION DISCLOSURE STATEMENT

7. Attached is a supplemental information disclosure statement.

Fee: $180.00

TOTAL FEES DUE

8. The total fees due are:

Fees for Additional Claims	$ 52.00
Information Disclosure Statement	$180.00
Terminal Disclaimer	$ 70.00

Total Fees Due: $302.00

FEE PAYMENT

9. Authorization is hereby made to charge the amount of $302.00 to Deposit Account No. 19-4972.

Charge any additional fees required by this paper or credit any overpayment in the manner authorized above.

Date: December 8, 2010

/Jakub M. Michna, #61,033/

Jakub M. Michna
Registration No. 61,033
SUNSTEIN KANN MURPHY & TIMBERS LLP
125 Summer Street
Boston, MA 02110-1618
US
617-443-9292
Customer No. 02101

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Hedloy	Attorney Docket:	3324/102
Serial No.:	12/182,048	Art Unit:	2176
Filing Date:	July 29, 2008	Examiner:	Tran
Invention:	METHOD, SYSTEM AND COMPUTER READABLE MEDIUM FOR ADDRESSING HANDLING FROM A COMPUTER PROGRAM		

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Response A

Dear Sir/Madam:

In response to the Office Action dated October 28, 2010, the Applicant submits the following amendment and remarks.

Amendments to the Specification begin on page 2 of this paper.

Amendments to the Drawings begin on page 6 of this paper.

Amendments to the Claims are reflected in the listing of claims which begin on page 7 of this paper.

Remarks begin on page 13 of this paper.

AMENDMENTS TO THE SPECIFICATION

The amendments to the specification indicated below are solely to correct obvious typographical errors.

Please replace the paragraph beginning on page 3, line 7 of the originally filed application with the following amended paragraph:

The above and other objects are achieved according to the present invention by providing a novel method, system and computer readable medium for providing a function item, such as a key, button, icon, or menu, tied to a user operation in a computer, whereby a single click on the function item in a window or program on a computer screen, or one single selection in a menu in a program, initiates retrieval of name and addresses and/or other person or company related information, [[w]] while the user works simultaneously in another program, e.g., a word processor. The click on the function item initiates a program connected to the button to search a database or file available on or through the computer, containing the person, company or address related data, in order to look up data corresponding to what the user types, or partly typed, e.g., name and/or address in the word processor, the correct data from the database, data related to the typed data, e.g., the name of the person, company, or the traditional or electronic address, or other person, or company, or address related data, and alternatively the persons, companies, or addresses, are displayed and possibly entered into the word processor, if such related data exists.

Please replace the paragraph beginning on page 7, line 8 of the originally filed application with the following amended paragraph:

In FIG. 1, after the user has inserted the address in the word processor, the user commands the button at step 2 and the program analyzes what the user has typed in the document at step 4. [[AT]] At step 6, the program decides what was found in the document and if the program found nothing in the document or what it found was un-interpretable the program goes to step 8 and outputs an appropriate message to the user and then quits at step 16. The program analyzes what the user has typed in the document at step 4, for example, by analyzing (i) paragraph/line separations/formatting, etc.; (ii) street, avenue, drive, lane, boulevard, city, state, zip code, country designators and abbreviations, etc.; (iii) Mr., Mrs., Sir, Madam, Jr., Sr. designators and abbreviations, etc.; (iv) Inc., Ltd., P.C., L.L.C, designators and abbreviations, etc.; and (v) a database of common male/female names, etc.

Please replace the paragraph beginning on page 7, line 19 of the originally filed application with the following amended paragraph:

If the program find an e-mail address mailing list/category name telephone number or other information, at step 10 an appropriate action is performed by the program and then the program execution quits at step 16. If the program only finds a name or initials, or the like, the program looks up the name in the database at step 12 and at step 18 the program determines what was found. If the program finds more than one possible contact/address match, at step 20 the program displays menu choices to the user to let him choose an appropriate answer. Then at step 22 the program inserts a correct address and name in the document and then at step 16 the program quits execution. If the program finds one match exactly, i.e., one contact with one address, the program inserts the correct address and name in the document at step 22 then quits and then quits execution at step 16. If the program does not find a name in the database, at step 24 the program prompts the user to specify an address and then quits execution at step 16. If the program at step 6 finds a name and an address, at step 14 the name is looked up in the database. Then, at step 26, if no match is found, at step 28 the program inserts an address and a name which are possibly corrected by the user into the database and then quits execution at step 16. If at step 26, the name and address is found, at step 32 the program either takes no action or displays ~~t he~~ the data for the user to edit. If at step 26, the name is found but not the address, the program prompts the user for a decision at step 30. If the user decides that this is another contact with a same name, the program goes to step 28. If the user decides that this is a one time occurrence, no action is taken and the program quits at step 16. If the user decides that the contact has, for example, moved and that this is a new address, at step 34 one of the old addresses for the contact is replaced with the new one and the program ~~with the new one and the program~~ quits at step 16. If the user decides that this is an additional address for the contact, at step 36 the additional address is inserted into the database for that contact and execution quits at step 16.

Please replace the paragraph beginning on page 11, line 21 of the originally filed application with the following amended paragraph:

FIG. 3 illustrates a starting point in word processor document, such as WORD document, wherein the user has typed a name of a contact 40. The user commands the button 42, for example, marked "OneButton," and the program according to the present invention retrieves the name 40 from the document, searches a database for the name of the contact 40 and generates a screen as shown in, for example, FIG. 8. This screen includes a message 68 informing the user that the contact does not exist in the database and to specify an address, and "OK" buttons 56. At this point when the user commands the OK button 56, the user returns to the document so that [[he]] the contact's address can be included as in Example 2 above.

Please replace the paragraph beginning on page 13, line 21 of the originally filed application with the following amended paragraph:

At this point the user may command the Choose button 86 to use the selected address and return to the document, or the user may command the More >>> button 90 to view how the program interpreted what [[he]] the user typed in the word processor, and possibly change this data, wherein the program generates an updated screen as shown in, for example, FIG. 11. ~~The updated screen includes the data 62 which displays the name for example, FIG. 11.~~ The updated screen includes the data 62 which displays the name typed in the word processor as interpreted by the program, address fields, and the fields for the address type selection 54, such as home, business, etc., which may be changed by the user before the program stores it in the database, the Add and Choose button 64, a "<<<Less" button 90 corresponding to the More>>> button 90 for returning to the screen of FIG. 10, and an "Add this address to the selected contact above" button 92. The user might then command the Add this address to the selected contact above button 92 and the result in the word processor is illustrated in FIG. 4. The user can also cancel the operations by commanding the Cancel button 60, or command the add choose button 64 to add this name and address as a new contact and address, or open the database before storing data into the database by commanding a "Full details" button 88 as will be later described.

AMENDMENTS TO THE DRAWINGS

Please replace the 17 sheets of original drawings presently of record with the 14 sheets of replacement drawings presented with this response.

The 14 sheets of replacement drawings presented with this response correct grammatical mistakes in the original drawings:

- The spelling of the word "possibly" in item 28 of Fig 1B.
- The spelling of the word "possibly" in item 28 of Fig 2B.
- The spelling of the word "address" in item 34 of Fig 2B.

The 14 sheets of replacement drawings presented with this response also incorporate the subject matter of Figs. 1a and 1b into a single sheet labeled Fig.1. Similarly, the subject matter of Figs. 2a and 2b is incorporated into a single sheet labeled Fig. 2. No other changes have been made. Applicant believes that this amendment provides for easier reading of Figs. 1a, 1b, 2a, and 2b. No new matter has been added with the replacement drawings.

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

What is claimed is:

1. (Currently Amended) A computer-implemented method for finding data related to the contents of a document using a first computer program running on a computer, the method comprising:

displaying the document electronically using the first computer program;

while the document is being displayed, analyzing, in a computer process, ~~the document to identify any~~ first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information;

retrieving the first information ~~thus identified~~;

providing an input device, configured by the first computer program, that allows a user to enter a user command to initiate an operation, the operation comprising (i) performing a search using at least part of the first information as a search term in order to find the second information, of a specific type or types, associated in an information source with the search term, wherein the specific type or types of second information is dependent at least in part on the type or types of the first information, and (ii) performing an action using at least part of the second information;

in consequence of ~~following~~ receipt by the first computer program of ~~a~~ the user command from the input device, causing a search ~~searching~~ for the ~~first information~~ search term in the information source, using a second computer program, in order to find second information related to the search term ~~first information~~; and

if searching finds any second information related to the search term ~~first information~~, performing the action ~~an operation~~ using at least part of the second information, wherein the action is of a type depending at least in part on the type or types of the first information.

2. (Original) A method according to claim 1, wherein the first information comprises at least one of name-, person-, company- and address-related information.

3. (Currently Amended) A method according to claim 1, wherein performing the action ~~operation~~ includes performing ~~an operation~~ the action in the first computer program.

4. (Currently Amended)A method according to claim 2, wherein performing the action ~~operation~~ includes performing the action ~~an operation~~ in the first computer program.

5. (Currently Amended) A method according to claim 3, wherein performing the action ~~operation~~ includes causing addition of at least part of ~~adding~~ the second information to the first information in the document.

6. (Currently Amended) A method according to claim 3, wherein performing the action ~~operation~~ includes causing display of at least part of ~~displaying~~ the second information.

7. (Currently Amended) A method according to claim 3, wherein performing the action ~~operation~~ includes causing display of at least part of ~~displaying~~ the second information by the first computer program.

8. (Currently Amended) A method according to claim 1, further comprising, providing a prompt for updating the information source ~~application~~ to include the first information.

9. (Currently Amended) A method according to claim 1, further comprising, if the search is not successful, providing a prompt for updating ~~a database~~ the information source ~~associated with the second computer program~~ to include the first information.

10. (Currently Amended) A method according to claim 1, wherein receipt by the first computer program of the user command precedes analyzing the document.

11. (Original) A method according to claim 1, wherein analyzing the document is completed after the receipt of the user command is completed and before searching is initiated.

12. (Currently Amended) A method according to claim 1, wherein the input device is a ~~further comprising providing~~ a graphical input device ~~for entry of the user command~~.

13. (Currently Amended) A method according to claim 1, wherein the user command is the only command from a user necessary as a condition to initiate ~~initiating~~ performing the operation.

14. (Currently Amended) A method according to claim 1, ~~further comprising providing a graphical input device for entry of the user command,~~ wherein the input device is a menu~~,~~ and

the entry of the user command includes a user's selection of the menu and click on a menu choice from the menu.

15. (Currently Amended) A method according to claim 1, further comprising, if searching results in a plurality of distinct instances of second information, displaying such instances to enable user selection of one of them for use in performing the action ~~operation~~.

16. (Currently Amended) A method according to claim 1, wherein the information source ~~a database~~ is associated with the second computer program and is available on the computer.

17. (Currently Amended) A method according to claim 1, wherein the information source ~~a database~~ is associated with the second computer program and is available through the computer.

18. (Currently Amended) A computer-implemented method for finding data related to the contents of a document using a first computer program running on a computer, the method comprising:

displaying the document electronically using the first computer program;

while the document is being displayed, analyzing, in a computer process on the computer, ~~the document to identify any~~ first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information, and wherein the first information comprises at least one of name-, person-, company-, and address-related information;

providing an input device, configured by the first computer program, that allows a user to

enter a user command to initiate an operation, the operation comprising (i) performing a search using at least part of the first information as a search term in order to find the second information, of a specific type or types, associated with the search term in a user editable information source outside the document, wherein the specific type or types of second information is dependent at least in part on the type or types of the first information, and (ii) performing an action using at least part of the second information ~~for entry of a user command~~, wherein the input device includes a menu;[[,]]

retrieving the first information ~~thus identified~~;

in consequence of ~~following~~ receipt by the first computer program of the user command, such user command including a user's selection of the menu and click on a menu choice from the menu, causing a search ~~searching~~ for the search term in the information source ~~first information~~, using a second computer program, in order to find second information related to the search term ~~first information~~ in ~~a~~ the user editable information source outside the document; and

if searching finds any second information related to the search term ~~first information~~, performing the action ~~an operation~~ using at least part of the second information, wherein the action is of a type depending at least in part on the type or types of the first information and performing the action ~~operation~~ includes at least causing display of ~~displaying~~ at least part of the second information.

19. (Currently Amended) A method according to claim 18, further comprising, if searching results in a plurality of occurrences of second information, causing display of ~~displaying~~ such instances to enable user selection of one of them for use in performing the action ~~operation~~.

20. (Currently Amended) A method according to claim 18, wherein performing the action ~~operation~~ includes causing addition of at least part of ~~adding~~ the second information to the first information in the document.

21. (New) A method according to claim 1, wherein performing the action includes causing insertion of at least part of the second information into the document.

22. (New) A method according to claim 1, wherein performing the action includes causing insertion of at least part of the second information into the document by the first computer program.

REMARKS

The Applicant thanks Examiner Tran for his analysis of the pending claims and for his time during the examiner interview on November 24, 2010, in which the Tso, Person, and Miller references were discussed with Bruce Sunstein and Jakub Michna, attorneys for the Applicant, and Atle Hedloy, the inventor and Applicant herein. The Examiner agreed that the claims, as amended, distinguish the cited prior art. Examiner Tran also asked the Applicant to further clarify the analyzing and providing processes of the independent claims. The amendments to the claims and the remarks below address the Examiner's concerns expressed during the interview.

Claims 1-22 are currently pending in the application. Claims 21 and 22 are new. Claims 1 and 3-10, 12-20 are currently amended. No new matter has been added. The rejections and objections to the claims are addressed below.

Objections

The office action objects to claims 16 and 17 as duplicate claims. Claim 16, however, is not a duplicate of claim 17. Claim 16 requires an information source that is "available *on* the computer", while claim 17 requires an information source that is "available *through* the computer."

The Applicant files with this response a corrected information disclosure statement (IDS), relisting those references that the Examiner did not consider in the previous IDS of January 26, 2009 and also listing new references. Section 3 of the IDS filed herewith lists issued and pending applications that are related to the present application. Applicant notes that Application Serial No. 12/841,302 (also before the Examiner) and the prior art references analyzed in the Accelerated Examination Support

Document (AESD) of July 22, 2010 are of particular interest in relation to the present application.

Non-Statutory Double Patenting

A terminal disclaimer is filed with this response to overcome the non-statutory double patenting rejection.

Amendments to the Claims

Claim 1 is directed to a computer-implemented method for finding data related to the contents of a document using a first computer program. The claim has been amended to require analyzing first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information. Support for this amendment can be found in the present application at, for example, page 7, line 8 to page 8, line 8. The claim also requires providing an input device to allow a user to enter a user command to initiate an operation. Support for such an input device can be found at, for example, page 1, lines 17-22, page 3, lines 7-13, and page 5, lines 18-22 of the application. According to the claim, the operation comprises:

(i) "performing a search using at least part of the first information as a search term in order to find the second information, of a specific type or types, associated in an information source with the search term, …, wherein the specific type or types of second information is dependent at least in part on the type or types of the first information"

(ii) "performing an action using at least part of the second information."

As amended, the claim requires that the action is of a type depending at least in part on the type or types of the first information. Support for these amendments can be found in

the Figures of the present application at, for example, Figs. 1a, 1b, 2a, 2b and at items 6, 8, 10, 12, 14, 26, 27, 28, 29, 30, 31, 32, 34, 36. Support can also be found in the present application at, for example, page 7, line 10 – page 8, line 20. No order is intended in recitation of the claim limitations except as may be otherwise required by the claims.

Applicant notes that the current amendments to the claims have not been made for reasons of patentability. Applicant believes that the original claims were patentable over the cited prior art at least because none of the cited references discloses "analyzing a document to identify any first information", as required by the original claims.

Accordingly, Applicant now amends the claims, not to overcome the cited prior art, but instead to provide more context and clarity to the claims. In fact, the limitation described in the previous paragraph has been amended out of the claims, which, in that respect, broadens the claims. On the other hand, the amendments have narrowed the claims in other respects. Yet these narrowing amendments are merely intended to streamline and facilitate prosecution of the claims and are not made to overcome the cited prior art.

Distinction of the Claimed Subject Matter from the Art of Record

Claims 1-20 are rejected as obvious in view of the combination of the Tso reference (US Patent No. 6,085,201) and the Person reference ("Special Edition Using Microsoft Word 97"). This combination, however, does not meet the limitations of the amended claims for at least the reasons provided below.

As amended, claim 1 requires:

1. "analyzing first information from the document to determine if the first information is at least one of a plurality of types of information that can be

searched for in order to find second information related to the first information"

2. "performing a search … in order to find the second information, of a specific type or types, … wherein the specific type or types of second information is dependent at least in part on the type or types of the first information"
3. "performing the action using at least part of the second information, wherein the action is of a type depending at least in part on the type or types of the first information."

None of the cited prior art references discloses all three of these limitations, which are powerful in distinguishing most, if not all, of the art of record.

For example, with respect to the present rejection against the claims, none of the cited prior art references discloses item (1) above, "analyzing first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information."

Tso discloses composing an e-mail message with the help of a context-sensitive template engine. Tso, Abstract. When the template engine receives an input text string, Tso discloses "decomposing" the text string into individual words and comparing the individual words to keywords associated with predetermined templates. Tso, col. 4, lines 32-35, col. 5, lines 7-17. The template with the best match between keywords and individual words is chosen as the most appropriate template and that template is presented to the user. Tso, col. 5, lines 7-17.

Tso does not analyze first information "from the document to *determine if the first information is at least one of a plurality of types* of information that can be searched for in order to find second information related to the first information", as required by claim 1. This is so because Tso decomposes and compares every word in the input text string irrespective of the type of information that is within the text string.

The Person reference does not make up for the deficiencies of the Tso reference. The Person reference describes a functionality in Microsoft Word 97 known as Mail Merge. Mail Merge allows a user to create multiple letters "by merging together a list of names and addresses with letters, envelopes, or address labels." Person, page 485. The method starts when the user selects a main document and inserts merge fields into the document for specifying the placement of merged information. The merge fields also specify a particular data field that is to replace them: "First_Name", "Last_Name", and "Company_Name." *See* Person, Fig. 17.30; Person, page 485. Next, the user selects a list of contacts that includes data fields for name and address information. Person, page 488. When the user selects the "merge" button, the name and address information from the list of contacts is retrieved and inserted into the main document according to the merge fields. Person, page 492, 507.

The Person reference also does not disclose or suggest item (1) above, analyzing first information from the document *to determine if the first information is at least one of a plurality of types of information* that can be *searched* for in order to find second information related to the first information, as required by the claims. Instead, as explained above, the Person reference merely discloses inserting data retrieved from a list of contacts *en masse* into merge fields within the document. Person fails to disclose or

suggest analyzing first information in the document, as required by the claims, because Person requires the merge fields in the document to already have been specified by the user as merge fields and as related to a specific data fields in the list of contacts (*e.g.*, "First_Name", "Last_Name", and "Company_Name"). Thus, there is no need to analyze text to determine its type.

Additionally, Person fails to disclose or suggest a search for first information, as required by item (2) above, because the information to be inserted by mail merger is never searched for in the information source. Rather, the records specified by the user (*i.e.*, *every* contact from the contact list) are retrieved and used for insertion into the main document. Since there is no search, as required by the claims, Person also cannot determine if the first information is of a type *that can be searched*, as further required by item (1) above. Accordingly, neither the Tso reference nor the Person reference discloses or suggests analyzing first information from the document to determine if the first information is of a type of information that can be searched for in order to find second information related to the first information, as required by claim 1. And, for this reason, among others, the claims are patentable over the combination of the Tso and Person references.

Independent claim 18 requires limitations that are similar to claim 1 and, therefore, claim 18 is patentable over the cited prior art for similar reasons. The dependent claims are also allowable for similar reasons.

The presently amended claims also distinguish the Miller reference (U.S. 5,946, 647), which has been used in a rejection against the claims in Application Serial No. 12/841,302 (also before the Examiner). Miller discloses a method for detecting data in a

document and performing a particular action on the detected data. Miller, Abstract. Miller does not disclose "performing a search … in order to find … second information, of a specific type or types, associated in an information source", by item (2) above. In Miller, no search is performed at all. Furthermore, since Miller does not search for second information, it also does not disclose "performing the action *using at least part of the second information*", as required by item (3) above. For at least these reasons, all of the present claims distinguish the Miller reference.

Applicant believes that all of the rejections have been addressed and a notice of allowance is respectfully solicited. If any fees are required, please charge deposit account number 19-4972. To further expedite prosecution, the Examiner may call Bruce Sunstein or Jakub Michna at 617-443-9292 if he has any further questions.

Respectfully submitted,

/Bruce D. Sunstein, #27,234/

Bruce D. Sunstein
Registration No. 27,234

/Jakub M. Michna, #61,033/

Jakub M. Michna
Registration No. 61,033

Attorneys for Applicant

SUNSTEIN KANN MURPHY
& TIMBERS LLP
125 Summer Street
Boston MA 02110-1618
Tel: 617 443 9292
Fax: 617 443 0004
03324/00102 1371465.1

Exhibit 6Z

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Hedloy	Attorney Docket:	3324/102
Serial No.:	12/182,048	Art Unit:	2176
Filing Date:	July 29, 2008	Examiner:	Tran
Invention:	METHOD, SYSTEM AND COMPUTER READABLE MEDIUM FOR ADDRESSING HANDLING FROM A COMPUTER PROGRAM		

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Supplemental Response A

Dear Sir/Madam:

In response to the Office Action dated October 28, 2010, the Applicant submits the following amendment and remarks:

Amendments to the Specification begin on page 2 of this paper.

Amendments to the Drawings begin on page 6 of this paper.

Amendments to the Claims are reflected in the listing of claims which begin on page 7 of this paper.

Remarks begin on page 19 of this paper.

This Supplemental Response adds new claims 23-44, which are computer readable medium claims that mirror claims 1-22. Otherwise, this Supplemental Response is substantively identical to the Response filed by Applicant on December 8, 2010.

AMENDMENTS TO THE SPECIFICATION

The amendments to the specification indicated below are solely to correct obvious typographical errors.

Please replace the paragraph beginning on page 3, line 7 of the originally filed application with the following amended paragraph:

The above and other objects are achieved according to the present invention by providing a novel method, system and computer readable medium for providing a function item, such as a key, button, icon, or menu, tied to a user operation in a computer, whereby a single click on the function item in a window or program on a computer screen, or one single selection in a menu in a program, initiates retrieval of name and addresses and/or other person or company related information, [[w]] while the user works simultaneously in another program, e.g., a word processor. The click on the function item initiates a program connected to the button to search a database or file available on or through the computer, containing the person, company or address related data, in order to look up data corresponding to what the user types, or partly typed, e.g., name and/or address in the word processor, the correct data from the database, data related to the typed data, e.g., the name of the person, company, or the traditional or electronic address, or other person, or company, or address related data, and alternatively the persons, companies, or addresses, are displayed and possibly entered into the word processor, if such related data exists.

Please replace the paragraph beginning on page 7, line 8 of the originally filed application with the following amended paragraph:

In FIG. 1, after the user has inserted the address in the word processor, the user commands the button at step 2 and the program analyzes what the user has typed in the document at step 4. [[AT]] At step 6, the program decides what was found in the document and if the program found nothing in the document or what it found was un-interpretable the program goes to step 8 and outputs an appropriate message to the user and then quits at step 16. The program analyzes what the user has typed in the document at step 4, for example, by analyzing (i) paragraph/line separations/formatting, etc.; (ii) street, avenue, drive, lane, boulevard, city, state, zip code, country designators and abbreviations, etc.; (iii) Mr., Mrs., Sir, Madam, Jr., Sr. designators and abbreviations, etc.; (iv) Inc., Ltd., P.C., L.L.C, designators and abbreviations, etc.; and (v) a database of common male/female names, etc.

Please replace the paragraph beginning on page 7, line 19 of the originally filed application with the following amended paragraph:

If the program find an e-mail address mailing list/category name telephone number or other information, at step 10 an appropriate action is performed by the program and then the program execution quits at step 16. If the program only finds a name or initials, or the like, the program looks up the name in the database at step 12 and at step 18 the program determines what was found. If the program finds more than one possible contact/address match, at step 20 the program displays menu choices to the user to let him choose an appropriate answer. Then at step 22 the program inserts a correct address and name in the document and then at step 16 the program quits execution. If the program finds one match exactly, i.e., one contact with one address, the program inserts the correct address and name in the document at step 22 then quits and then quits execution at step 16. If the program does not find a name in the database, at step 24 the program prompts the user to specify an address and then quits execution at step 16. If the program at step 6 finds a name and an address, at step 14 the name is looked up in the database. Then, at step 26, if no match is found, at step 28 the program inserts an address and a name which are possibly corrected by the user into the database and then quits execution at step 16. If at step 26, the name and address is found, at step 32 the program either takes no action or displays ~~t he~~ the data for the user to edit. If at step 26, the name is found but not the address, the program prompts the user for a decision at step 30. If the user decides that this is another contact with a same name, the program goes to step 28. If the user decides that this is a one time occurrence, no action is taken and the program quits at step 16. If the user decides that the contact has, for example, moved and that this is a new address, at step 34 one of the old addresses for the contact is replaced with the new one and the program ~~with the new one and the program~~ quits at step 16. If the user decides that this is an additional address for the contact, at step 36 the additional address is inserted into the database for that contact and execution quits at step 16.

Please replace the paragraph beginning on page 11, line 21 of the originally filed application with the following amended paragraph:

FIG. 3 illustrates a starting point in word processor document, such as WORD document, wherein the user has typed a name of a contact 40. The user commands the button 42, for example, marked "OneButton," and the program according to the present invention retrieves the name 40 from the document, searches a database for the name of the contact 40 and generates a screen as shown in, for example, FIG. 8. This screen includes a message 68 informing the user that the contact does not exist in the database and to specify an address, and "OK" buttons 56. At this point when the user commands the OK button 56, the user returns to the document so that [[he]] the contact's address can be included as in Example 2 above.

Please replace the paragraph beginning on page 13, line 21 of the originally filed application with the following amended paragraph:

At this point the user may command the Choose button 86 to use the selected address and return to the document, or the user may command the More >>> button 90 to view how the program interpreted what [[he]] the user typed in the word processor, and possibly change this data, wherein the program generates an updated screen as shown in, for example, FIG. 11. ~~The updated screen includes the data 62 which displays the name for example, FIG. 11.~~ The updated screen includes the data 62 which displays the name typed in the word processor as interpreted by the program, address fields, and the fields for the address type selection 54, such as home, business, etc., which may be changed by the user before the program stores it in the database, the Add and Choose button 64, a "<<<Less" button 90 corresponding to the More>>> button 90 for returning to the screen of FIG. 10, and an "Add this address to the selected contact above" button 92. The user might then command the Add this address to the selected contact above button 92 and the result in the word processor is illustrated in FIG. 4. The user can also cancel the operations by commanding the Cancel button 60, or command the add choose button 64 to add

this name and address as a new contact and address, or open the database before storing data into the database by commanding a "Full details" button 88 as will be later described.

AMENDMENTS TO THE DRAWINGS

Please replace the 17 sheets of original drawings presently of record with the 14 sheets of replacement drawings presented with this response.

The 14 sheets of replacement drawings presented with this response correct grammatical mistakes in the original drawings:

- The spelling of the word "possibly" in item 28 of Fig 1B.
- The spelling of the word "possibly" in item 28 of Fig 2B.
- The spelling of the word "address" in item 34 of Fig 2B.

The 14 sheets of replacement drawings presented with this response also incorporate the subject matter of Figs. 1a and 1b into a single sheet labeled Fig.1. Similarly, the subject matter of Figs. 2a and 2b is incorporated into a single sheet labeled Fig. 2. No other changes have been made. Applicant believes that this amendment provides for easier reading of Figs. 1a, 1b, 2a, and 2b. No new matter has been added with the replacement drawings.

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

What is claimed is:

1. (Currently Amended) A computer-implemented method for finding data related to the contents of a document using a first computer program running on a computer, the method comprising:

displaying the document electronically using the first computer program;

while the document is being displayed, analyzing, in a computer process, ~~the document to identify any~~ first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information;

retrieving the first information ~~thus identified~~;

providing an input device, configured by the first computer program, that allows a user to enter a user command to initiate an operation, the operation comprising (i) performing a search using at least part of the first information as a search term in order to find the second information, of a specific type or types, associated in an information source with the search term, wherein the specific type or types of second information is dependent at least in part on the type or types of the first information, and (ii) performing an action using at least part of the second information;

in consequence of ~~following~~ receipt by the first computer program of ~~a~~ the user command from the input device, causing a search ~~searching~~ for the ~~first information~~ search term in the

information source, using a second computer program, in order to find second information related to the search term ~~first information~~; and

if searching finds any second information related to the search term ~~first information~~, performing the action ~~an operation~~ using at least part of the second information, wherein the action is of a type depending at least in part on the type or types of the first information.

2. (Original) A method according to claim 1, wherein the first information comprises at least one of name-, person-, company- and address-related information.

3. (Currently Amended) A method according to claim 1, wherein performing the action ~~operation~~ includes performing ~~an operation~~ the action in the first computer program.

4. (Currently Amended)A method according to claim 2, wherein performing the action ~~operation~~ includes performing the action ~~an operation~~ in the first computer program.

5. (Currently Amended) A method according to claim 3, wherein performing the action ~~operation~~ includes causing addition of at least part of ~~adding~~ the second information to the first information in the document.

6. (Currently Amended) A method according to claim 3, wherein performing the action ~~operation~~ includes causing display of at least part of ~~displaying~~ the second information.

7. (Currently Amended) A method according to claim 3, wherein performing the action ~~operation~~

includes causing display of at least part of ~~displaying~~ the second information by the first computer program.

8. (Currently Amended) A method according to claim 1, further comprising, providing a prompt for updating the information source ~~application~~ to include the first information.

9. (Currently Amended) A method according to claim 1, further comprising, if the search is not successful, providing a prompt for updating ~~a database~~ the information source ~~associated with the second computer program~~ to include the first information.

10. (Currently Amended) A method according to claim 1, wherein receipt by the first computer program of the user command precedes analyzing the document.

11. (Original) A method according to claim 1, wherein analyzing the document is completed after the receipt of the user command is completed and before searching is initiated.

12. (Currently Amended) A method according to claim 1, wherein the input device is ~~further comprising providing~~ a graphical input device ~~for entry of the user command~~.

13. (Currently Amended) A method according to claim 1, wherein the user command is the only command from a user necessary as a condition to initiate ~~initiating~~ performing the operation.

14. (Currently Amended) A method according to claim 1, ~~further comprising providing a graphical input device for entry of the user command,~~ wherein the input device is a menu[[,]] and the entry of the user command includes a user's selection of the menu and click on a menu choice from the menu.

15. (Currently Amended) A method according to claim 1, further comprising, if searching results in a plurality of distinct instances of second information, displaying such instances to enable user selection of one of them for use in performing the action ~~operation~~.

16. (Currently Amended) A method according to claim 1, wherein the information source ~~a database~~ is associated with the second computer program and is available on the computer.

17. (Currently Amended) A method according to claim 1, wherein the information source ~~a database~~ is associated with the second computer program and is available through the computer.

18. (Currently Amended) A computer-implemented method for finding data related to the contents of a document using a first computer program running on a computer, the method comprising:

displaying the document electronically using the first computer program;

while the document is being displayed, analyzing, in a computer process on the computer, ~~the document to identify any~~ first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information, and wherein the first information

comprises at least one of name-, person-, company-, and address-related information;

providing an input device, configured by the first computer program, that allows a user to enter a user command to initiate an operation, the operation comprising (i) performing a search using at least part of the first information as a search term in order to find the second information, of a specific type or types, associated with the search term in a user editable information source outside the document, wherein the specific type or types of second information is dependent at least in part on the type or types of the first information, and (ii) performing an action using at least part of the second information ~~for entry of a user command~~, wherein the input device includes a menu;[[,]]

retrieving the first information ~~thus identified~~;

in consequence of ~~following~~ receipt by the first computer program of the user command, such user command including a user's selection of the menu and click on a menu choice from the menu, causing a search ~~searching~~ for the search term in the information source ~~first information~~, using a second computer program, in order to find second information related to the search term ~~first information~~ in ~~a~~ the user editable information source outside the document; and

if searching finds any second information related to the search term ~~first information~~, performing the action ~~an operation~~ using at least part of the second information, wherein the action is of a type depending at least in part on the type or types of the first information and performing the action ~~operation~~ includes at least causing display of ~~displaying~~ at least part of the second information.

19. (Currently Amended) A method according to claim 18, further comprising, if searching results in a plurality of occurrences of second information, causing display of ~~displaying~~ such

instances to enable user selection of one of them for use in performing the action ~~operation~~.

20. (Currently Amended) A method according to claim 18, wherein performing the action ~~operation~~ includes causing addition of at least part of ~~adding~~ the second information to the first information in the document.

21. (New) A method according to claim 1, wherein performing the action includes causing insertion of at least part of the second information into the document.

22. (New) A method according to claim 1, wherein performing the action includes causing insertion of at least part of the second information into the document by the first computer program.

23. (New) At least one non-transitory computer readable medium encoded with instructions which, when loaded on a computer, establish processes for finding data related to the contents of a document using a first computer program running on a computer, the processes comprising:

displaying the document electronically using the first computer program;

while the document is being displayed, analyzing, in a computer process, first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information;

retrieving the first information;

providing an input device, configured by the first computer program, that allows a user to enter a user command to initiate an operation, the operation comprising (i) performing a search using at least part of the first information as a search term in order to find the second information, of a specific type or types, associated in an information source with the search term, wherein the specific type or types of second information is dependent at least in part on the type or types of the first information, and (ii) performing an action using at least part of the second information;

in consequence of receipt by the first computer program of ~~a~~ the user command from the input device, causing a search for the search term in the information source, using a second computer program, in order to find second information related to the search term; and

if searching finds any second information related to the search term, performing the action using at least part of the second information, wherein the action is of a type depending at least in part on the type or types of the first information.

24. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein the first information comprises at least one of name-, person-, company- and address-related information.

25. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein performing the action includes performing the action in the first computer program.

26. (New) At least one non-transitory computer readable medium according to claim 24, the instructions further establishing processes wherein performing the action includes performing the action in the first computer program.

27. (New) At least one non-transitory computer readable medium according to claim 25, the instructions further establishing processes wherein performing the action includes causing addition of at least part of the second information to the first information in the document.

28. (New) At least one non-transitory computer readable medium according to claim 25, the instructions further establishing processes wherein performing the action includes causing display of at least part of the second information.

29. (New) At least one non-transitory computer readable medium according to claim 25, the instructions further establishing processes wherein performing the action includes causing display of at least part of the second information by the first computer program.

30. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes comprising:

providing a prompt for updating the information source to include the first information.

31. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes comprising:

if the search is not successful, providing a prompt for updating the information source to include the first information.

32. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein receipt by the first computer program of the user command precedes analyzing the document.

33. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein analyzing the document is completed after the receipt of the user command is completed and before searching is initiated.

34. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein the input device is a graphical input device.

35. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein the user command is the only command from a user necessary as a condition to initiate performing the operation.

36. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein the input device is a menu and the entry of the user command includes a user's selection of the menu and click on a menu choice from the menu.

37. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes comprising:

if searching results in a plurality of distinct instances of second information, displaying such instances to enable user selection of one of them for use in performing the action.

38. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein the information source is associated with the second computer program and is available on the computer.

39. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein the information source is associated with the second computer program and is available through the computer.

40. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein performing the action includes causing insertion of at least part of the second information into the document.

41. (New) At least one non-transitory computer readable medium according to claim 23, the instructions further establishing processes wherein performing the action includes causing insertion of at least part of the second information into the document by the first computer program.

42. (New) At least one non-transitory computer readable medium encoded with instructions which, when loaded on a computer, establish processes for finding data related to the contents of a document using a first computer program running on a computer, the processes comprising:

displaying the document electronically using the first computer program;

while the document is being displayed, analyzing, in a computer process on the computer, first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information, and wherein the first information comprises at least one of name-, person-, company-, and address-related information;

providing an input device, configured by the first computer program, that allows a user to enter a user command to initiate an operation, the operation comprising (i) performing a search using at least part of the first information as a search term in order to find the second information, of a specific type or types, associated with the search term in a user editable information source outside the document, wherein the specific type or types of second

information is dependent at least in part on the type or types of the first information, and (ii) performing an action using at least part of the second information, wherein the input device includes a menu;

retrieving the first information;

in consequence of receipt by the first computer program of the user command, such user command including a user's selection of the menu and click on a menu choice from the menu, causing a search for the search term in the information source, using a second computer program, in order to find second information related to the search term in the user editable information source outside the document; and

if searching finds any second information related to the search term, performing the action using at least part of the second information, wherein the action is of a type depending at least in part on the type or types of the first information and performing the action includes at least causing display of at least part of the second information.

43. (New) At least one non-transitory computer readable medium according to claim 42, the instructions further establishing processes comprising:

if searching results in a plurality of occurrences of second information, causing display of such instances to enable user selection of one of them for use in performing the action.

44. (New) At least one non-transitory computer readable medium according to claim 42, the instructions further establishing processes wherein performing the action includes causing addition of at least part of the second information to the first information in the document.

REMARKS

This Supplemental Response adds new claims 23-44, which are computer readable medium claims that mirror claims 1-22. Otherwise, this Supplemental Response is substantively identical to the Response filed by Applicant on December 8, 2010.

The Applicant thanks Examiner Tran for his analysis of the pending claims and for his time during the examiner interview on November 24, 2010, in which the Tso, Person, and Miller references were discussed with Bruce Sunstein and Jakub Michna, attorneys for the Applicant, and Atle Hedloy, the inventor and Applicant herein. The Examiner agreed that the claims, as amended, distinguish the cited prior art. Examiner Tran also asked the Applicant to further clarify the analyzing and providing processes of the independent claims. The amendments to the claims and the remarks below address the Examiner's concerns expressed during the interview.

Claims 1-44 are currently pending in the application. Claims 21-44 are new. Claims 1, 3-10, and 12-20 are currently amended. No new matter has been added. The rejections and objections to the claims are addressed below.

Objections

The office action objects to claims 16 and 17 as duplicate claims. Claim 16, however, is not a duplicate of claim 17. Claim 16 requires an information source that is "available *on* the computer", while claim 17 requires an information source that is "available *through* the computer."

The Applicant filed with the response of a December 8, 2010 a corrected information disclosure statement (IDS), relisting those references that the Examiner did not consider in the previous IDS of January 26, 2009 and also listing new references.

Section 3 of the IDS filed herewith lists issued and pending applications that are related to the present application. Applicant notes that Application Serial No. 12/841,302 (also before the Examiner) and the prior art references analyzed in the Accelerated Examination Support Document (AESD) of July 22, 2010 are of particular interest in relation to the present application.

Non-Statutory Double Patenting

A terminal disclaimer was filed on December 8, 2010 to overcome the non-statutory double patenting rejection.

Amendments to the Claims

Claim 1 is directed to a computer-implemented method for finding data related to the contents of a document using a first computer program. The claim has been amended to require analyzing first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information. Support for this amendment can be found in the present application at, for example, page 7, line 8 to page 8, line 8. The claim also requires providing an input device to allow a user to enter a user command to initiate an operation. Support for such an input device can be found at, for example, page 1, lines 17-22, page 3, lines 7-13, and page 5, lines 18-22 of the application. According to the claim, the operation comprises:

(i) "performing a search using at least part of the first information as a search term in order to find the second information, of a specific type or types, associated in an information source with the search term, …, wherein the specific type or types of second information is dependent at least in part on the type or types of the first information"

(ii) "performing an action using at least part of the second information."

As amended, the claim requires that the action is of a type depending at least in part on the type or types of the first information. Support for these amendments can be found in the Figures of the present application at, for example, Figs. 1a, 1b, 2a, 2b and at items 6, 8, 10, 12, 14, 26, 27, 28, 29, 30, 31, 32, 34, 36. Support can also be found in the present application at, for example, page 7, line 10 – page 8, line 20. No order is intended in recitation of the claim limitations except as may be otherwise required by the claims.

Applicant notes that the current amendments to the claims have not been made for reasons of patentability. Applicant believes that the original claims were patentable over the cited prior art at least because none of the cited references discloses "analyzing a document to identify any first information", as required by the original claims.

Accordingly, Applicant now amends the claims, not to overcome the cited prior art, but instead to provide more context and clarity to the claims. In fact, the limitation described in the previous paragraph has been amended out of the claims, which, in that respect, broadens the claims. On the other hand, the amendments have narrowed the claims in other respects. Yet these narrowing amendments are merely intended to streamline and facilitate prosecution of the claims and are not made to overcome the cited prior art.

Distinction of the Claimed Subject Matter from the Art of Record

Claims 1-20 are rejected as obvious in view of the combination of the Tso reference (US Patent No. 6,085,201) and the Person reference ("Special Edition Using Microsoft Word 97"). This combination, however, does not meet the limitations of the amended claims for at least the reasons provided below.

As amended, claim 1 requires:

1. "analyzing first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information"
2. "performing a search … in order to find the second information, of a specific type or types, … wherein the specific type or types of second information is dependent at least in part on the type or types of the first information"
3. "performing the action using at least part of the second information, wherein the action is of a type depending at least in part on the type or types of the first information."

None of the cited prior art references discloses all three of these limitations, which are powerful in distinguishing most, if not all, of the art of record.

For example, with respect to the present rejection against the claims, none of the cited prior art references discloses item (1) above, "analyzing first information from the document to determine if the first information is at least one of a plurality of types of information that can be searched for in order to find second information related to the first information."

Tso discloses composing an e-mail message with the help of a context-sensitive template engine. Tso, Abstract. When the template engine receives an input text string, Tso discloses "decomposing" the text string into individual words and comparing the individual words to keywords associated with predetermined templates. Tso, col. 4, lines

32-35, col. 5, lines 7-17. The template with the best match between keywords and individual words is chosen as the most appropriate template and that template is presented to the user. Tso, col. 5, lines 7-17.

Tso does not analyze first information "from the document to *determine if the first information is at least one of a plurality of types* of information that can be searched for in order to find second information related to the first information", as required by claim 1. This is so because Tso decomposes and compares every word in the input text string irrespective of the type of information that is within the text string.

The Person reference does not make up for the deficiencies of the Tso reference. The Person reference describes a functionality in Microsoft Word 97 known as Mail Merge. Mail Merge allows a user to create multiple letters "by merging together a list of names and addresses with letters, envelopes, or address labels." Person, page 485. The method starts when the user selects a main document and inserts merge fields into the document for specifying the placement of merged information. The merge fields also specify a particular data field that is to replace them: "First_Name", "Last_Name", and "Company_Name." *See* Person, Fig. 17.30; Person, page 485. Next, the user selects a list of contacts that includes data fields for name and address information. Person, page 488. When the user selects the "merge" button, the name and address information from the list of contacts is retrieved and inserted into the main document according to the merge fields. Person, page 492, 507.

The Person reference also does not disclose or suggest item (1) above, analyzing first information from the document *to determine if the first information is at least one of a plurality of types of information* that can be *searched* for in order to find second

information related to the first information, as required by the claims. Instead, as explained above, the Person reference merely discloses inserting data retrieved from a list of contacts *en masse* into merge fields within the document. Person fails to disclose or suggest analyzing first information in the document, as required by the claims, because Person requires the merge fields in the document to already have been specified by the user as merge fields and as related to a specific data fields in the list of contacts (*e.g.*, "First_Name", "Last_Name", and "Company_Name"). Thus, there is no need to analyze text to determine its type.

Additionally, Person fails to disclose or suggest a search for first information, as required by item (2) above, because the information to be inserted by mail merger is never searched for in the information source. Rather, the records specified by the user (*i.e.*, *every* contact from the contact list) are retrieved and used for insertion into the main document. Since there is no search, as required by the claims, Person also cannot determine if the first information is of a type *that can be searched*, as further required by item (1) above. Accordingly, neither the Tso reference nor the Person reference discloses or suggests analyzing first information from the document to determine if the first information is of a type of information that can be searched for in order to find second information related to the first information, as required by claim 1. And, for this reason, among others, the claims are patentable over the combination of the Tso and Person references.

Independent claim 18 (a method claim) and independent claims 23 and 42 (corresponding media claims) require limitations that are similar to claim 1 and,

therefore, claim 18, 23, and 42 are patentable over the cited prior art for similar reasons. The dependent claims are also allowable for similar reasons.

The presently amended claims also distinguish the Miller reference (U.S. Patent No. 5,946, 647), which has been used in a rejection against the claims in Application Serial No. 12/841,302 (also before the Examiner). Miller discloses a method for detecting data in a document and performing a particular action on the detected data. Miller, Abstract. Miller does not disclose "performing a search … in order to find … second information, of a specific type or types, associated in an information source", by item (2) above. In Miller, no search is performed at all. Furthermore, since Miller does not search for second information, it also does not disclose "performing the action *using at least part of the second information*", as required by item (3) above. For at least these reasons, all of the present claims distinguish the Miller reference.

Applicant believes that all of the rejections have been addressed and a notice of allowance is respectfully solicited. If any fees are required, please charge deposit account number 19-4972. To further expedite prosecution, the Examiner may call Bruce Sunstein or Jakub Michna at 617-443-9292 if he has any further questions.

Respectfully submitted,

/Bruce D. Sunstein, #27,234/

Bruce D. Sunstein
Registration No. 27,234

/Jakub M. Michna, #61,033/

Jakub M. Michna
Registration No. 61,033

Attorneys for Applicant

SUNSTEIN KANN MURPHY
& TIMBERS LLP
125 Summer Street
Boston MA 02110-1618
Tel: 617 443 9292
Fax: 617 443 0004
03324/00102 1371465.1