EXHIBIT 2001

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SAMSUNG ELECTRONICS AMERICA, INC.
Petitioner

v.

KANNUU PTY LTD.
Patent Owner

Case IPR2020-TBD
Patent No. 8,676,852

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 8,676,852 UNDER 35 U.S.C § 312 AND 37 C.F.R. § 42.104

Mail Stop "PATENT BOARD"
Patent Trial and Appeal Board
U.S. Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

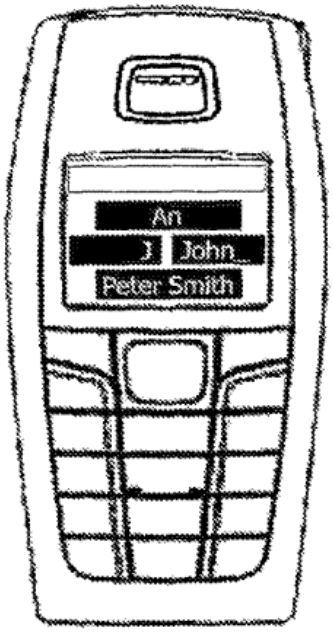
EXHIBIT LIST

No.	Short Name	Exhibit
1001	’852 Patent	U.S. Patent No. 8,676,852 to Dinn
1002	’852 File History	File History of U.S. Patent No. 8,676,852
1003	Forlines Decl.	Declaration of Clifton Forlines, Ph.D.
1004	Forlines CV	Curriculum Vitae of Clifton Forlines, Ph.D.
1005	*Perlman*	U.S. Pat. Pub. No. 2002/0113825 A1 to Perlman *et al.*
1006	*Pu*	U.S. Patent No. 7,152,213 to Pu *et al.*
1007	*Krohn*	U.S. Patent No. 6,593,913 to Krohn *et al.*
1008	*Schroeder*	U.S. Patent No. 5,797,098 to Schroeder *et al.*
1009	*Strubbe*	U.S. Patent No. 5,223,924 to Strubbe
1010	*Hargrove*	U.S. Patent No. 5,787,417 to Hargrove
1011	*Dostie*	U.S. Pat. Pub. No. 2004/0021691 to Dostie *et al.*
1012	SDNY Docket	Docket Report (as of March 3, 2020) for *Kannuu Pty Ltd. v. Samsung Electronics Co., Ltd. et al.*, Case No. 1:19-cv-04297-ER (S.D.N.Y.)

I. INTRODUCTION

Samsung Electronics America, Inc. ("Samsung" or "Petitioner") respectfully requests *inter partes* review ("IPR") of Claims 1-22 ("Challenged Claims") of U.S. Patent No. 8,676,852 to Dinn ("the '852 Patent"), Exhibit ("Ex.") 1001, under 35 U.S.C. §§311-319, 37 C.F.R. §§42.1-.80 and 37 C.F.R. §§42.100-.123. The '852 Patent is assigned to Kannuu Pty Ltd. ("Patent Owner" or "Kannuu").

The '852 Patent describes various techniques for more quickly selecting items from a database, such as names from an electronic telephone address book. Ex. 1001, Abstract; 4:61-5:12. The four most likely items in the database are presented on a display screen corresponding to the left, right, up, and down positions of a joystick, sometimes referred to as a 4-way direction-pad or d-pad. *Id.*, 4:38-45. The user can then select an item from one of the available displayed items using the joystick, and the item is appended to the "name thus far specified." *Id.* at 4:46-53. If the selection is not complete, the user can repeat the process to append one of the next four most likely items to the previous selection. *Id.* Figures 5 and 6 show the first two selections corresponding to the name "John Smith" contained in the user's address book. As seen in Figure 6, once the name "John" is selected, the most likely last names (including "Smith") are then displayed for selection.



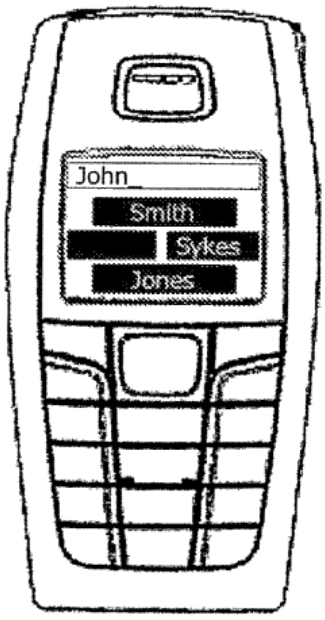


FIGURE 5 **FIGURE 6**

Id., Figs. 5 and 6. According to the '852 Patent, this selection method allows for fewer key presses or movements of the joystick because "the most likely letters are presented first" and the use of an on-screen labels (as opposed to printed on buttons) allows users to "maintain their gaze on the screen." *Id.*, 5:1-19.

As described in more detail below, however, Patent Owner did not invent the claimed selection method. In the late 1990s and early 2000s—well before the earliest possible priority date of the '852 Patent—significant efforts were being undertaken to make textual input more efficient using input devices with reduced set keyboards, such as mobile telephones and television remote controls. Numerous enhanced user interfaces with "dynamic" displays (that re-positioned letters, groups of letters, or words based on the likelihood of being inputted next)

were developed to input textual data more efficiently and without the use of a standard keyboard. As shown below, each of these interfaces discloses or renders obvious all the elements of the Challenged Claims. The Challenged Claims are therefore invalidated by the prior art presented in this Petition. Accordingly, the Board should grant this Petition and find the Challenged Claims of the '852 Patent invalid.

II. MANDATORY NOTICES

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

Petitioner certifies that the '852 Patent is available for *inter partes* review and Petitioner is not barred or estopped from requesting review of the Challenged Claims on the grounds identified in this Petition.

B. Real Party-in-Interest (37 C.F.R. § 42.8(b)(1))

The real parties-in-interest for Petitioner are Samsung Electronics America, Inc. and Samsung Electronics Co., Ltd.

C. Related Matters (37 C.F.R. § 42.8(b)(2))

The following judicial or administrative matters may affect or be affected by a decision in this proceeding: *Kannuu Pty Ltd. v. Samsung Electronics Co., Ltd. et al.*, 1:19-cv-04297-ER (S.D.N.Y.). In that action, Patent Owner has asserted the '852 Patent. The '852 Patent shares the same specification as U.S. Patent Nos. 8,370,393; 9,436,354; and 8,996,579. These patents also are asserted in the above litigation, and Petitioner is concurrently filing petitions challenging these patents.

Moreover, Patent Owner also has asserted U.S. Patent No. 9,697,264 in the above litigation, and Petitioner also is concurrently filing a petition challenging that patent.

D. Lead and Backup Counsel and Service Information (37 C.F.R. § 42.8(b)(3)-(4))

The designations of counsel and address for service[1] are:

LEAD COUNSEL	BACKUP COUNSEL
Kevin P.B. Johnson Reg. No. 38927 kevinjohnson@quinnemanuel.com **Postal and Hand Delivery Address:** Quinn Emanuel Urquhart & Sullivan 555 Twin Dolphin Drive, 5th Floor Redwood City, CA 94065 Tel: (650) 801-5000 Fax: (650) 801-5100	Brian Mack Reg. No. 57189 brianmack@quinnemanuel.com **Postal and Hand Delivery Address:** Quinn Emanuel Urquhart & Sullivan 50 California St, 22nd Floor San Francisco, CA 94111 Tel: (415) 875-6600 Fax: (415) 875-6700 Marissa Ducca Reg. No. 59807 marissaducca@quinnemanuel.com **Postal and Hand Delivery Address:** Quinn Emanuel Urquhart & Sullivan 1300 I Street NW, Suite 900 Washington DC 20005 Tel: (202) 538-8000 Fax: (202) 538-8100 John T. McKee Reg. No. 65926

[1] Petitioner consents to electronic service to QE-Samsung-Kannuu@quinnemanuel.com and the email addresses listed in the table below.

	johnmckee@quinnemanuel.com **Postal and Hand Delivery Address:** Quinn Emanuel Urquhart & Sullivan 51 Madison Ave, 22nd Floor New York, New York 10010 Tel: (212) 849-7000 Fax: (212) 849-7100

E. Fee for *Inter Partes* Review

The required fees are submitted herewith. Petitioner authorizes the Patent Office to charge Deposit Account No. 505708 for any additional fees required for this filing.

III. IDENTIFICATION OF CHALLENGE AND RELIEF REQUESTED

Petitioner challenges claims 1-22 of the '852 Patent and requests that these claims be found unpatentable in view of the following references:

1. U.S. Pat. Pub. No. 2002/0113825 A1 to Perlman *et al.*, filed on March 26, 2001 and published on August 22, 2002 (Ex. 1005, hereinafter "*Perlman*");

2. U.S. Patent No. 7,152,213 to Pu *et al.*, filed on October 4, 2001, published as U.S. Pat. Pub. No. 2003/0067495 A1 on April 10, 2003, and issued on December 19, 2006 (Ex. 1006, hereinafter "*Pu*");

3. U.S. Patent No. 6,593,913 to Krohn *et al.*, filed on March 14, 2000 and issued on July 15, 2003 (Ex. 1007, hereinafter "*Krohn*"); and

4. U.S. Pat. Pub. No. 2004/0021691 A1 to Dostie *et al.*, filed on October 18, 2001 and published on February 5, 2004 (Ex. 1011, hereinafter "*Dostie*").

Pursuant to §§42.22(a)(1) and 42.22(b)(1)-(2), Petitioner requests cancellation of claims 1-22 of the '852 Patent on the following grounds:

Ground 1: Claims 1-8 and 12-19 are unpatentable under 35 U.S.C. §103(a) in view of *Perlman* and *Dostie*.

Ground 2: Claims 9-11 and 20-22 are unpatentable under 35 U.S.C. §103(a) in view of *Perlman*, *Dostie*, and *Pu*.

Ground 3: Claims 1-22 are unpatentable under 35 U.S.C. §103(a) in view of *Pu* and *Dostie.*

Ground 4: Claims 1-22 are unpatentable under 35 U.S.C. §103(a) in view of *Krohn*, *Dostie*, and *Pu*.

A. How the Claims are Unpatentable (37 C.F.R. § 42.104(b)(4))

Pursuant to 37 C.F.R. §42.104(b)(4), an explanation of how the Challenged Claims of the '852 Patent are unpatentable under the statutory grounds identified above, including the identification of where each element of the claim is found in the prior art, is provided in Section VIII, *infra*.

B. Supporting Evidence (37 C.F.R. § 42.104(b)(5))

Pursuant to 37 C.F.R. §42.104(b)(5), the exhibit numbers of the supporting evidence relied upon and the relevance of the evidence to the challenges raised, including identification of specific portions of the evidence that support the challenge, are provided herein. An Exhibit List identifying the exhibits is included. *See supra* at iii. In further support of the proposed grounds of rejection,

this Petition is accompanied by the declaration of Dr. Clifton Forlines, Ex. 1003 ("Forlines Decl.").

IV. OVERVIEW OF THE TECHNOLOGY

As described in the Introduction section above, the alleged invention described in the Challenged Claims of the '852 Patent relates to techniques for more quickly selecting an item from a database, such as a name from an electronic telephone address book. The techniques aim to limit the number of keystrokes needed to traverse the database to the desired item. As acknowledged in the background of the '852 Patent itself, there was a well-known problem inputting textual data into on-screen user interfaces without a full keyboard. Ex. 1001, 1:39-2:34. Mobile phones, PDAs, television remote controls, and set-top boxes all shared this problem. Forlines Decl. ¶39.

A. Predictive Keyboards

The use of on-screen keyboards exploded in the 1990s, thanks to the proliferation of interactive televisions and smart phones. Almost immediately thereafter, user interface designers began working on ways to simplify and expedite textual input using the on-screen keyboards. Forlines Decl. ¶¶40-54. For example, U.S. Patent No. 5,797,098 to Schroeder *et al.* (Ex. 1008), filed in 1995, describes an improved mobile telephone user interface with both a predictive keyboard input method and automatic word completion. By using the predictive

keyboard input method described in *Schroeder*, the most commonly used characters for selected words in a particular language can be associated with programmable keys, and then after accepting a user input of a character, a next set of characters is displayed that are "most likely to occur after the character previously input by the user." Ex. 1008, 1:39-45, 4:20-6:15.

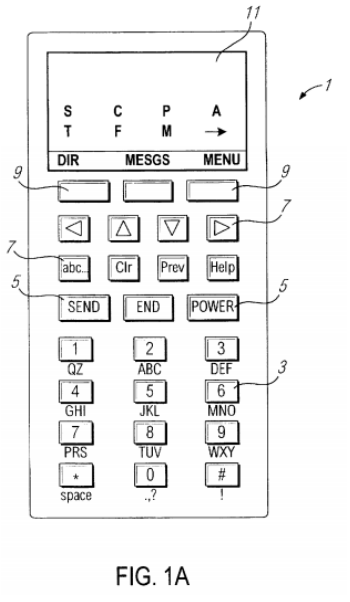


FIG. 1A

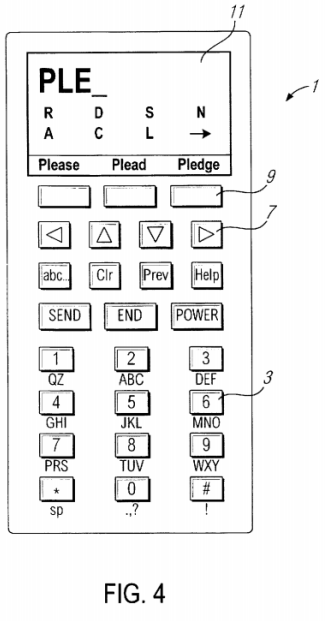


FIG. 4

B. Database Indexing, Searching, and Selection

Databases have been used to store information, such as names, addresses, and messages for decades prior to the '852 Patent. Forlines Decl. ¶¶55-60. Almost immediately after databases for storing vast amounts of information were developed, methods for automatically indexing and searching the information were also developed. *Id.*; *see also* U.S. Patent No. 3,309,677, filed January 2, 1964,

entitled "Automatic Information Indexing" (indexing information in databases and searching based on indices).

As one example, U.S. Patent No. 5,223,924 to Strubbe (Ex. 1009), filed in 1992, discloses "program information databases" for television programs that are automatically correlated with the preferences of the user. The program information databases are indexed by start time, channel, and/or "liked" or "disliked" indications selected by the user. Ex. 1009, 4:17-5:32. A user browses the data in the databases by specifying one or more of these indices. *Id.* A "free text" search field also allows users to perform simply queries over the records in the program information databases and display the results. *Id.*, 5:48-6:24.

Database searches routinely used indices, lookup tables, and hierarchical search trees to carry out efficient database searches. For example, Figure 6 of *Schroeder* and Figures 4 and 14 of *Dostie* (described above) show a dictionary tree used for searching, auto-completion, and candidate character/word prediction. *Schroeder* even acknowledges that even by 1995, when *Schroeder* was filed, "[t]he implementation of such dictionary trees or lookup tables in the computer arts [was] well known." Ex. 1008, 7:10-25.

V. THE '852 PATENT

A. Overview

As explained herein, the '852 Patent relates to a method and apparatus for selecting database items from a database. Ex. 1001, Abstract. Database items are indexed by item identifiers, such as text. *Id.* A user may select one or more parts of an identifier from a display, where the one or more parts may be selected by adding to the previously selected one or more parts to build a larger part or whole of an item identifier. In this way, "[s]election from a large list of item identifiers may be carried out in a relatively short time period." *Id.* The specification provides an example of selecting a contact from an address book within a mobile telephone. *Id.*, 4:11-13.

B. Prosecution History

The '852 Patent was filed on January 8, 2013 and claims the benefit of Australian Prov. App. No. 2005904378, filed August 12, 2005.[2] Ex. 1002, .201-.238.

[2] For the purposes of this Petition only, Petitioner has assumed the '852 Patent is entitled to its August 12, 2005 provisional priority date but may take other positions in the underlying district court litigation.

On September 11, 2013, the Examiner held a telephonic interview with the applicant, where minor amendments to the claims were suggested to the applicant. *Id.*, .122-.123.

On September 19, 2013, the Examinee issued a Notice of Allowance with Examiner's amendment correcting minor informalities in the claims. The application issued as U.S. Patent No. 8,676,852 on March 18, 2014. *Id.*, .01.

C. Level of Ordinary Skill in the Art

The '852 Patent claims priority to Australian Prov. App. No. 2005904378, filed August 12, 2005. A person of ordinary skill in the art ("POSITA") at that time would have had at least an undergraduate degree in electrical engineering, computer science, or physics (or a related field, such as computer engineering, human-computer interaction, or industrial design) and 2-3 years of work experience with input interfaces to electronic devices. Forlines Decl. ¶¶14-20. Each of the arguments below is made from the standpoint of a POSITA in the field of the '852 Patent. Petitioner's expert, Dr. Forlines, was at least a POSITA at the time of the alleged invention. *Id.*; Ex. 1004 (noting Master's degree in Human-Computer Interaction from Carnegie Mellon University in 2001 and significant work and teaching experience in user input interfaces for a number of years prior to 2005).

VI. CONSTRUCTION OF THE CHALLENGED CLAIMS[3]

In an *inter partes* review, claims are "construed using the same claim construction standard that would be used to construe the claim in a civil action under 35 U.S.C. 282(b)." 37 C.F.R. §42.100(b). Claims must be given their ordinary and customary meaning as understood by one of ordinary skill in the art at the time of the invention in light of the specification and the prosecution history pertaining to the patent. *Id.*; *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312-1313 (Fed. Cir. 2015) (*en banc*); see also 83 Fed. Reg. 51,340.

Petitioner submits that no claim terms require construction for the resolution of this Petition. Forlines Decl. ¶¶80-81 & n.5.

VII. OVERVIEW OF THE PRIOR ART REFERENCES

As shown in the table below, each of the cited references qualifies as prior art to the Challenged Claims, which are entitled to a priority date of no earlier than August 12, 2005.

---

[3] Petitioners reserve the right to pursue different claim constructions, including that certain claim terms are indefinite, during this and related proceedings as well as in any district court litigation concerning the '852 Patent. *See Samsung Elecs. Am., Inc. v. Prisua Eng'g Corp.*, 948 F.3d 1342, 1350 (Fed. Cir. 2020) ("[T]he Board may not cancel claims for indefiniteness in an IPR proceeding.").

Prior Art Reference	Publication/Issue Date	Section(s)
Perlman	August 22, 2002	102(b), 103
Pu	April 10, 2003 (as U.S. Pat. Pub. No. 2003/0067495 A1)	102(b), 103
Krohn	July 15, 2003	102(b), 103
Dostie	February 5, 2004	102(b), 103

A. *Perlman*

Perlman (Ex. 1005) relates to an improved user input interface for a television remote control having functional buttons arranged in a "star" pattern. Ex. 1005, ¶[0014]; Forlines Decl. ¶¶61-63. A first group of characters to be mapped to a group of remote control buttons is selected based on how probable it is that individual characters within the first group correspond to a first character of a word representing information sought by a user in a database. Then, the first group of characters are mapped to the group of remote control buttons. *Id.*, Abstract. For example, the function buttons 101 on remote control 100 of Figure 1b could be mapped to the characters in buttons 110 displayed on television/computer display 105 in Figure 2.

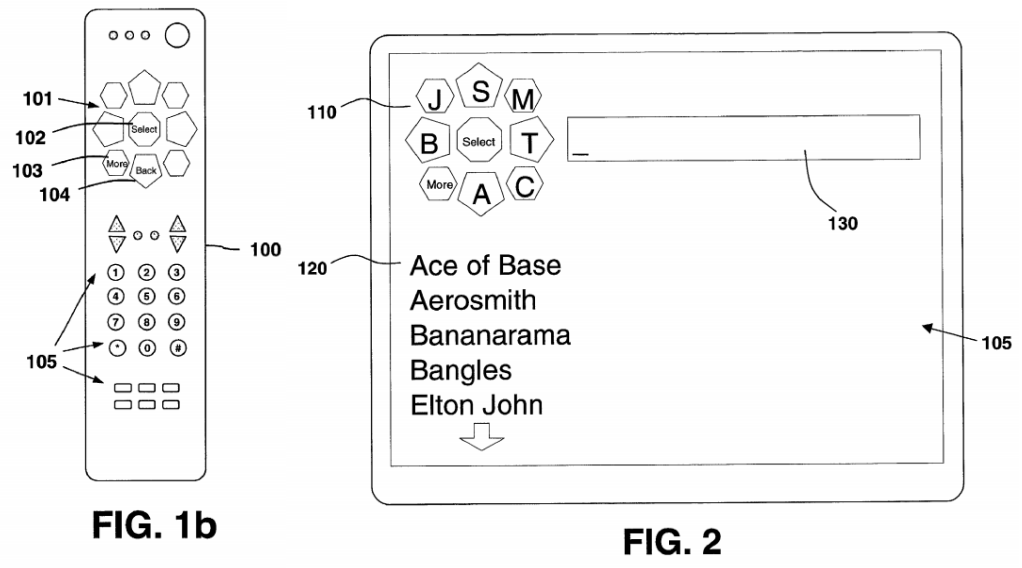


FIG. 1b

FIG. 2

The characters mapped to the functions buttons (and displayed on the display) correspond to "the most common first letters of words in the database list 120." *Id.*, ¶[0017]. In the depicted embodiments, the database list corresponds to "a list of available multimedia content." *Id.* The operation of the improved user input interface is described below:

> As the user selects letters via the displayed character-mapped buttons 110, the user's letter selections will appear in a text box 130, and the database list 120 will change to reflect the user's selections. Initially (i.e., before the user has selected a character), the alphanumeric characters mapped to the character-mapped buttons 101, 110 correspond to the most common first letters of words in the database list 120. Once the user selects a first character, a new set of characters are mapped to the character-mapped buttons 101, 110 which correspond to the most common second letters of words in the database list 120 which begin with the first-selected

> character. As the user continues to select characters, only those multimedia programs/files with titles which begin with the selected characters will be displayed in the list 120.

Id.

Individual characters (*i.e.*, letters) or groups of characters (*i.e.*, multiple letters) may be displayed within each button 110 (*id.*, ¶[0029]), and the letter/letter groups may be selected based on "commonality," "user's preferences," or a weighted combination of the two ("e.g., 50% character incidence/50% user preferences; 75% character incidence/25% user preferences") (*id.*, ¶[0019]). Various types of built-in dictionaries and "pre-trained" word sequences may also be used to prioritize the linking operation between function buttons and letter/letter groups. *Id.*, ¶[0035].

As can be seen from the above description and according to Dr. Forlines, the invention described in *Perlman* is "virtually indistinguishable" from the invention recited in the Challenged Claims. Forlines Decl. ¶63. They both address the same problem and reach the same solution—namely, a dynamic keyboard that re-positions the next most likely characters for input around directional input buttons.

B. *Pu*

Pu (Ex. 1006) also relates to an improved user input interface that uses "dynamic key assignment" and "a predefined list that is presented to a user in an arrangement that statistically reduces the number of keystrokes required for data

entry." Ex. 1006, Title & Abstract. The presentation is the result of "determining the relative frequency of each valid selection in the predefined list and presenting those valid selections with the highest frequency items in a position that minimizes the number keystrokes required for data entry." *Id.*; Forlines Decl. ¶¶64-68.

As a user is inputting characters into the display, tables are created listing the possibilities for the next character input according to the frequency of that input. For example, if "SAN" were entered by the user in display area 408, frequency table 404 would indicate that the next four most likely letters are '_' (space), 'T', 'D', and 'G'. These four possibilities are then displayed in section list 410.

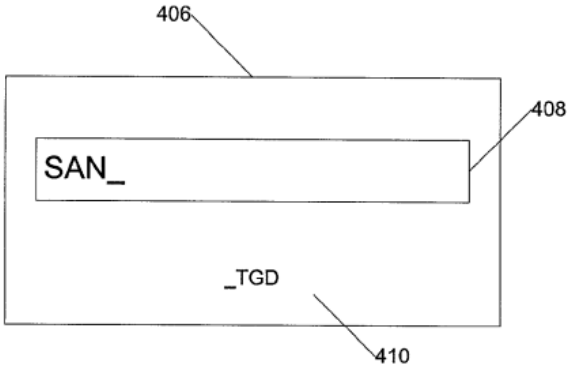
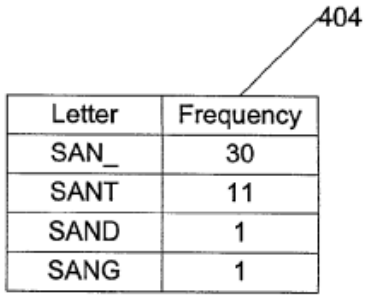


FIG. 6C



Letter	Frequency
SAN_	30
SANT	11
SAND	1
SANG	1

FIG. 6B

Pu explains that the "dynamic key assignment" can be accomplished using "a standard telephone keypad can be used in combination with a display that shows the relationship between the physical keys and the associated characters" or a "virtual keypad that is presented on a touch screen display." *Id.*, 10:50-64. In

addition, the input control system can dynamically change the key labels of an on-screen display via software control. *Id.*

As can be seen from the above description, the invention described in *Pu* is "virtually identical" to the invention described in the '852 Patent and recited in the Challenged Claims. *Id.* Forlines Decl. ¶68. They both address the same problem and reach the same solution—namely, a dynamic keyboard that re-positions the next most likely characters for input at the four directions of a shuttle control system (up, down, left, and right).

C. *Krohn*

Krohn (Ex. 1007) also relates to an improved user input interface for selecting a character with a user input device comprising a plurality of buttons. Ex. 1007, Abstract; Forlines Decl. ¶¶69-71. A first plurality of characters is displayed on a display device in a pattern corresponding to a pattern of a plurality of buttons of a user input device, and a character from the first plurality of characters is selected in response to actuation of one of the plurality of buttons. The number of characters displayed on the display device for selection by the user input device is less than or equal to the number of buttons in the plurality of buttons. In this way, any of the characters displayed on the display device for selection by the user input device can be selected by actuation of a single one of the plurality of buttons. *Id.*, Abstract.

Krohn describes the use of a "character prediction method" to select the characters from the character set that are "the most likely to be selected or follow the first selected character." *Id.*, 4:29-36. For example, if the first selected character is 'Q', the processor can determine that 'U' is one of the four most-likely characters to be chosen after 'Q' and can present 'U' as one of the characters in the second plurality of characters for selection by the user. *Id.*

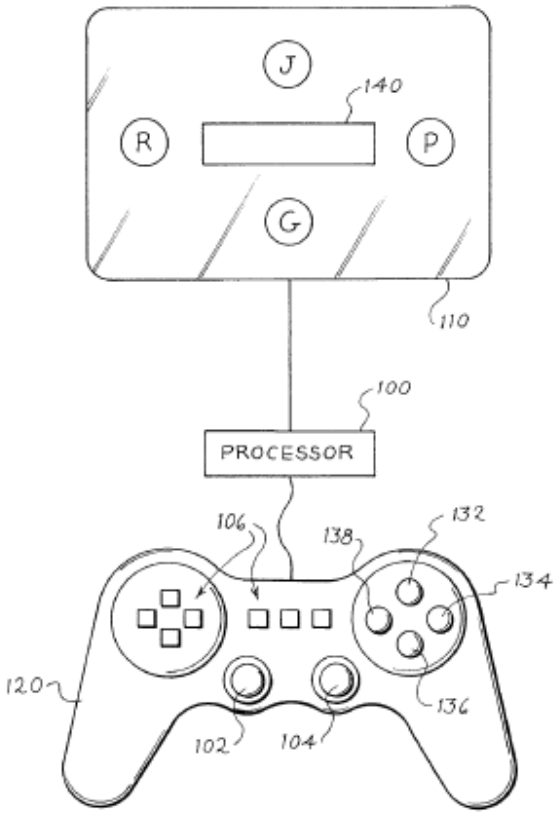


Fig. 1

Figure 1, above, shows display device 110 with character entry field 140 and user input device 120. The processor 100 arranges and displays a first plurality of characters on the display device 110 in a pattern corresponding to a pattern of the plurality of buttons 132, 134, 136, 138 of the user input device 120. *Id.*, 3:38-59. As can be seen above in the example of Figure 1, the plurality of buttons can be arranged in an up, down, left, right configuration. As the user inputs characters, new characters are populated in the "circles" on display device 110, enabling the user to input a complete word or correct/incorrect answer to a prompt. *Id.*, 4:9-54. The circles are also arranged in an up, down, left, right configuration. As such, *Krohn* is "nearly indistinguishable" from the invention recited in the Challenged Claims. Forlines Decl. ¶71.

D. *Dostie*

Dostie (Ex. 1011) describes a personal computing device that supports the "rapid search for data, such as text," on digital devices, such as PDAs and cell phones, by having the user input one or more characters and then using the inputted character sequence to search a dictionary for a set of completion candidates that begin with the typed sequence. *Dostie* at Abstract; ¶[0073]; Forlines Decl. ¶¶77-79. Those completion candidates are chosen so that they are "more likely to be the ones that the user is attempting to type." *Id.*, ¶[0073]. Figures 4, 14, 19, and 22 of *Dostie* show the tree-based representation of the

database and the relationship between characters in the tree. For example, in Figure 14, the keys 'a' 'n' and 'y' have already been typed by the user, and the system suggests the characters 'o' 'h' and 'w' based on the current location in the tree. *Id.*, ¶¶[0088]-[0090]. Furthermore, *Dostie* discloses that full sequences of characters ("completion candidates") be presented based on the current location in the tree. *Id*. In the example of Figure 14, having typed the word "any," the system would then return the candidates "any" "anyone" "anyhow" and "anyway" and order them based on their "preference values" (which could be based on their frequency of prior selection or frequency in an English language or personal dictionary). *Id.*, ¶¶[0085], [0209]-[0215]

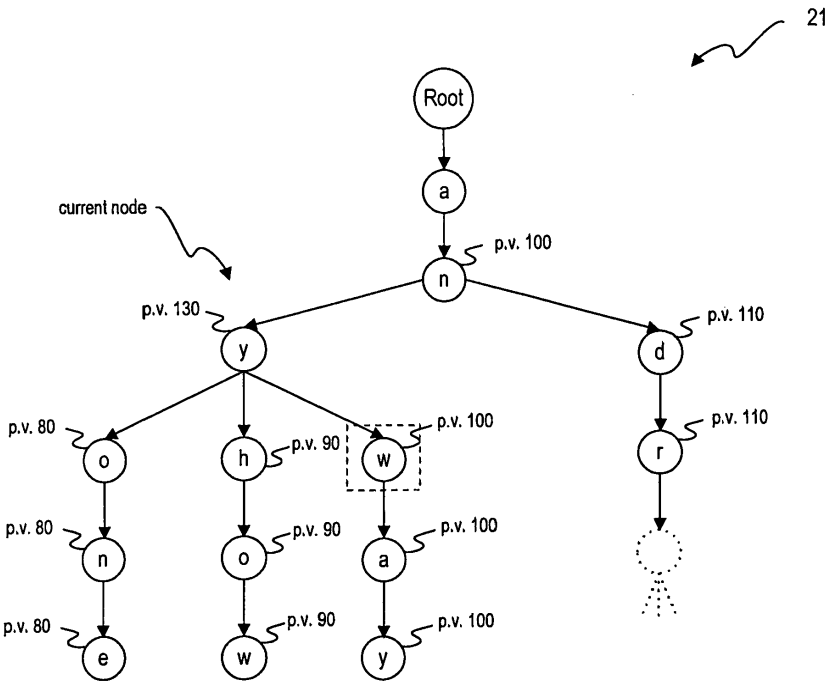


FIG. 14

Figure 3 from *Dostie* shows one method of presenting these completion candidates (either whole words or partial words). In this example, having typed "bu", the completion candidates presented are partial completion candidates (called "chunks" by *Dostie*), which include "build…" "built…" "business…" "but…" and "buy…". *Id.*, ¶¶[0079], [0087]. The ellipses indicates that the dictionary contains one or more further completion candidates that begin with this chunk (for example, "buy" "buying" "buyback" "buyoff" for "buy…"). *Id.*

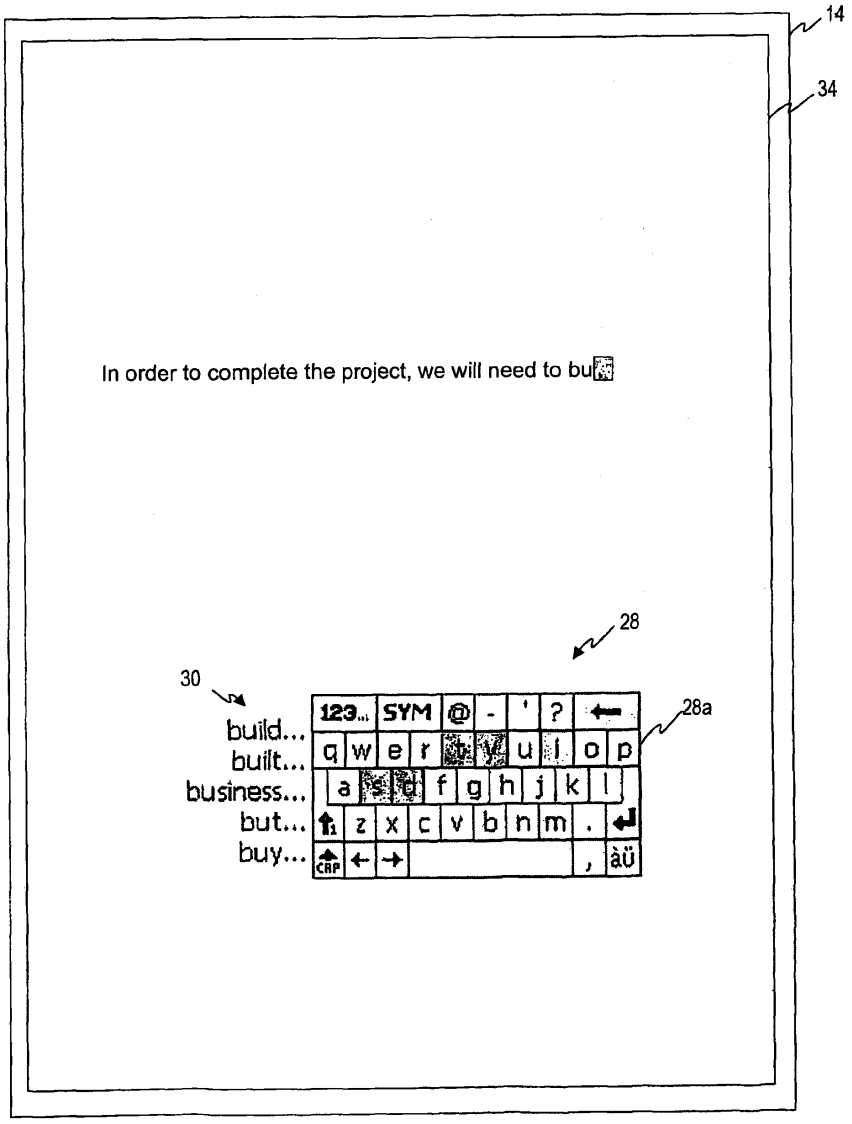


FIG. 3

VIII. DETAILED EXPLANATION OF GROUNDS FOR INVALIDITY

As explained below pursuant to 37 C.F.R. §42.104(b), the cited prior art renders anticipates or renders obvious the Challenged Claims of the '852 Patent. The cited references ***were not considered*** by the Patent Office during the

examination of the '852 Patent.[4] The Challenged Claims are unpatentable based on each of the grounds presented below.

A. Ground 1: Claims 1-8 and 12-19 are obvious in view of *Perlman* and *Dostie*

1. Claim 1

a) Preamble: "A computer-implemented method of selecting an item from a plurality of items, the method comprising:"

To the extent the preamble is considered limiting, *Perlman* discloses this limitation. Forlines Decl. ¶288. *Perlman* describes various "embodiments of an apparatus and method for selecting data" including an embodiment that "allows user to rapidly enter alphanumeric characters (or other types of symbols) without looking away from the television/computer screen." Ex. 1005, ¶[0014]. The alphanumeric characters/symbols described in Perlman constitute a plurality of items, and the embodiments described therein relate to selecting alphanumeric characters/symbols from a plurality of characters/symbols.

b) [1A]: "generating, by at least one computer processor, a first display, the first display comprises: a part of an item identifier for at least a first set of items and a part of an item identifier for at least a second set of items in a database;"

Perlman discloses this limitation. Forlines Decl. ¶¶289-90. In Figures 2-8, Perlman describes a GUI with character-mapped buttons 110 in a "star"

[4] *See* Ex. 1001, (56) (References Cited).

configuration corresponding to the same "star" configuration on a remote control input device to a television/computer. *Id.*, ¶[0014]. Other configurations are also expressly disclosed. Ex. 1005, ¶[0016].

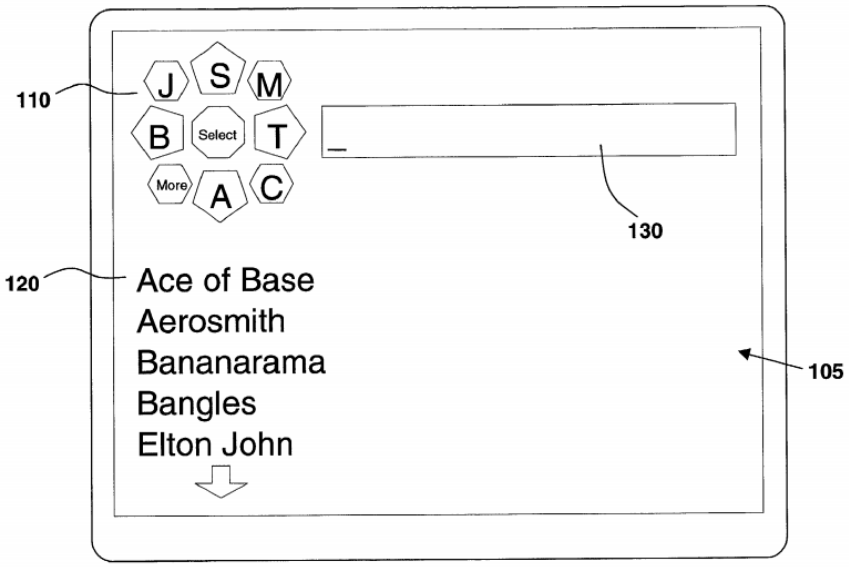


FIG. 2

Id., Fig. 2.

The "star" configuration includes within character-mapped buttons 110 a part of an item identifier for at least a first set of items and a part of an item identifier for at least a second set of items in a database. For example, the characters shown in Figure 2 correspond to the "most common first letters of words in the database list 120," which corresponds to "available multimedia content" accessible by the television/computer. Ex. 1005, ¶[0017]. Although the embodiment shown in Figure 2 shows multimedia content, *Perlman* notes that the invention is "not limited to any particular type of database" and in fact describes

multiple different types of content databases, such as those for television "program content" (*id.*, ¶[0004]), "multimedia programs/files" (*id.*, ¶[0017]), and "MP3 music" titles (*id.*, Figs. 7-8). Each of these different types of content databases includes a different set of items comprising the entries in that particular database.[5] Forlines Decl. ¶290.

c) *[1B]: "enabling, by the at least one computer processor, selection of one of the two parts of the item identifiers by a user using a user interface;"*

Perlman discloses this limitation. Forlines Decl. ¶291. A user may select a letter from the GUI by using the character-mapped buttons of the star configuration of the remote control device. Ex. 1005, ¶[0017] ("As the user selects letters via the displayed character-mapped buttons 110, the user's letter selections will appear in a text box 130, and the database list 120 will change to reflect the user's selections.").

d) *[1C]: "generating, by the at least one computer processor, in response to the selection of the one of the two parts, a further display; wherein the further display comprises: an additional part of an item identifier for at least a third set of items and an additional part of an*

[5] Moreover, each part of an item identifier (*i.e.*, letter) would correspond to the set of content/titles having that item identifier as a first letter. Forlines Decl. ¶290 & n.11. In this way, each displayed part of an item indenter is for a different set of items in the database.

item identifier for at least a fourth set of items; and the previously selected one of the two parts of the item identifiers;"

Perlman discloses this limitation. Forlines Decl. ¶292. The user's previously selected one of the two parts of the item identifiers is displayed in text box 130 while a different set of item identifiers are displayed on character-mapped buttons 110, as shown in Figures 2 and 3 of *Perlman*. Ex. 1005, ¶[0017] ("As the user selects letters via the displayed character-mapped buttons 110, the user's letter selections will appear in a text box 130, and the database list 120 will change to reflect the user's selections.").

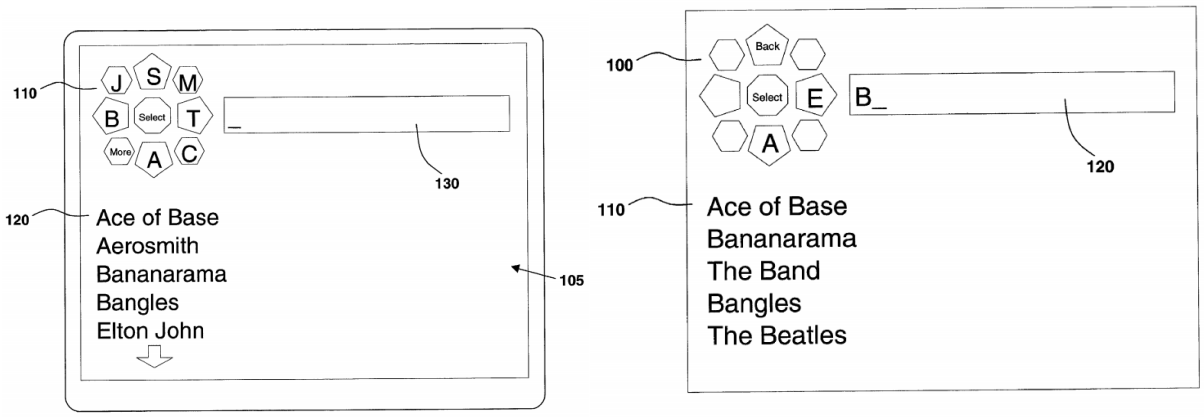


FIG. 2 **FIG. 3**

Id., Figs. 2 & 3.

As shown above, once the part 'B' of the item identifier is selected, database list 110 is updated to reflected database items containing a word beginning with the selected letter. Additional parts of item identifiers 'A' and 'E' are both displayed. Part 'A' corresponds to the set of entries in the database containing 'BA' as the

first two letters of a word in the database item, while part 'E' corresponds to the set of entries in the database containing 'BE' as the first two letters of a word in the database item. Forlines Decl. ¶293.

e) [1D]: "enabling, by the at least one computer processor, selection of one of the two additional parts of the item identifiers by the user using the user interface;"

Perlman discloses this limitation. Forlines Decl. ¶294. After the user selects a letter (*e.g.*, the character 'B'), a new set of parts of item identifiers are automatically mapped to the character-mapped buttons of the "star" configuration GUI and displayed for another user selection. Ex. 1005, ¶[0017] ("Initially (i.e., before the user has selected a character), the alphanumeric characters mapped to the character-mapped buttons 101, 110 correspond to the most common first letters of words in the database list 120. Once the user selects a first character, a new set of characters are mapped to the character-mapped buttons 101, 110 which correspond to the most common second letters of words in the database list 120 which begin with the first-selected character.").

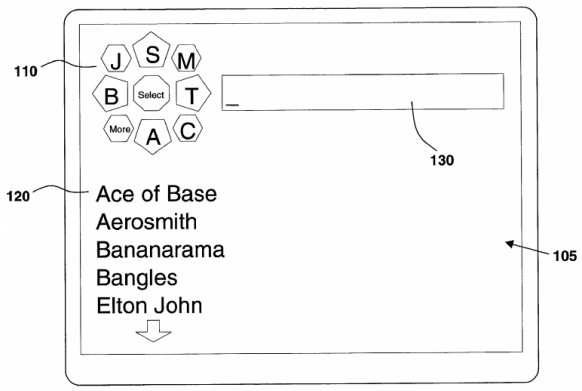


FIG. 2

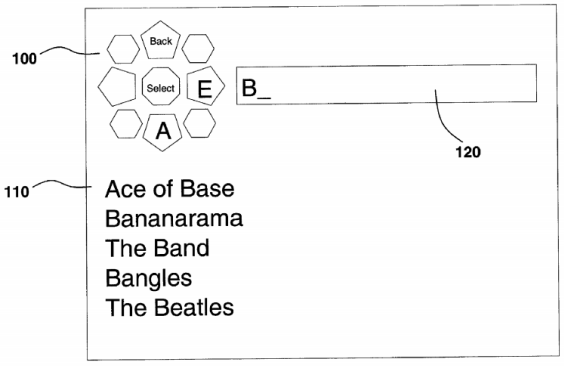


FIG. 3

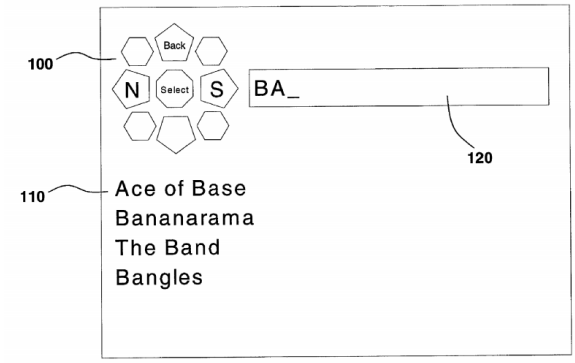


FIG. 4

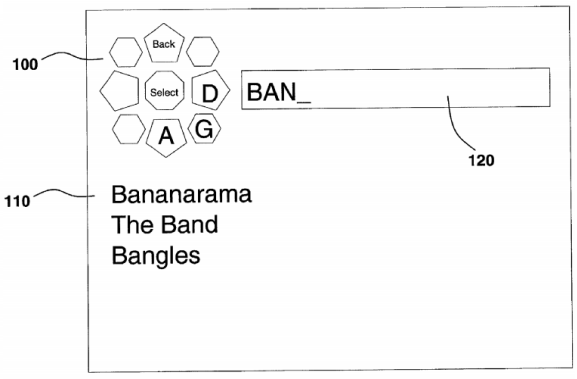


FIG. 5

Id., Figs. 2-5.

In the above figures, at least parts of item identifiers ‘E’ and ‘A’ (Figure 3) are chosen for display based on the user’s prior selection of the part ‘B’. At least portions ‘N’ and ‘S’ (Figure 4) are chosen for display based on the user’s prior selection of the parts ‘B’ and ‘A’. At least parts ‘D’, ‘G’, and ‘A’ (Figure 5) are chosen for display in the first area based on the user’s prior selection of the parts ‘B’, ‘A’, and ‘N’. Forlines Decl. ¶295.

f) [1E]: "combining, by the at least one computer processor, the selected one of the two parts of the item identifiers with the selected one of the two additional parts of the item identifiers to create a larger part of the item identifiers; and"

Perlman discloses this limitation. Forlines Decl. ¶296. As shown in Figures 2-5, the selected portions are combined in text box 130 (Figure 2) to form a larger portion of the multimedia program or title. Ex. 1005, ¶¶[0017], [0025]-[0026] and Figs. 2-5.

g) [1F]: "displaying, by the at least one computer processor, the larger part of the item identifiers;"

Perlman discloses this limitation. Forlines Decl. ¶297. As shown in Figures 2-5, the selected portions are displayed in text box 130 (Figure 2) to form a larger part of the multimedia program or title. Ex. 1005, ¶[0017] and Figs. 2-5.

h) [1G]: "wherein the parts of the item identifiers and the additional parts of the item identifiers are text symbols and the additional parts of the item identifiers are shorter than complete item identifiers; and"

Perlman discloses this limitation. Forlines Decl. ¶298. All the parts of item identifiers are letters or characters within the English alphabet, which are text symbols in the English language. Ex. 1005, ¶¶[0004], [0017]. A built-in dictionary comprising English letters or characters may also be used. *Id.*, ¶¶[0035]-[0036]. The additional parts of item identifiers (*i.e.*, the single letters representing the second letter of the content/title) are shorter than complete item identifiers, as shown above in Figures 2-5.

i) [1H]: "the first set of items identifiers and the second set of item identifiers are mutually exclusive of one another and the third set of item identifiers and the fourth set of item identifiers are mutually exclusive subsets of the first set of item identifiers or the second set of item identifiers."

Perlman discloses this limitation. Forlines Decl. ¶¶299-302. As explained above, once a letter is selected, a new set of item identifiers is displayed reflecting "the most common second letters of words in the database list 120 which begin with the first-selected character." Ex. 1005, ¶[0017]. Because each part of an item identifier (*i.e.*, letter) corresponds to the matching content/titles having that item identifier as a first letter, the first and second sets of item identifiers (which correspond to different letters in *Perlman*) are mutually exclusive. As one simple example, the part 'J' shown in Figure 2 would correspond to the set of all content/titles starting with 'J', while the part 'S' would correspond to the set of all content/titles starting with 'S'. Since the content/titles cannot start with both 'J' and 'S', these two sets of item identifiers would be mutually exclusive.

Moreover, the third set of item identifiers and the fourth set of item identifiers are mutually exclusive subsets of the first set of item identifiers or the second set of item identifiers because each of those sets requires a first and second letter to match. As the user enters each additional letter, the set of matching content/titles is reduced to a subset of the prior set. Forlines Decl. ¶300.

Dostie also discloses this limitation. *Dostie*'s implementation organizes its database in a "tree structure" which can take the form of a binary tree or "B-tree." Ex. 1011, ¶[0088]. Because "[c]haracters making up a completion candidate are stored in separate nodes within the tree structure" and "each node pointing to a child node containing the next character in the completion candidate,"*Dostie*'s tree structure discloses that the first set of items identifiers and the second set of item identifiers are mutually exclusive of one another and the third set of item identifiers and the fourth set of item identifiers are mutually exclusive subsets of the first set of item identifiers or the second set of item identifiers. Forlines Decl. ¶301.

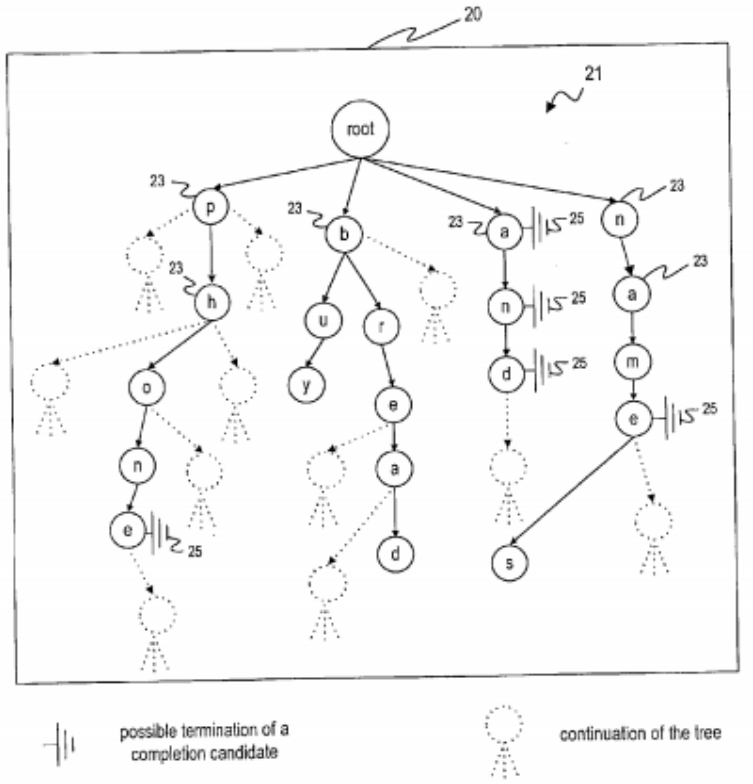


FIG. 4

For example, Figure 4 of *Dostie* shows various sets of item identifiers that are mutually exclusive of one another and subsets of one another. The mutually exclusive sets would constitute different braches within candidate tree 21, while the mutually exclusive subsets would constitute all items below a certain node in the same branch or a different branch. *Id.* The mutually exclusive subsets would also be described by the "child list" or "child nodes pointer" for each node in the tree, which represent "a list of nodes which directly follow the current node, i.e. a list of nodes which are the immediate children of the current node." *Id.;* Ex. 1011, ¶¶[0090]-[0096]. It would have been obvious to incorporate *Dostie*'s tree structure

into *Perlman* for the same reasons described below in connection with claim 5 and because *Dostie* itself recognizes that use of a candidate tree helps "to rapidly predict potential completion candidates" and "provides a mechanism for supporting enhanced data entry techniques such as character prediction." Ex. 1011, ¶[0097]. The "enhanced data entry techniques" and "character prediction" are the same techniques described by *Perlman*, so a POSITA would have found it natural to incorporate *Dostie*'s candidate tree into the character selection system of *Perlman*. Forlines Decl. ¶302.

2. Claim 2

Claim 2 recites "[t]he method in accordance with claim 1, comprising a further step of iterating the selections until the item identifiers are completed." *Perlman* discloses this limitation. Forlines Decl. ¶303. A user can continue selecting additional letters until the multimedia program or title is completed. Ex. 1005, ¶¶[0017], [0025]-[0026] (describing examples in which the user iterates the selections until 'ACE OF BASE' or 'BANGLES' is selected using the select button).

3. Claim 3

Claim 3 recites "[t]he method in accordance with claim 1, wherein the step of generating the first display includes selecting the parts of the item identifiers for the first and second set of items to be displayed based on predetermined criteria."

Perlman discloses this limitation. Forlines Decl. ¶304. The selected two parts of item identifiers are selected based on "the commonality of each of the characters" and "the user's preferences for each of the multimedia selections." Ex. 1005, ¶[0019]. In addition, predetermined weights may be assigned to the selection criteria. *Id.* (describing the selection based on "0% character incidence/50% user preferences; 75% character incidence/25% user preferences"). At least the user preferences and the predetermined weights are two examples of a predetermined criteria. Forlines Decl. ¶304.

4. Claim 4

Claim 4 recites "[t]he method in accordance with claim 3, wherein one of the predetermined criteria is a frequency of selection of database items associated within the item identifier parts for the first and second set of items." *Perlman* discloses this limitation. Forlines Decl. ¶305. As described above in connection with claims 1 and 3, the selected two portions of item identifiers are selected because they are the "most common" first/second/third letters of words in the database list. Ex. 1005, ¶[0017]. In addition, "the user's preferences for each of the multimedia selections" may be taken into account in a weighted selection. *Id.*, ¶[0019]. The user's preferences can be based on whether "a user plays back a particular multimedia program more frequently than any other program." *Id.*

In addition, Perlman supports "multi-word typing" where the most likely series of words are displayed based on monitoring the "user's prior typing" or how frequently the user types a certain word. *Id.*, ¶[0033]-[0034]. A built-in dictionary can also be automatically extended each time the user enters a full word and used in prioritizing parts of item identifiers for display. *Id.*, ¶[0034]. Each of these examples uses "a frequency of selection of database items associated within the item identifier parts for the first and second set of items" as the predetermined criteria. Forlines Decl. ¶306.

5. Claim 5

Claim 5 recites "[t]he method in accordance with claim 3, wherein one of the predetermined criteria is a ranking of the parts of item identifiers for the first and second set of items in relation to a tree hierarchy classification of the parts of item identifiers for the first and second set of items." *Perlman* in view of *Dostie* renders this claim obvious. Forlines Decl. ¶307.

In *Perlman*, "the list of entries 120 will continuously change to reflect the user's character selections." Ex. 1005, ¶[0025]. After the user inputs a first character, only those multimedia programs or titles containing that first character will be displayed in the list of entries and "the only possible second letters of words in the list which begin with [the selected second character]" are displayed. *Id.* This process continues until a complete multimedia program or title is inputted.

Id., ¶¶[0025]-[0026]. Because the process determines all possible next letters from the set of database entries matching the letters inputted so far, it would have been obvious in view of *Perlman* alone to use a ranking in relation to a tree hierarchy classification as the predetermined criteria. Forlines Decl. ¶307. In other words, "because the system in *Perlman* searches through all possible database items matching the first *N* characters where *N* corresponds to the letter position of the character currently being inputted by the user and displays matching titles in the database list, it would make intuitive sense to use a tree hierarchy classification (*i.e.*, one with levels and orders) based on the sequencing and ordering of letters within the matching database items." *Id.*

Dostie also discloses this feature. *Dostie* shows a dictionary organized in a hierarchical tree structure where character or word completion candidates are stored in the nodes of the hierarchical tree structure (which may take the form of a binary tree or B-tree). Ex. 1011, ¶¶[0088]-[0090]. As shown in Figure 14, a "preference value" (denoted "p.v.") can be associated with each node within the tree for "ranking completion candidates relative to each other." *Id.*, ¶[0085], Figs. 4 & 14.

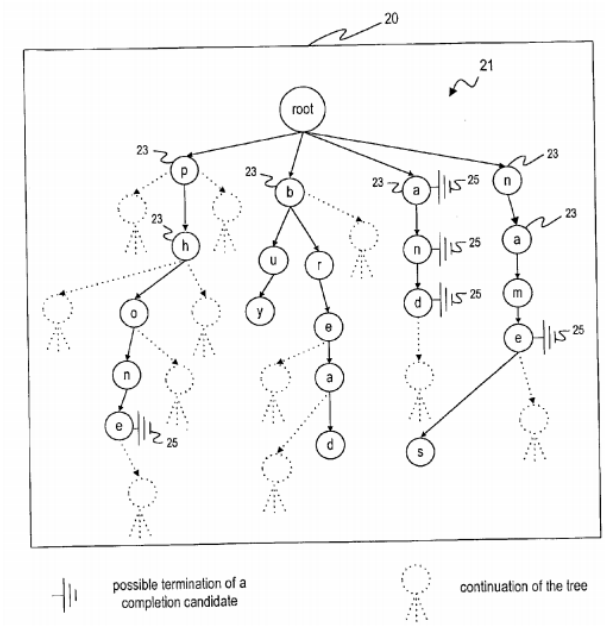


FIG. 4

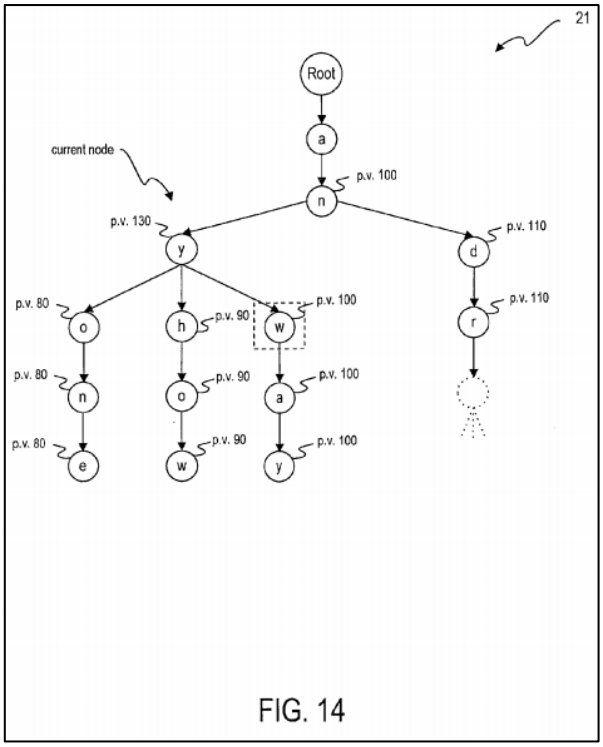


FIG. 14

The tree structures shown in Figures 4, 14, 19, and 22 of *Dostie* are each "a ranking of the parts of item identifiers for the first and second set of items in relation to a tree hierarchy classification of the parts of item identifiers for the first and second set of items." Forlines Decl. ¶¶308-310. Therefore, to the extent this limitation is not already disclosed by *Perlman*, *Dostie* clearly discloses such a classification system.

6. Claim 6

Claim 6 recites "[t]he method in accordance with claim 1, wherein the item identifiers are text strings." *Perlman* discloses this limitation. Forlines Decl. ¶311. In addition to character item identifiers, *Perlman* supports groups of letters and "textual multi-word typing" where "the most likely series of words are provided in

the content list 110 when the user selects a particular set of characters (e.g., based on the user's prior typing)." Ex. 1005, ¶¶[0029] (groups of letters), [0033]-[0034] (words). Groups of letters and sets of characters, such as the examples given in *Perlman*—"Ace of Base" and "Bangles"—are text strings. *Id.*, ¶¶[0025]-[0026].

7. Claim 7

Claim 7 recites "[t]he method in accordance with claim 1, wherein the item identifiers are words." *Perlman* discloses this limitation. Forlines Decl. ¶312. As described above in connection with claim 6, "Ace of Base" and "Bangles" are both words. *Id.*

8. Claim 8

Claim 8 recites "[t]he method in accordance with claim 1, wherein the database is stored in a memory of the computing device." *Perlman* in view of *Dostie* renders this claim obvious. Forlines Decl. ¶313.

Although *Perlman* may be silent as to exactly where its database is stored, *Dostie* explains that "[t]he size of the dictionary 20 is limited only by the amount of memory available on the personal computing device concerned." Ex. 1011, ¶[0090]. *Dostie* also discloses an embodiment where its candidate tree is stored in "compact" form in order to "reduce the memory requirements" on the computing device. *Id.*, ¶[0220]. It would have been obvious to store the database in a memory of the computing device because that is the most readily accessible

storage location and would eliminate the need to access network resources in order to access the database. Forlines Decl. ¶314.

9. Claim 12

Claim 12 recites all the same elements as claim 1, but in system form. *Perlman* discloses all the limitations of claim 12 for the same reasons as described above in connection with claim 1. Forlines Decl. ¶315.

Perlman discloses an apparatus for selecting items, the apparatus comprising an output display, an input, and a computer processor. *See* Ex. 1005, ¶¶[0014] (television/computer display 105 and remote control device 100), [0037]-[0038] ("general-purpose or special-purpose processor"). The processor is configured to associate items with corresponding item identifiers as well as perform all the steps of claim 1. *Id.*, ¶¶[0017] (describing associating "available multimedia content" with items in database list 120), [0037]-[0038] (describing implementing all the steps using a processor).

10. Claim 13

Claim 13 recites the same features as claim 3 and is disclosed by *Perlman* for the same reasons. Forlines Decl. ¶317.

11. Claim 14

Claim 14 recites the same features as claim 4 and is disclosed by *Perlman* for the same reasons. Forlines Decl. ¶318.

12. Claim 15

Claim 15 recites the same features as claim 5 and is disclosed by *Perlman* for the same reasons. Forlines Decl. ¶319.

13. Claim 16

Claim 16 recites "[t]he apparatus in accordance with claim 15, including a memory storing the tree hierarchy." *Perlman* in view of *Dostie* renders this claim obvious. Forlines Decl. ¶320. *Dostie* discloses that its dictionary, which is organized in a tree hierarchy, is stored in data structure 29 within "memory available on the personal computing device concerned." Ex. 1011, ¶¶[0090]-[0091]. The most natural place to store the tree hierarchy would be in a memory. Forlines Decl. ¶320.

14. Claim 17

Claim 17 recites the same features as claim 6 and is disclosed by *Perlman* for the same reasons. Forlines Decl. ¶321.

15. Claim 18

Claim 18 recites the same features as claim 7 and is disclosed by *Perlman* for the same reasons. Forlines Decl. ¶322.

16. Claim 19

Claim 19 recites the same features as claim 8 and is obvious in view of *Perlman* and *Dostie* for the same reasons. Forlines Decl. ¶323.

17. Motivation to Combine *Perlman* and *Dostie*

It would have been obvious to incorporate *Dostie*'s dictionary trees (with mutually exclusive identifiers and subsets of identifiers) stored in memory of computing devices and rankings based on tree hierarchy classifications into *Perlman* because it was well known at the time of *Perlman* to conduct database searches using hierarchical trees like the ones described in *Dostie*. Forlines Decl. ¶324. Moreover, *Dostie* provides express motivation to incorporate its candidate trees into systems like *Perlman* in order to "rapidly predict potential completion candidates" and "provide[] a mechanism for supporting enhanced data entry techniques such as character prediction," like the mechanisms described in *Perlman*. Ex. 1011, ¶[0097].

As explained by Dr. Forlines, "the most efficient way to identify matching prefixes would be to use a hierarchical search tree." Forlines Decl. ¶325. This is because "a hierarchical search tree promotes efficient traversal forward and backward through a series of search candidates, which are often stored in nodes of the tree." *Id*. In addition, "the nodes of a search tree are often associated with weights, which can be used in the candidate ranking process." *Id.* Hierarchical trees also "promote simple ranking of candidates stored in the tree because the number of nodes in the search path through the tree can simply be summed (for example, ranking of 1 per node)." *Id.* In fact, database search techniques routinely

used search trees and other hierarchical classifications in the searching, ranking, and selection processes. *Id*. Moreover, as described above, *Perlman* already displays a sorted list of database items matching the first *N* letters of the user's input. At the time of *Perlman*, this form of search was almost always performed using a hierarchical search tree. *Id.* Because of the well-known benefits of using hierarchical search trees to search a database, these trees would have been "a trivial and routine addition" to *Perlman* that would have permitted simple and efficient searching based on a known prefix of database entries, like the searching discussed in *Perlman*. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 416 (2007) (combination of familiar elements according to known methods is likely obvious when it does no more than yield predictable results). This is especially true since *Perlman* discloses "forward and backward/reverse searching capabilities" (Ex. 1006 at [0027]) that would "likely have used hierarchical search trees as these were in standard use for database searching at the time to move back and forward through the tree. Forlines Decl. ¶325.

B. Ground 2: Claims 9-11 and 20-22 are obvious in view of *Perlman*, *Dostie*, and *Pu*

1. Claim 9

Claim 9 recites "[t]he method in accordance with claim 8, wherein the computing device is a mobile telephone." *Perlman* in view of *Dostie* and *Pu* renders this claim obvious. Forlines Decl. ¶326. *Pu* describes an enhanced user

interface for "intelligent devices" and explains that "[s]uch intelligent devices include automobile computers ('auto PCs'), major home appliances, personal digital assistants ('PDAs'), telephones, cellular telephones, internet phones, pagers, portable computers, navigational devices, and the like." Ex. 1006, 1:19-23. It would have been obvious to port *Perlman*'s enhanced input interface to mobile telephones because the benefits espoused by *Perlman* (more rapid text input without looking away from the display, *see* Ex. 1005, ¶[0006]) are equally applicable to mobile telephones. Forlines Decl. ¶326. Moreover, *Perlman* already contemplates various types of electronic "displays" (computer, television, and automobile in-dashboard displays) as well as "touch screens" that were widely used in mobile telephones at the time. Ex. 1005, ¶[0032]. A POSITA would therefore have ample motivation to port *Perlman*'s interface to a mobile telephone. Forlines Decl. ¶326.

2. Claim 10

Claim 10 recites "[t]he method in accordance with claim 8, wherein the computing device is a Personal Digital Assistant (PDA)." *Perlman* in view of *Dostie* and *Pu* renders this claim obvious. Forlines Decl. ¶327. *Pu* describes an enhanced user interface for intelligent devices and explains that "[s]uch intelligent devices include automobile computers ('auto PCs'), major home appliances, personal digital assistants ('PDAs'), telephones, cellular telephones, internet

phones, pagers, portable computers, navigational devices, and the like." Ex. 1006, 1:19-23. A POSITA would have ample motivation to port *Perlman*'s interface to a PDA for the same reasons discussed above with respect to claim 9.

3. Claim 11

Claim 11 recites "[t]he method in accordance with claim 8, wherein the steps of enabling selections allow selections of the one of the two parts or the one of the two additional parts of the item identifiers by way of a joystick." *Perlman* in view of *Dostie* and *Pu* renders this claim obvious. Forlines Decl. ¶328.

Pu uses a shuttle control system that is "implemented using a single joystick-like central key that can be pivoted up, down, right or left." Ex. 1006, 1:62-64. It would be obvious to allow selections of the parts of item identifiers by way of a joystick because this was one well-known way of selecting letters for database input using a mobile device or gaming system. Forlines Decl. ¶329. In addition, the '852 Patent itself acknowledges that "a menu driven by a joystick" was "commonly found on a mobile phone" and allowed a user to select menu options using the four directions selectable by the joystick. Ex. 1001 at 6:14-18.

4. Claim 20

Claim 20 recites the same features as claim 9 and is obvious for the same reasons. Forlines Decl. ¶330.

5. Claim 21

Claim 21 recites the same features as claim 10 and is obvious for the same reasons. Forlines Decl. ¶331.

6. Claim 22

Claim 22 recites the same features as claim 11 and is obvious for the same reasons. Forlines Decl. ¶332.

7. Motivation to Combine *Perlman, Dostie,* and *Pu*

Perlman and *Pu* both address very similar problems (textual input without use of a full keyboard) in a very similar way (by dynamically assigning characters to character-mapped buttons) in order to reduce the number of input keystrokes and decrease selection time. *KSR Int'l*, 550 U.S. at 417 ("[I]f a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill."). It would have been natural to incorporate the relevant teachings of *Pu* into the combination of *Perlman* and *Dostie* for all the reasons explained above and because *Pu* expressly indicates its teachings can be applied to any "intelligent device," home appliance, or device where use of standard keyboards are both "undesirable and impracticable." Ex. 1006 at 1:15-26. A POSITA reading *Pu* would have therefore tried to apply *Pu*'s teachings to *Perlman*'s system in order to achieve greater input efficiency on other types of devices requiring user input

using an input device without a full keyboard (such as using a joystick or shuttle system). Forlines Decl. ¶333. Moreover, the teachings of *Pu* being applied to the combination of *Perlman* and *Dostie* are all "trivial, routine, and commonplace design choices" well within the scope of a POSITA. *Id.*; *see also KSR Int'l*, 550 U.S. at 417. In other words, the small changes made to the combination of *Perlman* and *Dostie* to support databases stored in the memory of local computing devices, mobile telephones, and PDAs would been mere "workshop improvements" involving little or no technical difficulty and "would have not involved the exercise of any technical skill or ability beyond that to be expected of a POSITA." Forlines Decl. ¶333.

C. Ground 3: Claims 1-22 are obvious in view of *Pu* and *Dostie*

1. Claim 1

a) Preamble[6]

To the extent the preamble is considered limiting, *Pu* discloses this limitation. Forlines Decl. ¶334. *Pu* describes various "[a]n improved user interface for data input without the use of a standard keyboard" where "[t]he data that is entered is selected from a predefined list that is presented to a user in an arrangement that statistically reduces the number of keystrokes required for data

[6] The full claim language has been omitted in this and all subsequent grounds, but can be found in the analysis for Grounds 1-2.

entry." Ex. 1006, Abstract. For example, "a database comprising a predefined list of all cities" within certain states may be used as the plurality of items. *Id.*, 5:11-17.

b) [1A]

Pu discloses this limitation. Forlines Decl. ¶¶335-37. In Figures 3A-3C, *Pu* shows a "display area" 108 and a "selection list" 110. The selection list "contains only valid choices" based on the letters inputted so far and "the letters that are most frequently used appear at the beginning of the selection list." Ex. 1006, 6:62-7:4. In other words, "the most common letters appear closer to the beginning of the selection list" so that the "required keystrokes to select letters are statistically reduced." *Id.*, 7:5-15.

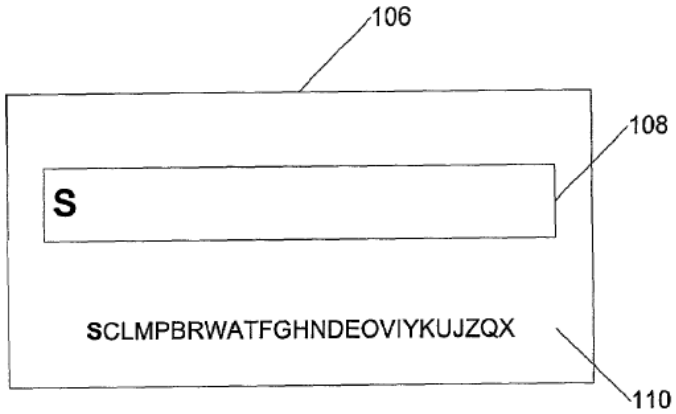


FIG. 3A

Id., Fig. 3A.

The valid choices presented in the selection list are parts of item identifiers for sets of items in a database. For example, the expected textual input could be a particular city within the states of California or Nevada. *Id.*, 5:11-17. The item identifiers (*i.e.*, names of the cities) correspond to entries in a "database comprising a predefined list of all cities within those two states." *Id.* Before the user enters any input into display area 108, selection list 110 includes parts corresponding to all valid first letters of the cities in the database. *See id.*, Fig. 2 (tables of alphabetically ordered lists corresponding to valid cities); 12:42-13:23 (discussing databases in data warehouses used to store the predefined lists).

The parts of item identifiers are for at least a first and second set of items in a database. For example, part 'S' is for all the cities in the database starting with the letter 'S,' while part 'C' is for all the cities in the database starting with the letter 'C'. Forlines Decl. ¶337. In addition to supporting a database storing city names, "all words defined for any particular language" may be included in another database. Ex. 1006, 5:47-50. The city names would therefore constitute a first set of items and the words in the French language would constitute the second set of items. Forlines Decl. ¶337.

c) [1B]

Pu discloses this limitation. Forlines Decl. ¶338. In one embodiment, a user may select a displayed letter from selection list 110 using a cursor and enter key. Ex. 1006, 7:5-15.

d) [1C]

Pu discloses this limitation. Forlines Decl. ¶339. The user's selection is displayed in display area 108/208 (Figure 3A or 4C), while additional parts of item identifiers are displayed in selection lists 110/210. *Id.*, 4:49-7:23, 8:22-45.

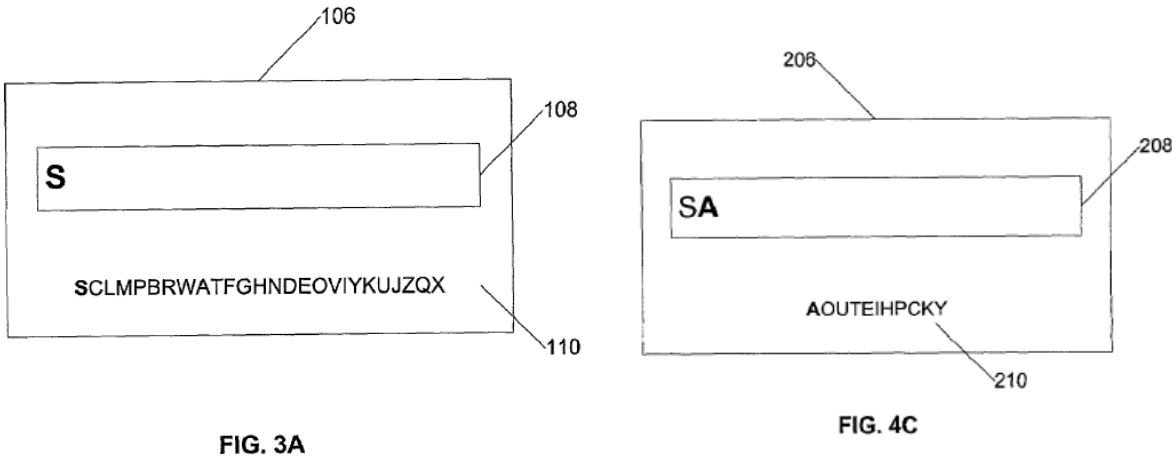


FIG. 3A

FIG. 4C

Id., Figs. 3A & 4C.

The additional parts of an item identifiers shown in Figure 4C are for at least a third set of items and a fourth set of items. For example, part 'A' corresponds to the set of cities in the database containing 'SA' as the first two letters of the city name, while part 'O' corresponds to the set of cities in the database containing 'SO' as the first two letters of the city name. Forlines Decl. ¶340.

e) [1D]

Pu discloses this limitation. Forlines Decl. ¶341. Just as the user selected a first letter corresponding to a city in the database, a second letter can be "selected by the user" at step 1112 so that a word (*e.g.*, a city name) can be inputted. *Id.*, 7:59-8:4, 13:28-51.

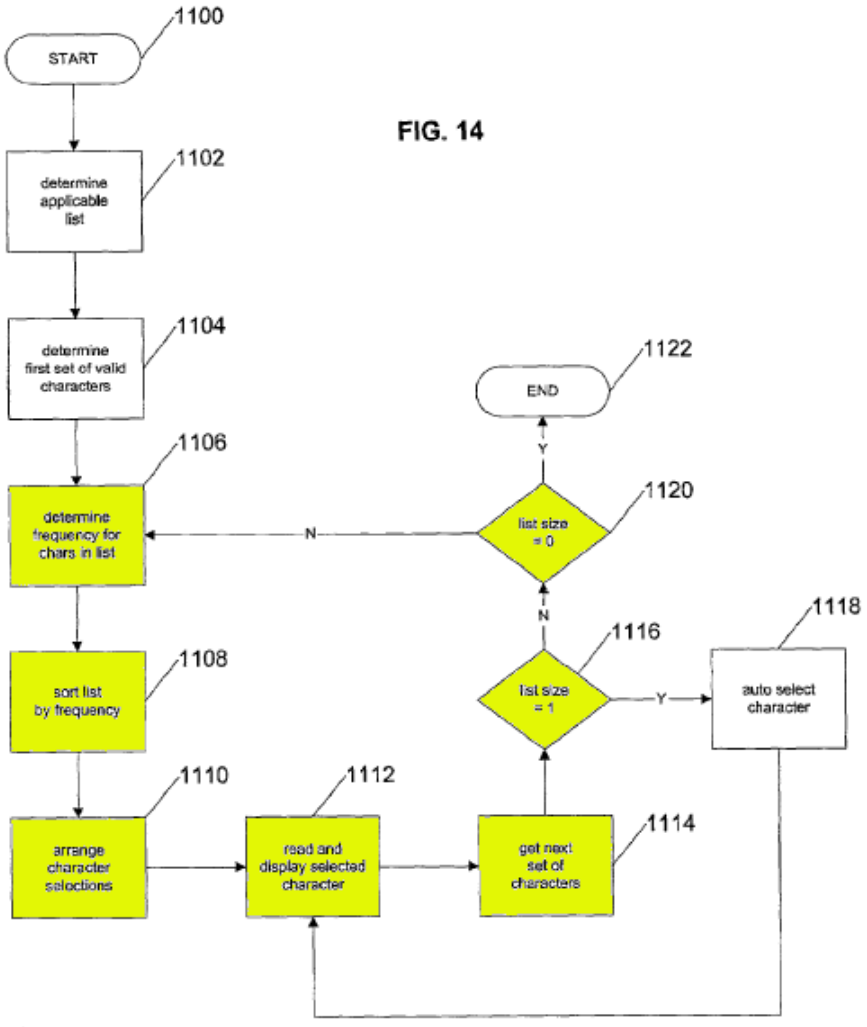
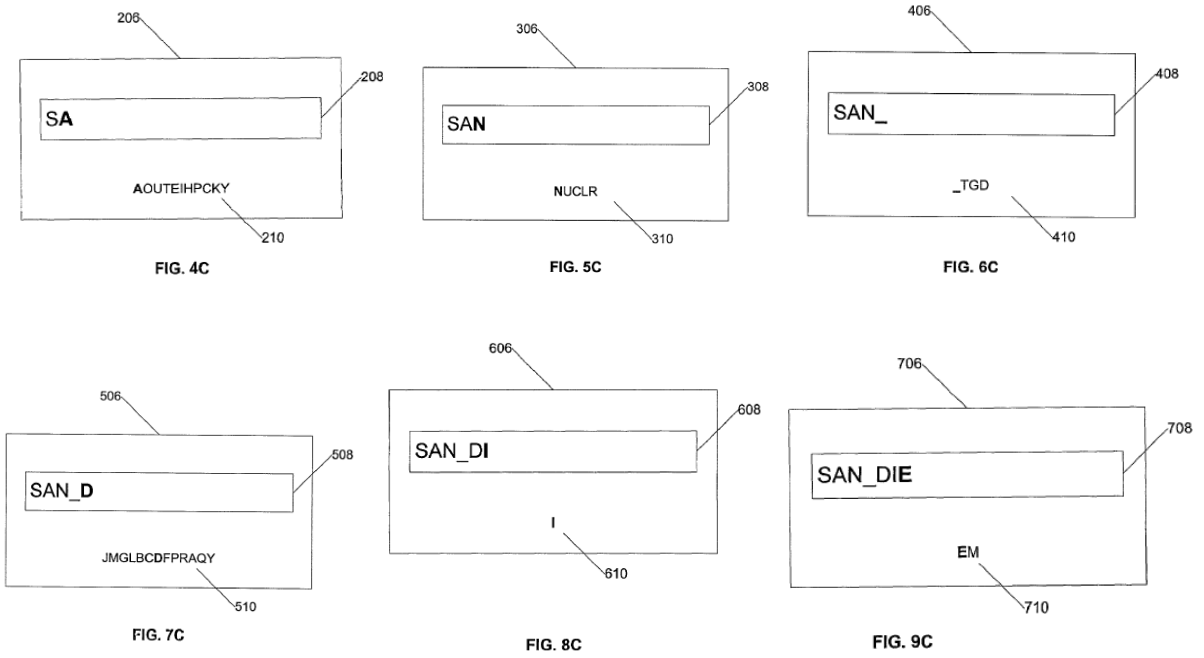


FIG. 14

Id., Fig. 14.

f) [1E]

Pu discloses this limitation. Forlines Decl. ¶342. As shown in Figures 4C, 5C, 6C, 7C, 8C, 9C, and 10, all of the portions are combined in the display area to create a larger portion of the item identifier corresponding to the "SAN DIEGO" entry in the database. Ex. 1006, 10:18-28.



Id., Figs. 4-9.

g) [1F]

Pu discloses this limitation. Forlines Decl. ¶343. As shown in the figures above, the selected portions are combined and displayed in the text field to form a larger portion of the city name at step 1112. Ex. 1006, 13:38-51 ("[T]he system reads and displays the selected character").

h) [1G]

Pu discloses this limitation. Forlines Decl. ¶344. The first portions and second portions are alphanumeric letters or characters within the English alphabet, which are text symbols, and the additional parts of the item identifiers are shorter than complete city names, as shown in the figures above. Ex. 1006, 14:16-24.

i) [1H]

Pu discloses this limitation. Forlines Decl. ¶¶345-47. As explained above, once a letter is selected, a new set of item identifiers is ranked (and displayed in ranked order) reflecting the valid second letter choices matching words in the database. Ex. 1006, 6:62-7:15, 8:15-45. Because each part of an item identifier (*i.e.*, letter) corresponds to the matching cities having that item identifier as a first letter, the first and second sets of item identifiers (which correspond to different letters in *Pu*) are mutually exclusive. As one simple example, the part 'S' shown in Figure 3A would correspond to the set of all cities starting with 'S', while the part 'C' would correspond to the set of all cities starting with 'C'. Since the city names cannot start with both 'S' and 'C', these two sets of item identifiers would be mutually exclusive.

Moreover, the third set of item identifiers and the fourth set of item identifiers are mutually exclusive subsets of the first set of item identifiers or the second set of item identifiers because each of those sets requires a first and second

letter to match. As the user enters each additional letter, the set of matching cities is reduced to a subset of the prior set. Forlines Decl. ¶346.

Dostie also discloses this limitation for the same reasons described above in connection with *Perlman*. *See supra*, §VIII(A)(1). It would have been obvious to incorporate *Dostie*'s tree structure into *Pu* for the same reasons described below in connection with claim 5 and because *Dostie* itself recognizes that use of a candidate tree helps "to rapidly predict potential completion candidates" and "provides a mechanism for supporting enhanced data entry techniques such as character prediction." Ex. 1011, ¶[0097]. The "enhanced data entry techniques" and "character prediction" are the same techniques described by *Pu*, so a POSITA would have found it natural to incorporate *Dostie*'s candidate tree into the character selection system of *Pu*. Forlines Decl. ¶347.

2. Claim 2

Pu discloses iterating the selections until the item identifiers are completed. Forlines Decl. ¶348. A user can continue selecting additional characters until the city name is completed, as shown in the flowchart of Figure 14 and the text fields shown above. Ex. 1006, 13:38-51.

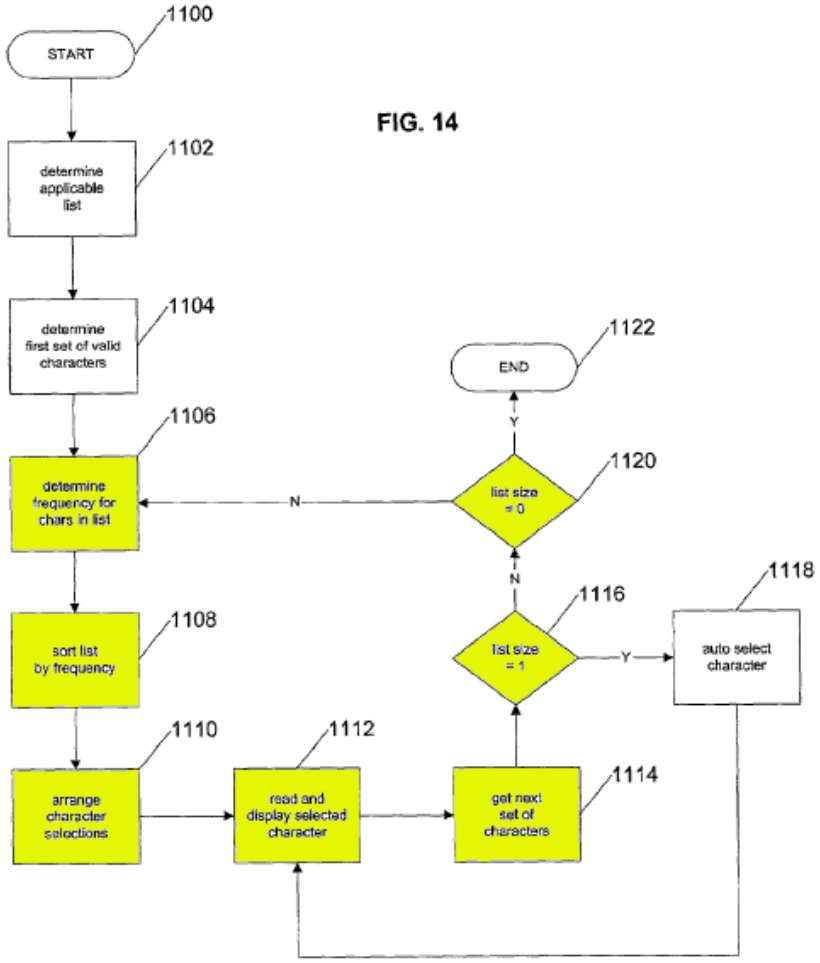


FIG. 14

Id., Fig. 14; *see also* Figs. 4-10.

3. Claim 3

Pu discloses selecting the parts based on predetermined criteria. Forlines Decl. ¶349. The selected two portions of item identifiers are selected based on "predefined lists," which may be subsets of larger lists (Ex. 1006, 12:42-13:3), and "relative frequency of each valid selection in the predefined list" (*id*, 2:53-57, 11:14-23, 11:55-65 (letters selected based on "relative frequencies and according

to the frequency mapping"))—both of which are examples of a predetermined criterion. Forlines Decl. ¶349.

4. Claim 4

Pu discloses that the predetermined criteria is a frequency of selection of database items. Forlines Decl. ¶350. As described above in connection with claims 1 and 3, the selected two portions of item identifiers are selected by "determining the relative frequency of each valid selection in the predefined list and presenting those valid selections with the highest frequency in a position that minimizes the number keystrokes required for data entry." Ex. 1006, 2:53-57. *See also id.*, 11:14-23, 11:55-65 (selecting portions based on "relative frequencies and according to the frequency mapping"); 11:14-23 (dynamically changing frequency mappings "in accordance with the most commonly used characters" so that the "most commonly used characters" are selected for display).

5. Claim 5

Pu in view of *Dostie* renders this claim obvious. Forlines Decl. ¶¶351-52. In *Pu*, the section list is continuously updated as the user inputs characters according to the flow chart of Figure 14. This process continues until a complete item identifier is inputted. Ex. 1006, 10:18-36. Because the process determines all possible next letters from the set of database entries matching the letters inputted so far, it would have been obvious in view of *Pu* alone to use a ranking in relation

to a tree hierarchy classification of the parts of the item identifiers as the predetermined criterion. Forlines Decl. ¶351. In other words, "because the system in *Pu* searches through all possible database items matching the first *N* characters where *N* corresponds to the letter position of the character currently being inputted by the user and displays matching titles in the database list, it would make intuitive sense to use a tree hierarchy classification (*i.e.*, one with levels and orders) based on the sequencing and ordering of letters within the matching database items." *Id.*

Dostie also discloses this limitation as discussed above. *See supra* §(VIII)(A)(5).

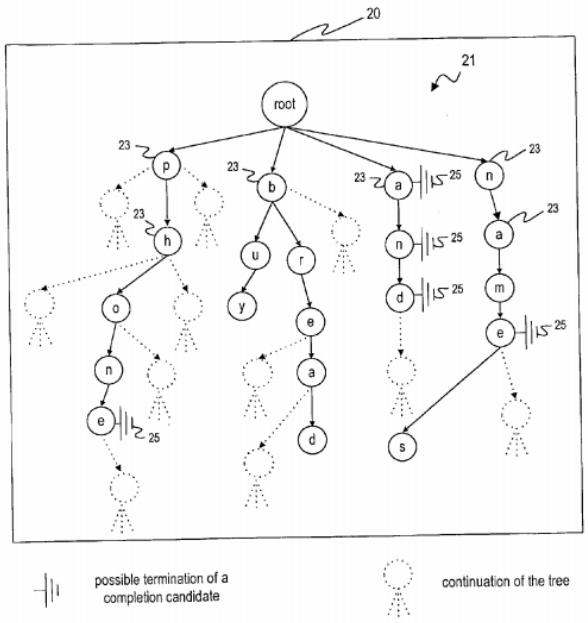


FIG. 4

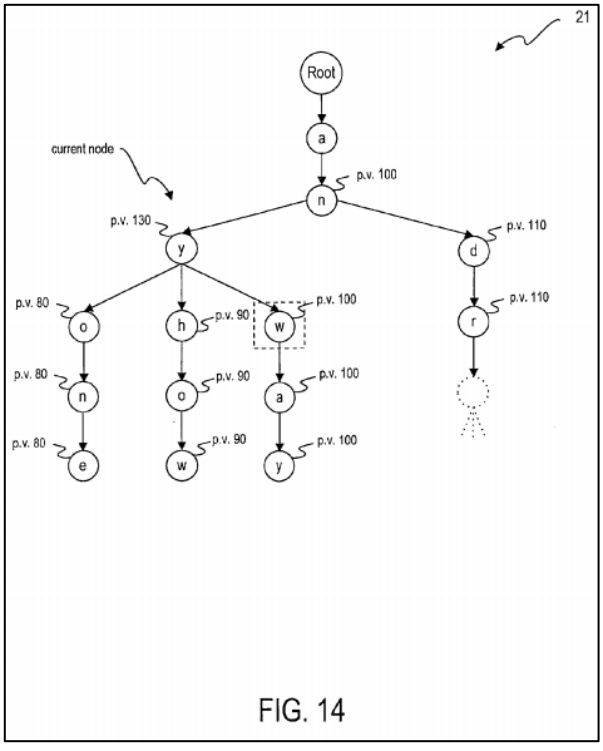


FIG. 14

Ex. 1011, ¶[0085], [0088]-[0090], Figs. 4 & 14. Therefore, to the extent rankings of the item identifiers in relation to a tree hierarchy classification was not already disclosed used in *Pu*, *Dostie* discloses such a classification system.

6. Claim 6

Pu discloses that the item identifiers are text strings. Forlines Decl. ¶353. In the example of Figure 10, the item identifier "SAN_DIEGO" is a text string and corresponds to a city name in California in the predefined list. Ex. 1006, 12:57-13:15.

7. Claim 7

Pu discloses that the item identifiers are words. Forlines Decl. ¶354. As described above in connection with claim 6, the item identifier "SAN_DIEGO" is a word that corresponds to a city name in California in the predefined list. Ex. 1006, 12:57-13:15.

8. Claim 8

Pu discloses that the database is stored in a memory of the computing device. Forlines Decl. ¶355. *Pu* explains that the databases used to store the predefined lists "may be locally located on a fixed storage drive such as a hard drive integrated with the auto PC or on a removable storage disc such as a CD." Ex. 1006, 5:18-25. *Pu* also explains that the predefined lists may be stored within the "storage capacity of the computer system or device" or "network centric

databases (e.g. web sites on the Internet)." Ex. 1006, 13:4-15. *See also id.*, 12:42-56 (noting that the "predefined list may be saved in a data storage area").

9. Claim 9

Pu discloses that the computing device is a mobile telephone. Forlines Decl. ¶356. *Pu* explains that its enhanced input interface is applicable to "intelligent devices" that may include "automobile computers ("auto PCs"), major home appliances, personal digital assistants ("PDAs"), telephones, cellular telephones, internet phones, pagers, portable computers, navigational devices, and the like." Ex. 1006, 1:15-26, 2:17-25, 5:1-10. A telephone keypad embodiment in combination with a dynamic display is described and shown in Figures 11-13 in *Pu*. *Id.*, 10:37-64.

10. Claim 10

Pu discloses that the computing device is a PDA. Forlines Decl. ¶357. *Pu* explains that its enhanced input interface is applicable to "intelligent devices" that may include "automobile computers ("auto PCs"), major home appliances, personal digital assistants ("PDAs"), telephones, cellular telephones, internet phones, pagers, portable computers, navigational devices, and the like." Ex. 1006, 1:15-26, 2:17-25, 5:1-10.

11. Claim 11

Pu discloses that the enabling selections allow selections by way of a joystick. Forlines Decl. ¶358. *Pu* uses a shuttle control system that is

"implemented using a single joystick-like central key that can be pivoted up, down, right or left." Ex. 1006, 1:62-64. It would be obvious to allow selections of the parts of item identifiers by way of a joystick because this was one well-known way of selecting letters for database input using a mobile device or gaming system. Forlines Decl. ¶358. In addition, the '354 Patent itself acknowledges that "a menu driven by a joystick" was "commonly found on a mobile phone" and allowed a user to select menu options using the four directions selectable by the joystick. Ex. 1001 at 6:23-27.

12. Claim 12

Claim 12 recites all the same elements as claim 1, but in system form. *Pu* discloses all the limitations of claim 12 for the same reasons as described above in connection with claim 1. Forlines Decl. ¶¶359-60.

Pu discloses an apparatus for selecting items, the apparatus comprising an output display, an input, and a computer processor. *See* Ex. 1006, 4:49-67 (cell phone and PC embodiments as control system 10 and including data entry device 40), 1:8-26 (describing portable computing devices such as PCs and PDAs which would have computer processors). The processor is configured to associate items with corresponding item identifiers as well as perform all the steps of claim 1. *Id.*, 5:11-50 (describing associating database content with characters in the name),

5:57-6:9 (describing tables associating predefined lists with first letter of each city name).

13. Claim 13

Claim 13 recites the same features as claim 3 and is obvious for the same reasons. Forlines Decl. ¶361.

14. Claim 14

Claim 14 recites the same features as claim 4 and is obvious for the same reasons. Forlines Decl. ¶362.

15. Claim 15

Claim 15 recites the same features as claim 5 and is obvious for the same reasons. Forlines Decl. ¶363.

16. Claim 16

Claim 16 recites "[t]he apparatus in accordance with claim 15, including a memory storing the tree hierarchy." *Pu* in view of *Dostie* renders this claim obvious. Forlines Decl. ¶364. *Dostie* discloses that its dictionary, which is organized in a tree hierarchy, is stored in data structure 29 within "memory available on the personal computing device concerned." Ex. 1011, ¶¶[0090]-[0091]. The most natural place to store the tree hierarchy would be in a memory. Forlines Decl. ¶364.

17. Claim 17

Claim 17 recites the same features as claim 6 and is obvious for the same reasons. Forlines Decl. ¶365.

18. Claim 18

Claim 18 recites the same features as claim 7 and is obvious for the same reasons. Forlines Decl. ¶366.

19. Claim 19

Claim 19 recites the same features as claim 8 and is obvious for the same reasons. Forlines Decl. ¶367.

20. Claim 20

Claim 20 recites the same features as claim 9 and is obvious for the same reasons. Forlines Decl. ¶368.

21. Claim 21

Claim 21 recites the same features as claim 10 and is obvious for the same reasons. Forlines Decl. ¶369.

22. Claim 22

Claim 22 recites the same features as claim 11 and is obvious for the same reasons. Forlines Decl. ¶370.

23. Motivation to Combine *Pu* and *Dostie*

It would have been obvious to incorporate *Dostie*'s dictionary trees (with mutually exclusive identifiers and subsets of identifiers) and rankings based on tree

hierarchy classifications into *Pu* for the same reasons described above in connection with *Perlman*. *See supra* §(VIII)(A)(17). Moreover, as described above, *Pu* already displays a sorted list of database items matching the first *N* letters of the user's input. At the time of *Pu*, this form of search was almost always performed using a hierarchical search tree. Forlines Decl. ¶371. Because of the well-known benefits of using hierarchical search trees to search a database, these trees would have been a trivial and routine addition to *Pu* that would have permitted simple and efficient searching based on a known prefix of database entries, like the searching discussed in *Pu*. *Id.*

D. Ground 4: Claims 1-22 are obvious in view of *Krohn*, *Dostie*, and *Pu*

1. Claim 1

a) Preamble

To the extent the preamble is considered limiting, *Krohn* discloses this limitation. Forlines Decl. ¶372. *Krohn* describes "a method and system for selecting a character with a user input device comprising a plurality of buttons." Ex. 1007, Abstract.

b) [1A]

Krohn in view of *Dostie* or *Pu* renders this claim obvious. Forlines Decl. ¶¶373-77. As shown in Figure 1, a plurality of buttons on a user input device 120 (*e.g.*, game controller) can be used to select a character displayed on the display

device 110. The display device is connected to processor 100 and presents an on-screen image of four characters ('J', 'P', 'G', and 'R') "arranged in a diamond pattern that corresponds to the diamond pattern of the four buttons 132, 134, 136, 138 on the user input device 120." Ex. 1007, 3:38-59.

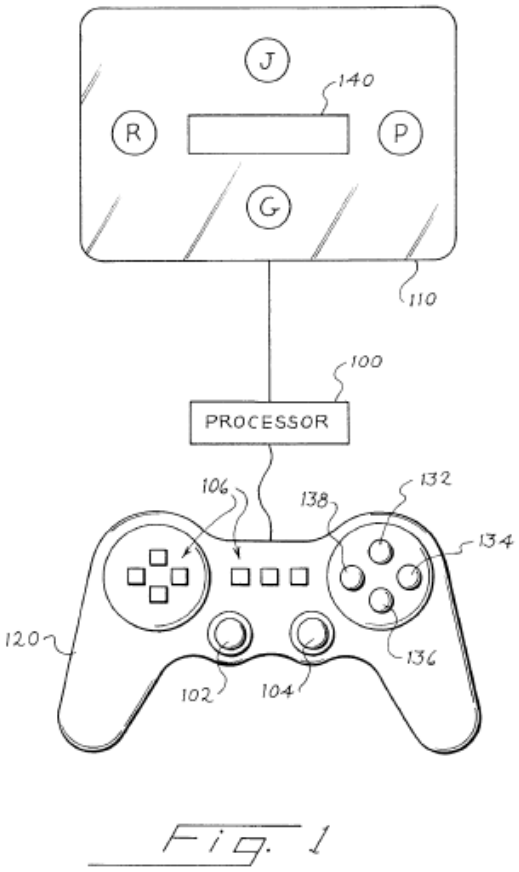


Fig. 1

Id., Fig. 1.

The portions of item identifiers are displayed in circles on display device 110 surrounding a character entry field 140. The portions of item identifiers are selected for display based on a "character prediction method to select the four characters from the character set that are the most likely to be selected or follow

the first selected character." *Id.*, 4:26-36. The item identifiers can also correspond to "predetermined words" known to processor 100. *Id.*, 4:37-54. The predetermined words can correspond to "a predetermined correct answer, a predetermined incorrect/wrong answer, or a random answer" in which case only valid letters (*i.e.*, matching the correct position of a predetermined word) are selected for display. *Id.*

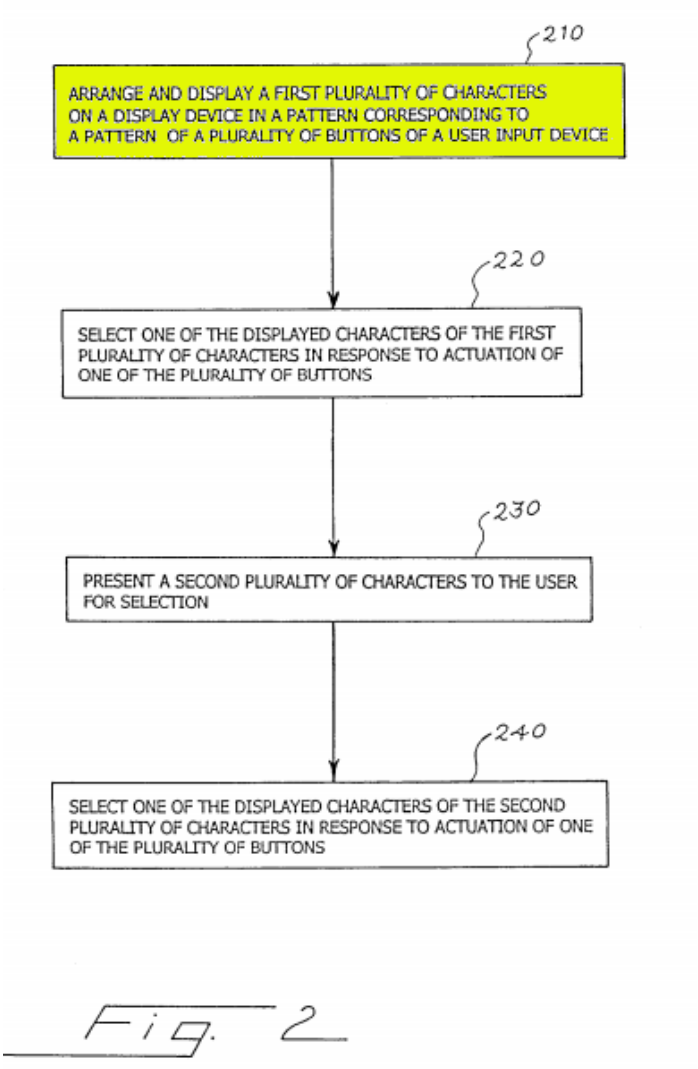


Fig. 2

Id., Fig. 2.

As shown in Figures 3A and 3B, the item identifiers can correspond to characters from correct answers or incorrect answers. *Id.*, 5:3-39. The characters of correct answers would be for a first set of items, while the characters for incorrect answers would be for a second set of items. In addition, each displayed character ('J', 'P', 'G', and 'R') in the example of Figure 1 would be for the set of all items starting with that character. As such, each displayed character would be a part of an item identifier for a different set of items. Forlines Decl. ¶375.

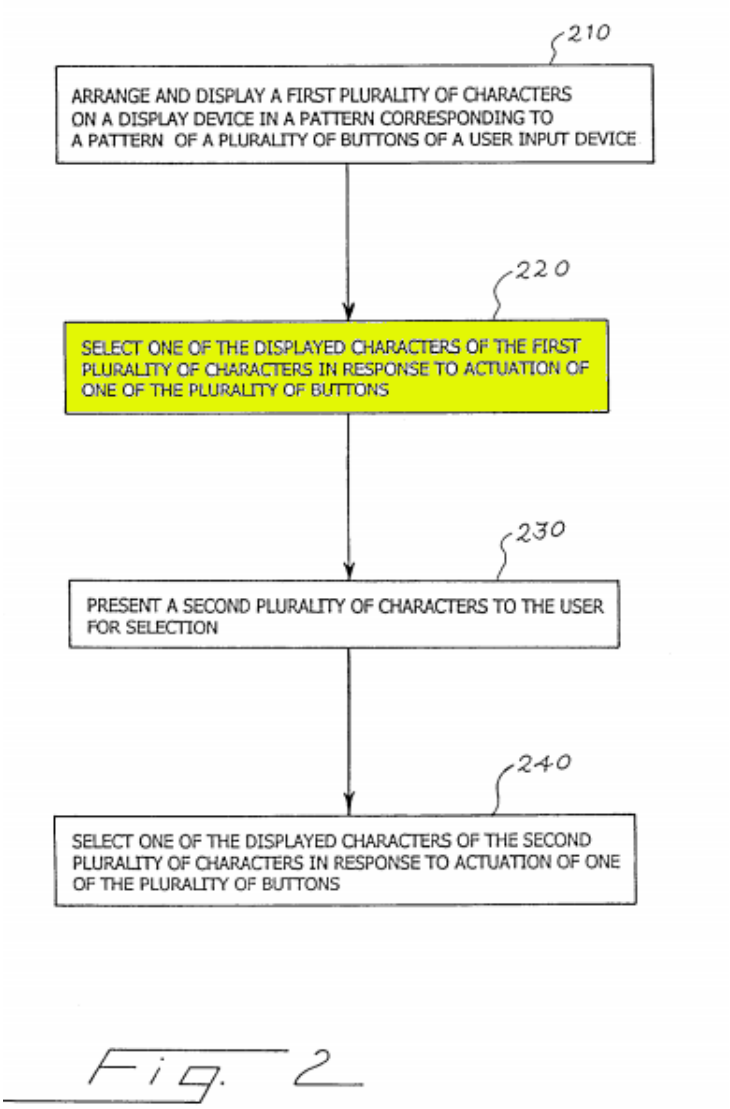
While *Krohn* is silent about how the predetermined words known to processor 100 are stored, it would have been obvious to store those words in a database, as taught by *Dostie* or *Pu*. As discussed above, *supra* §(VIII)(C)(1), in *Pu* the valid choices presented in the selection list are item identifiers corresponding to a plurality of items in a database. For example, the expected textual input could be a particular city within the states of California or Nevada. Ex. 1006, 5:11-17. In this example, the item identifiers correspond to entries in a "database comprising a predefined list of all cities within those two states." *Id.* Before the user enters any input into display area 108, selection list 110 includes item identifiers corresponding to all valid first letters of the cities in the database. *See id.*, Fig. 2. *Dostie* also describes using a "dictionary" (a form of databse) containing the completion candidates, where the dictionary/database size is

"limited only by the amount of memory available on the personal computing device." Ex. 1011, ¶[0090].

A database would have been the most natural storage location for the set of predetermined words because databases were routinely used to store sets of words (for example, dictionaries), names (for example, address books), lists of information, or any other type of content. Forlines Decl, ¶377. Moreover, a database would permit the use of well-known database searching techniques, which would simplify the character selection operations of processor 100 in *Krohn*. *Id.*

c) [1B]

Krohn discloses this limitation. Forlines Decl. ¶378. "[T]he processor 100 selects one of the displayed characters of the first plurality of characters in response to actuation of one of the plurality of buttons 132, 134, 136, 138 (act 220)." Ex. 1007, 3:48-51.

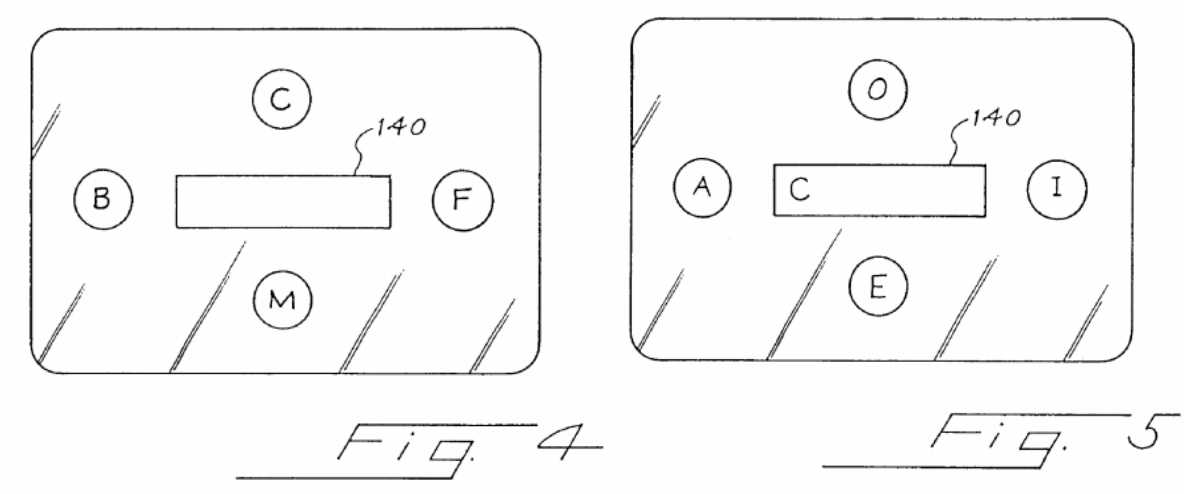


Id., Fig. 2.

d) [1C]

Krohn discloses this limitation. Forlines Decl. ¶379. The user's selection is displayed in character entry field 140 along with circles now displaying the additional parts of item identifiers. Ex. 1007, 3:57-59. As shown below, the previously selected portion of the item identifier (the character 'C') is displayed in

field 140, and additional parts (characters 'A', 'O,' 'E', and 'I') are displayed in the circles around field 140.



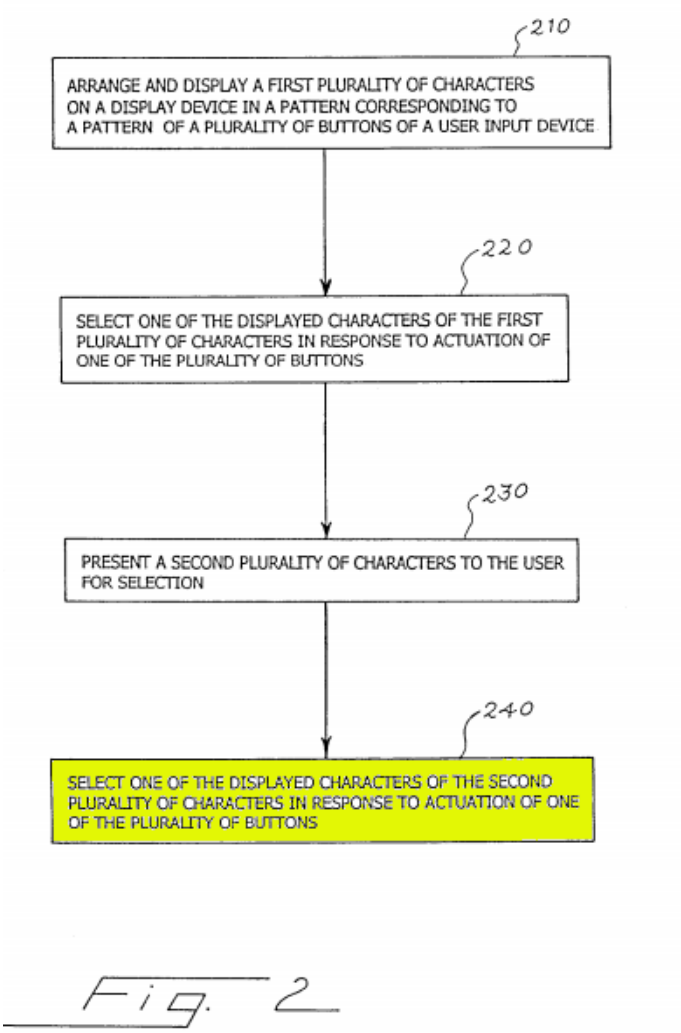
Id., Fig. 5.

The additional parts of an item identifiers shown in Figure 5 are for at least a third set of items and a fourth set of items. For example, part 'A' corresponds to the set of items containing 'CA' as the first two letters of the item name, while part 'O' corresponds to the set of items containing 'CO' as the first two letters of the item name. Forlines Decl. ¶380.

e) [1D]

Krohn discloses this limitation. Forlines Decl. ¶381. Just as the user selected a first character, a second character can be selected by the user "to allow the user to enter a series of characters, or a word." Ex. 1007, 4:9-25. "As with the first plurality of characters, the processor 100 selects one of the displayed

characters of the second plurality of characters in response to actuation of one of the plurality of buttons 132, 134, 136, 138 (act 240)." *Id.*



Id., Fig. 2.

f) [1E]

Krohn discloses this limitation. Forlines Decl. ¶382. The above method is repeated and selected characters are combined to permit the user to input "a series

of characters, or a word." Ex. 1007, 4:9-25. The next character "is placed at the end of the entry field 140 (act 370)." *Id.*, 6:24-25.

g) [1F]

Krohn discloses this limitation. Forlines Decl. ¶383. As explained above, the next character "is placed at the end of the entry field 140 (act 370)" and therefore displayed with the previously selected character. *Id.*, 6:24-25.

h) [1G]

Krohn discloses this limitation. Forlines Decl. ¶384. The first plurality of portions of item identifiers and the second plurality of portions of item identifiers are alphanumeric letters or characters within the English alphabet, which are text symbols representing the English language. Ex. 1007, 3:28-37. As shown above, the additional parts of the item identifiers are shorter than complete item identifiers because they are single letters or characters.

i) [1H]

Krohn discloses this limitation. Forlines Decl. ¶¶385-87. As explained above, once a letter is selected, a new set of item identifiers is displayed using a "character prediction method." Ex. 1011, 4:9-36. Because each part of an item identifier (*i.e.*, letter) corresponds to the matching "predetermined words" having that item identifier as a first letter, the first and second sets of item identifiers (which correspond to different letters in *Krohn*) are mutually exclusive. *Id.*, 4:37-54. As one simple example, the part 'C' shown in Figure 4 would correspond to

the set of all items ("predetermined words") starting with 'C', while the part 'B' would correspond to the set of all items starting with 'B'. Since the item names cannot start with both 'C' and 'B', these two sets of item identifiers would be mutually exclusive.

Moreover, the third set of item identifiers and the fourth set of item identifiers are mutually exclusive subsets of the first set of item identifiers or the second set of item identifiers because each of those sets requires a first and second letter to match. As the user enters each additional letter, the set of matching predetermined words is reduced to a subset of the prior set. Forlines Decl. ¶386.

Dostie also discloses this limitation for the same reasons described above in connection with *Perlman*. *See supra*, §VIII(A)(1). It would have been obvious to incorporate *Dostie*'s tree structure into *Krohn* for the same reasons described below in connection with claim 5 and because *Dostie* itself recognizes that use of a candidate tree helps "to rapidly predict potential completion candidates" and "provides a mechanism for supporting enhanced data entry techniques such as character prediction." Ex. 1011, ¶[0097]. The "enhanced data entry techniques" and "character prediction" are the same techniques described by *Krohn*, so a POSITA would have found it natural to incorporate *Dostie*'s candidate tree into the character selection system of *Krohn*. Forlines Decl. ¶387.

2. Claim 2

Krohn discloses iterating the selections until the item identifiers are completed. Forlines Decl. ¶388. A user can continue selecting additional characters "[t]o perform sequential character entry" until the word is completed, as shown in the flowchart of Figure 3. Ex. 1007, 4:9-25; Fig. 3.

3. Claim 3

Krohn discloses selecting the parts based on predetermined criteria. Forlines Decl. ¶389. The selected two portions of item identifiers are selected for display based on a "character prediction method to select the four characters from the character set that are the most likely to be selected or follow the first selected character." *Id.*, 4:26-36. The item identifiers can also correspond to "predetermined words" known to processor 100. *Id.*, 4:37-54. The predetermined words can correspond to "a predetermined correct answer, a predetermined incorrect/wrong answer, or a random answer" in which case only valid letters (*i.e.*, matching the correct position of a predetermined word) are selected for display. *Id.* In addition, the "most-likely" next characters can be selected for display (for example, the character 'U' after the character 'Q', see *id.*, 4:25-36) which is another form of "predetermined criteria." Forlines Decl. ¶389.

4. Claim 4

Krohn discloses that the predetermined criteria is a frequency of selection of database items. Forlines Decl. ¶390. As described above in connection with

claims 1 and 3, the selected two portions of item identifiers are selected based on matching predetermined words or being the "most-likely" next characters using a character prediction method, both of which indicate a "frequency of selection" of those items. Ex. 1007, 4:25-54.

To the extent *Krohn* does not disclose this feature, *Dostie* describes a "learning ability" of its dictionary to "adapt the rankings of completion candidates…as the data entry system 26 is used over time." Ex. 1011, ¶¶[0208]-[0209]. This permits the dictionary to learn new words automatically as well as increment the preference value for words each time they are typed. *Id.* It would have been obvious to use a frequency of selection of database items as the predetermined criteria, as taught by *Dostie*, in order for *Krohn*'s character prediction method to yield more relevant results that the user is actually interested in selecting. Forlines Decl. ¶391. In other words, a POSITA would have understood that items that were selected frequently in the past are likely to be selected again and that these words should be given higher priority than items selected less frequently. *Id.*

5. Claim 5

Krohn in view of *Dostie* renders obvious this limitation. Forlines Decl. ¶¶392-93. In *Krohn*, a user is allowed to perform sequential character entry until a complete item identifier is inputted. Ex. 1007, 4:9-25. Because the process

determines all possible next letters from the set of predetermined words, it would have been obvious in view of *Krohn* alone to use a ranking in relation to a tree hierarchy classification of the parts of the item identifiers as the predetermined criterion. Forlines Decl. ¶392. In other words, "because the system in *Krohn* searches through all possible database items matching the first *N* characters where *N* corresponds to the letter position of the character currently being inputted by the user, it would make intuitive sense to use a hierarchical classification (*i.e.*, one with levels and orders) based on the sequencing and ordering of letters within the matching database items." *Id.*

Dostie also discloses this limitation as discussed above. *See supra* §(VIII)(B)(1).

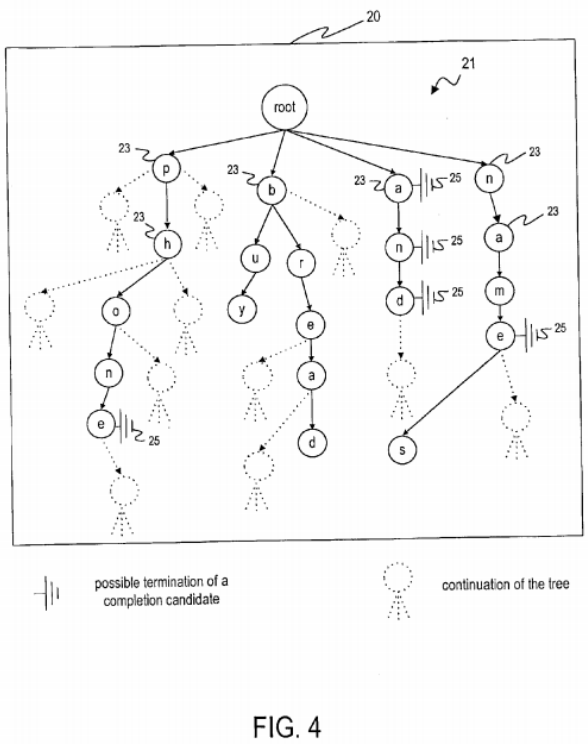


FIG. 4

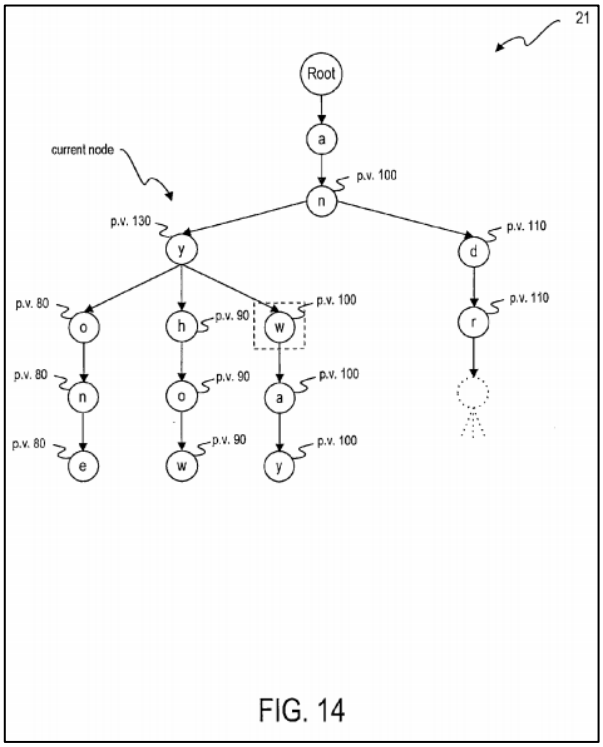


FIG. 14

Ex. 1011, ¶[0085], [0088]-[0090], Figs. 4 & 14. Therefore, to the extent rankings of the item identifiers in relation to a hierarchical classification was not already disclosed used in *Krohn*, *Dostie* discloses such a classification system.

6. Claim 6

Krohn discloses that the item identifiers are text strings. Forlines Decl. ¶394. *Krohn* provides several examples of item identifiers, including predetermined words, like HELLO, and predetermined answers, such as "Cadillac," "Camry," "Ford," and "Mercury," each of which is a text string. Ex. 1007, 4:37-54.

7. Claim 7

Krohn discloses that the item identifiers are words. Forlines Decl. ¶395. As described above in connection with claim 6, the predetermined words and predetermined answers are all words. Ex. 1007, 4:37-54.

8. Claim 8

Krohn in view of *Pu* renders this claim obvious. Forlines Decl. ¶396. *Krohn* already discloses that processor 100 (which accesses the predetermined words) can be local to a computing device (such as a game console, television set-top box, multimedia player, or general purpose computer) or "in a location remote from the display device." Ex. 1007, 3:8-27.

Pu explains that the databases used to store its predefined lists "may be locally located on a fixed storage drive such as a hard drive integrated with the auto PC or on a removable storage disc such as a CD." Ex. 1006, 5:18-25. *Pu* also explains that the predefined lists may be stored within the "storage capacity of the computer system or device" or "network centric databases (e.g. web sites on the Internet)." Ex. 1006, 13:4-15. *See also id.*, 12:42-56. In view of this disclosure in *Pu*, it would have been natural to modify *Krohn* to store its predetermined words in a memory of the computing device in view of the well-known benefits of doing so. Forlines Decl. ¶397.

9. Claim 9

Krohn in view of *Pu* renders this claim obvious. Forlines Decl. ¶398. *Pu* explains that its enhanced input interface is applicable to "intelligent devices" that may include "automobile computers ("auto PCs"), major home appliances, personal digital assistants ("PDAs"), telephones, cellular telephones, internet phones, pagers, portable computers, navigational devices, and the like." Ex. 1006, 1:15-26, 2:17-25, 5:1-10. A telephone keypad embodiment in combination with a dynamic display is described and shown in Figures 11-13 in *Pu*. *Id.*, 10:37-64. It would have been obvious to apply *Krohn*'s input interface to a mobile telephone. Forlines Decl. ¶398.

10. Claim 10

Krohn in view of *Pu* renders this claim obvious. Forlines Decl. ¶399. *Pu* describes an enhanced user interface for intelligent devices and explains that "[s]uch intelligent devices include automobile computers ('auto PCs'), major home appliances, personal digital assistants ('PDAs'), telephones, cellular telephones, internet phones, pagers, portable computers, navigational devices, and the like." Ex. 1006, 1:19-23, 5:1-10. It would have been obvious to apply *Krohn*'s input interface to a PDA. Forlines Decl. ¶399.

11. Claim 11

Krohn discloses that the enabling selections allow selections by way of a joystick. Forlines Decl. ¶400. *Krohn* explains that "[i]n addition to the plurality of buttons 132, 134, 136, 138 used to select a displayed character, the user input device 120 can also comprise additional user interface elements such as joysticks 102, 104 and additional buttons 106." Ex. 1007, 2:50-3:7. In addition, gaming consoles with joysticks are disclosed in a preferred embodiment. *Id.*, 3:8-11.

12. Claim 12

Claim 12 recites all the same elements as claim 1, but in system form. *Krohn* discloses all the limitations of claim 12 for the same reasons as described above in connection with claim 1. Forlines Decl. ¶¶401-02.

Krohn discloses an apparatus for selecting items, the apparatus comprising an output display, an input, and a computer processor. *See* Ex. 1007, 2:34-49 (processor 100 coupled to display device 110 and user input device 120). The processor is configured to associate items with corresponding item identifiers as well as perform all the steps of claim 1. *Id.*, 3:38-44, 4:29-54 (describing processor as associating predetermined words with characters in the words and displaying characters using "character prediction method").

13. Claim 13

Claim 13 recites the same features as claim 3 and is obvious for the same reasons. Forlines Decl. ¶403.

14. Claim 14

Claim 14 recites the same features as claim 4 and is obvious for the same reasons. Forlines Decl. ¶404.

15. Claim 15

Claim 15 recites the same features as claim 5 and is obvious for the same reasons. Forlines Decl. ¶405.

16. Claim 16

Claim 16 recites "[t]he apparatus in accordance with claim 15, including a memory storing the tree hierarchy." *Pu* in view of *Dostie* renders this claim obvious. Forlines Decl. ¶406. *Dostie* discloses that its dictionary, which is organized in a tree hierarchy, is stored in data structure 29 within "memory

available on the personal computing device concerned." Ex. 1011, ¶¶[0090]-[0091]. The most natural place to store the tree hierarchy would be in a memory. Forlines Decl. ¶406.

17. Claim 17

Claim 17 recites the same features as claim 6 and is obvious for the same reasons. Forlines Decl. ¶407.

18. Claim 18

Claim 18 recites the same features as claim 7 and is obvious for the same reasons. Forlines Decl. ¶408.

19. Claim 19

Claim 19 recites the same features as claim 8 and is obvious for the same reasons. Forlines Decl. ¶409.

20. Claim 20

Claim 20 recites the same features as claim 9 and is obvious for the same reasons. Forlines Decl. ¶410.

21. Claim 21

Claim 21 recites the same features as claim 10 and is obvious for the same reasons. Forlines Decl. ¶411.

22. Claim 22

Claim 22 recites the same features as claim 11 and is obvious for the same reasons. Forlines Decl. ¶412.

23. Motivation to Combine *Krohn* and *Pu*

It would have been obvious to incorporate *Pu*'s database located in a variety of different locations and types of computing devices into the system of *Krohn*. *Krohn*'s improved user input interface would have benefitted each of the "intelligent devices" described by *Pu* in the same way (by reducing the number of button or key actuations to input an entire word corresponding to a database item). Since there is no technical difficulty in porting the input interface of *Krohn* to any type of input device, the choice of environment and type of device would have been mere design choices well within the grasp of a POSITA. Forlines Decl. ¶413. Moreover, the use of a local or remote database would have also been a trivial and well understood modification to *Krohn* that yielded extremely predictable results. In essence, all the features relied upon in *Pu* (and combined with *Krohn*) are commonplace and routine substitutions well within the knowledge of a POSITA. Forlines Decl. ¶413.

24. Motivation to Combine *Krohn, Pu,* and *Dostie*

It would have been obvious to incorporate *Dostie*'s dictionary trees (with mutually exclusive identifiers and subsets of identifiers) and rankings based on hierarchical classifications into *Krohn* and *Pu* for the same reasons described above. *See supra* §(VIII)(A)(17). Moreover, as described above, *Krohn* already searches for predetermined words matching the first *N* letters of the user's input.

At the time of *Krohn*, this form of search was almost always performed using a hierarchical search tree. Forlines Decl. ¶414. Because of the well-known benefits of using hierarchical search trees to search a database, these trees would have been a trivial and routine addition to *Krohn* that would have permitted simple and efficient searching based on a known prefix of database entries, like the searching discussed in *Krohn*. *Id.* This is especially true since *Krohn* would have "likely used hierarchical search trees as these were in standard use for database searching at the time." *Id.*

IX. SECONDARY CONSIDERATIONS

Patent Owner has not identified any evidence of secondary considerations, and Petitioner is not aware of any such evidence. Petitioner reserves the right to present rebuttal evidence if and when the Patent Owner presents such evidence.

X. DISCRETION UNDER §314(A)

Institution of this Petition would be an efficient use of Board resources at least because the court in the underlying litigation has not yet set a schedule, a trial date, or even held a status or scheduling conference—meaning discovery has not commenced. Ex. 1012. *See NHK Spring Co., Ltd. v. Intri-Plex Technologies, Inc.*, Case IPR2018-00752, Paper 8 (PTAB Sept. 12, 2018) (precedential).

XI. CONCLUSION

Petitioner has established a reasonable likelihood that the Challenged Claims are unpatentable. Petitioner therefore respectfully requests that *inter partes* review of the '852 Patent be granted.

Respectfully submitted,

Date: March 17, 2020

/s/ Brian E. Mack
Brian Mack (Reg. No. 57189)

CERTIFICATION UNDER 37 C.F.R. § 42.24

Under the provisions of 37 C.F.R. § 42.24, the undersigned hereby certifies that the word count for the foregoing Petition for *inter partes* review (excluding the table of contents, table of authorities, mandatory notices, certificate of service or word count, and appendix of exhibits or claim listing) totals 13,984 words, which is less than the 14,000 words allowed under 37 C.F.R. § 42.24(a)(i).

Date: March 17, 2020

/s/ Brian E. Mack
Brian Mack (Reg. No. 57189)

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. §§ 42.6(e), 42.105(a), the undersigned hereby certifies service on the Patent Owner of a copy of this Petition and its respective exhibits at the official correspondence address for the attorney of record for the '852 Patent as shown in USPTO PAIR via EXPRESS MAIL:

JONES DAY
250 VESEY STREET
NEW YORK NY 10281-1047

Date: March 17, 2020

/s/ Brian E. Mack
Brian Mack (Reg. No. 57189)