UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/743,476	12/23/2003	Jong-Goo Lee	678-1264	9615

66547 7590 01/13/2012

THE FARRELL LAW FIRM, P.C.
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
LO, WEILUN	
ART UNIT	PAPER NUMBER
2179	
MAIL DATE	DELIVERY MODE
01/13/2012	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte JONG-GOO LEE, EYAL TOLEDANO, NATAN LINDER, YARIV EISENBERG, and RAN BEN-YAIR

Appeal 2009-012781
Application 10/743,476[1]
Technology Center 2100

Before JEFFREY S. SMITH, KALYAN K. DESHPANDE, and MICHAEL R. ZECHER, *Administrative Patent Judges*.

ZECHER, *Administrative Patent Judge*.

DECISION ON APPEAL

---

[1] Filed on December 23, 2003. This application claims priority to provisional application 60/500,669, filed on September 5, 2003. The real party in interest is Samsung Electronics Co. Ltd. App. Br. 1.

I. STATEMENT OF THE CASE

Appellants appeal under 35 U.S.C. § 134(a) (2002) from the Examiner's rejection of claims 1-7 and 109-141. App. Br. 2. Claims 8-108 and 142-179 have been cancelled. *Id.* We have jurisdiction under 35 U.S.C. § 6(b) (2008).

We affirm.

Appellants' Invention

Appellants invented a system and method directed to a proactive user interface for use with mobile information devices. Spec. 1: 6-7.[2]

Illustrative Claim

> 1. A proactive user interface for a computational device, the computational device having an operating system, comprising:
>
> (a) an interface unit for communicating between a user and said operating system; and
>
> (b) a learning module for detecting at least one pattern of interaction of the user with said interface unit and for proactively altering at least one function of said interface unit according to said detected pattern.

Prior Art Relied Upon

Hoffberg	US 6,400,996 B1	June 4, 2002

Rejection on Appeal

Claims 1-7 and 109-141 stand rejected under 35 U.S.C. § 102(b) as being anticipated by Hoffberg. Ans. 2-5.

---

[2] All references to the Specification are to the clean version of the Specification entered January 24, 2005.

Appellants' Contentions

Appellants contend that Hoffberg's figure 15, which illustrates a flow diagram of a predictive user interface, does not describe proactive altering according to a detected pattern, as required by independent claims 1 and 122. App. Br. 11. In particular, Appellants argue that Hoffberg's disclosure of providing frequently used choices for program selections fails to describe "proactively altering [of] at least one function of said interface unit according to said detected pattern," as claimed. *Id.* In response to the Answer, Appellants allege that the Examiner provides new citations from Hoffberg that improperly seek to combine different embodiments in order to formulate the anticipation rejection. Reply Br. 2-3. Further, Appellants allege that the Examiner's anticipation rejection is improper because the Examiner cites to multiple references. App. Br. 12-13; Reply Br. 3.

Examiner's Findings and Conclusions

The Examiner finds that Hoffberg's cited disclosure, namely column 42, lines 20-67, column 50, lines 50-67, and column 85, lines 5-67, teaches or suggests the disputed claim limitation. Ans. 3, 6-10. In particular, the Examiner finds that Hoffberg discloses an adaptive interface that predicts the desired user function by monitoring the user's history, interface context, and machine status, and then uses the predicted function to change the interface accordingly. *Id.* at 6-7. Further, the Examiner disagrees with Appellants' allegation that the anticipation rejection is improper because the Examiner cites to multiple references. *Id.* at 11-12. The Examiner maintains that the anticipation rejection is not based on multiple references, but rather is based solely upon the teachings of Hoffberg as a single reference. *Id.* at 11.

II. ISSUE

Did the Examiner err in finding that Hoffberg anticipates independent claim 1? In particular, the issue turns on whether:

(a) Hoffberg describes "proactively altering at least one function of said interface unit according to said detected pattern," as recited in independent claim 1, and similarly recited in independent claim 122; and

(b) the Examiner improperly relied upon multiple references in making the anticipation rejection.

III. FINDINGS OF FACT ("FF")

Hoffberg

FF 1. Hoffberg discloses signal analysis and complex pattern recognition. In particular, Hoffberg discloses analyzing any data set presented to a system either internally, via a user interface, or through the environment in which the system operates. Further, Hoffberg discloses that pattern recognition involves examining a complex data set in order to determine similarities with other data sets, typically data sets which have been previously characterized. Col. 10, ll. 15-25.

FF 2. Hoffberg discloses an adaptive user interface that changes in response to context, past history, and the status of the corresponding system. In particular, Hoffberg discloses that the user interface may provide a model of the user, which is employed in a predictive algorithm. Hoffberg discloses that the model parameters may be static or dynamic, and may be adaptive to the user or alterations in the user pattern. Abstract; Col. 50, ll. 53-62.

FF 3. Hoffberg's figure 15 illustrates a flow diagram of a predictive user interface. Col. 83, ll. 15-16; Fig. 15.

IV. ANALYSIS

Claims 1 and 122

We do not find error in the Examiner's anticipation rejection of independent claims 1 and 122. In particular, independent claim 1 recites, *inter alia*, "proactively altering at least one function of said interface unit according to said detected pattern."

At the outset, we adopt the Examiner's findings as our own. Ans. 3, 6-11; *see also* FFs 1-3. In particular, we agree with the Examiner that Hoffberg's disclosure of an adaptive user interface that predicts a desired user function by monitoring the user's history, interface context, and system status, in conjunction with using the predicted function to alter the user interface accordingly (FFs 2 and 3), describes the disputed claim limitation. Ans. 6-7. Moreover, we are not persuaded by Appellants' argument that the Examiner improperly combines different embodiments to formulate the anticipation rejection. Reply Br. 2-3. We note that the Examiner provides multiple citations from Hoffberg's disclosure, including the Background of the Invention section (FF 1), the Summary and Objects of the Invention section (FF 2), and the embodiment illustrated in figure 15. FF 3. However, we find that these cited portions of Hoffberg's disclosure, namely the complex pattern recognition and predictive algorithm (FFs 1 and 2), highlight the important features implemented by the predictive user interface illustrated in figure 15. FF 3.

Further, we are not persuaded by Appellants' argument that the Examiner's anticipation rejection is improper because the Examiner cites to multiple references. App. Br. 12-13; Reply Br. 3. We agree with the Examiner that the anticipation rejection was not based upon citations from multiple references, but rather was based solely upon Hoffberg's disclosure. *See* Ans. 11. Consequently, we find that the Examiner properly relied only upon Hoffberg to make the anticipation rejection.

Nonetheless, it is well settled law that multiple references may be used in the context of an anticipation rejection to show how an ordinarily skilled artisan would have understood words and phrases used in the anticipating reference. *In re Baxter Travenol Labs.*, 952 F.2d 388, 390 (Fed. Cir. 1991) (Extrinsic evidence may be used to explain, but not expand, the meaning of terms and phrases in an anticipatory reference.); *In re Samour*, 571 F.2d 559, 562-63 (CCPA 1978) ("[T]he key issue before us is whether the PTO, in making a rejection under 35 USC 102(b) on a single prior art reference that discloses every material element of the claimed subject matter, can properly rely on additional references for such purpose. We hold in the affirmative."). In this case, the Examiner's reference to Hoffberg's disclosure in column 42, lines 20-67, not only indicates the patents properly incorporated by reference, but also amounts to extrinsic evidence that explains the meaning of pattern recognition functions with respect to Hoffberg's predictive user interface.[3] It follows that the Examiner has not erred in finding that Hoffberg anticipates independent claims 1 and 122.

[3] A § 102 rejection over multiple references is proper to explain the meaning of a term used in the primary reference. *See* Manual of Patent Examining Procedure § 2131.01.

Claims 2-7, 109-121, and 123-141

Appellants do not provide separate and distinct arguments for patentability with respect to dependent claims 2-7, 109-121, and 123-141. *See* App. Br. 11-13; Reply Br. 2-3. Therefore, we select independent claims 1 and 122 as representative of these aforementioned claims. *See* 37 C.F.R. § 41.37(c)(1)(vii). Consequently, we find that the Examiner has not erred in rejecting dependent claims 2-7, 109-121, and 123-141 for the same reasons set forth in our discussion of independent claims 1 and 122.

V. CONCLUSION OF LAW

The Examiner has not erred in rejecting claims 1-7 and 109-141 as being anticipated under 35 U.S.C. § 102(b).

VI. DECISION

We affirm the Examiner's decision to reject claims 1-7 and 109-141.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED

msc

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Jong-Goo LEE et al.

Serial No.: 10/743,476 Docket: 678-1264

Filed: December 23, 2003 Dated: November 3, 2010

For: PROACTIVE USER INTERFACE

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT

Sir:

Pursuant to Applicant's duty of disclosure, it is respectfully requested that the each item of information listed in the attached form PTO-1449 be considered by the Examiner and made of record in the above-identified application. Copies of the listed items are attached hereto.

A Final Office Action was mailed on February 7, 2008 in connection with the above-identified patent application. This application is currently on appeal.

The listed items were cited by the Canadian Patent Office in a counterpart application, namely Appln. No. 2,540,397. A copy of the Canadian Office Action dated August 11, 2010 is attached hereto.

The filing of this Information Disclosure Statement is not an admission that the information cited herein is, or is considered to be, material to patentability as defined in 37 C.F.R. § 1.56(b).

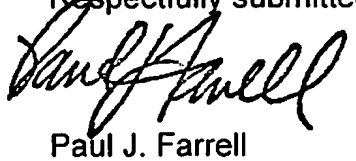
CERTIFICATION UNDER 37 C.F.R. §1.97(e)(1)

Applicant submits that each item of information contained in the attached Supplemental Information Disclosure Statement was first cited in any communication from a foreign patent office in a counterpart foreign application not more than three months prior to the filing of the Information Disclosure Statement.

The $180.00 surcharge will be paid via credit card.

The Director of the USPTO is hereby authorized to charge any additional fees which may become due in connection with this filing or credit any overpayment to Deposit Account No. 50-4053.

Respectfully submitted,



Paul J. Farrell
Reg. No. 33,494
Attorney for Applicant

THE FARRELL LAW FIRM, LLP
290 Broadhollow Road, Suite 210E
Melville, NY 11747
(516) 228-3565

PTO/SB/08a (07-09)
Approved for use through 07/31/2012. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 1 of 1

Complete if Known	
Application Number	10/743,476
Filing Date	December 23, 2003
First Named Inventor	Jong-Goo LEE et al.
Art Unit	
Examiner Name	
Attorney Docket Number	678-1264 (P11404)

U. S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document Country Code[3]-Number[4]-Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages Or Relevant Figures Appear	T[6]
		CA 2 536 233	03-17-2005			✓
		WO 2005/024649	03-17-2005			✓
		WO 03/073417	09-04-2003			✓

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. [1] Applicant's unique citation designation number (optional). [2] See Kinds Codes of USPTO Patent Documents at www.uspto.gov or MPEP 901.04. [3] Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language Translation is attached.

(19) Canadian Intellectual Property Office
An Agency of Industry Canada

Office de la Proprit Intellectuelle du Canada
Un organisme d'Industrie Canada

(11) CA 2 536 233 (13) A1

(40) 17.03.2005
(43) 17.03.2005

(12)

(21) 2 536 233

(22) 06.09.2004

(51) Int. Cl.:
G06F 3/048 (2006.01)
G06F 15/18 (2006.01)
G06N 3/12 (2006.01)
G06N 5/00 (2006.01)

(85) 17.02.2006

(86) PCT/KR04/002256

(87) WO05/024649

(30) 60/500,669 US 05.09.2003
10/743,476 US 23.12.2003
10-2004-0016266 KR 10.03.2004
10-2004-0016263 KR 10.03.2004
10-2004-0067663 KR 27.08.2004

(71) SAMSUNG ELECTRONICS CO., LTD.,
416 Maetan-dong
Yeongtong-gu
Suwon-shi
442-742, GYEONGGI-DO , XX (KR).

(72) EISENBERG, YARIV (IL).
LINDER, NATAN (IL).
BEN-YAIR, RAN (IL).
TOLEDANO, EYAL (IL).
LEE, JONG-GOO (KR).

(74) MARKS & CLERK

(54) INTERFACE UTILISATEUR PROACTIVE COMPRENANT UN AGENT EVOLUTIF
(54) PROACTIVE USER INTERFACE INCLUDING EVOLVING AGENT

Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2536233 A1 2005/03/17

(21) 2 536 233

(12) DEMANDE DE BREVET CANADIEN
CANADIAN PATENT APPLICATION

(13) A1

(86) Date de dépôt PCT/PCT Filing Date: 2004/09/06
(87) Date publication PCT/PCT Publication Date: 2005/03/17
(85) Entrée phase nationale/National Entry: 2006/02/17
(86) N° demande PCT/PCT Application No.: KR 2004/002256
(87) N° publication PCT/PCT Publication No.: 2005/024649
(30) Priorités/Priorities: 2003/09/05 (US60/500,669);
2003/12/23 (US10/743,476);
2004/03/10 (KR10-2004-0016266);
2004/03/10 (KR10-2004-0016263);
2004/08/27 (KR10-2004-0067663)

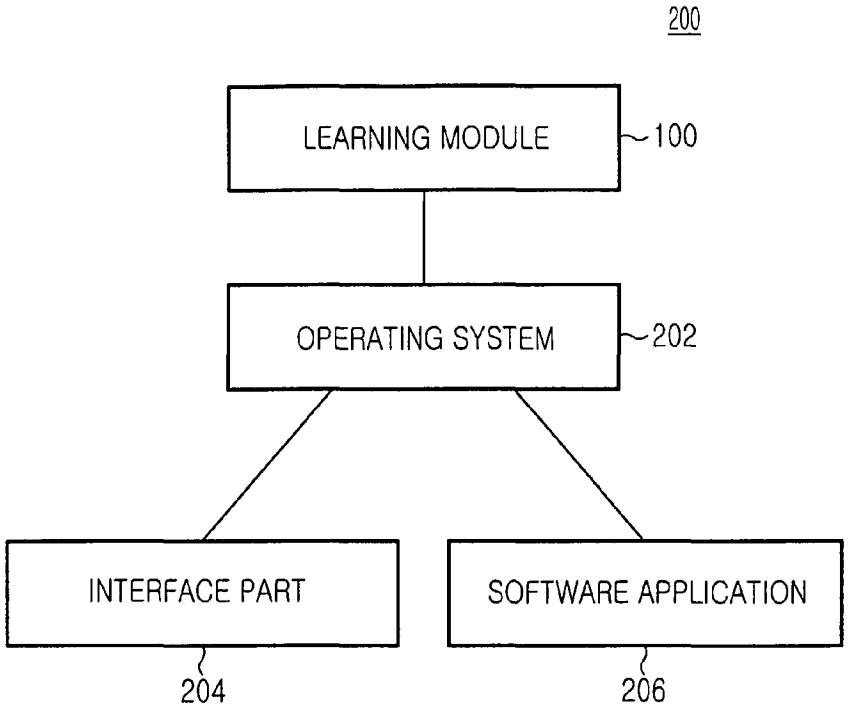
(51) Cl.Int./Int.Cl. *G06F 3/048* (2006.01),
G06N 5/00 (2006.01), *G06N 3/12* (2006.01),
G06F 15/18 (2006.01)

(71) Demandeur/Applicant:
SAMSUNG ELECTRONICS CO., LTD., KR

(72) Inventeurs/Inventors:
LEE, JONG-GOO, KR;
TOLEDANO, EYAL, IL;
LINDER, NATAN, IL;
BEN-YAIR, RAN, IL;
EISENBERG, YARIV, IL

(74) Agent: MARKS & CLERK

(54) Titre : INTERFACE UTILISATEUR PROACTIVE COMPRENANT UN AGENT EVOLUTIF
(54) Title: PROACTIVE USER INTERFACE INCLUDING EVOLVING AGENT



(57) Abrégé/Abstract:
A proactive user interface, installed in (or otherwise control and/or be associated with) any type of computational device. The proactive user interface actively makes suggestions to the user, based upon prior experience with a particular user and/or various preprogrammed patterns from which the computational device could select, depending upon user behavior. These suggestions can be made by altering the appearance of at least a portion of the display, for example by changing a menu or a portion thereof; providing different menus for display; and/or altering touch screen functionality. The suggestions can also be made audibly.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
17 March 2005 (17.03.2005)

PCT

(10) International Publication Number
WO 2005/024649 A1

(51) International Patent Classification[7]: G06F 15/18

(21) International Application Number:
PCT/KR2004/002256

(22) International Filing Date:
6 September 2004 (06.09.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/500,669	5 September 2003 (05.09.2003)	US
10/743,476	23 December 2003 (23.12.2003)	US
10-2004-0016266	10 March 2004 (10.03.2004)	KR
10-2004-0016263	10 March 2004 (10.03.2004)	KR
10-2004-0067663	27 August 2004 (27.08.2004)	KR

(71) Applicant *(for all designated States except US)*: **SAMSUNG ELECTRONICS CO., LTD.** [KR/KR]; 416, Maetan-dong, Yeongtong-gu, Suwon-si, Gyeonggi-do 442-742 (KR).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **LEE, Jong-Goo** [KR/KR]; 23/1, 360-65, Hongje-dong, Seodaemun-gu, Seoul 120-860 (KR). **TOLEDANO, Eyal** [IL/IL]; 52, Tzvi Herrer Street, 28000 Kiryat Ata (IL). **LINDER, Natan** [IL/IL]; 21, Hatapuach Street, 90820 Motaz Illit (IL). **BEN-YAIR, Ran** [IL/IL]; #63 Moshav Shoresh, m.p., 90860 Harrey Yehuda (IL). **EISENBERG, Yariv** [IL/IL]; #5 APT., 18 Bergson Street, Tel-Aviv (IL).

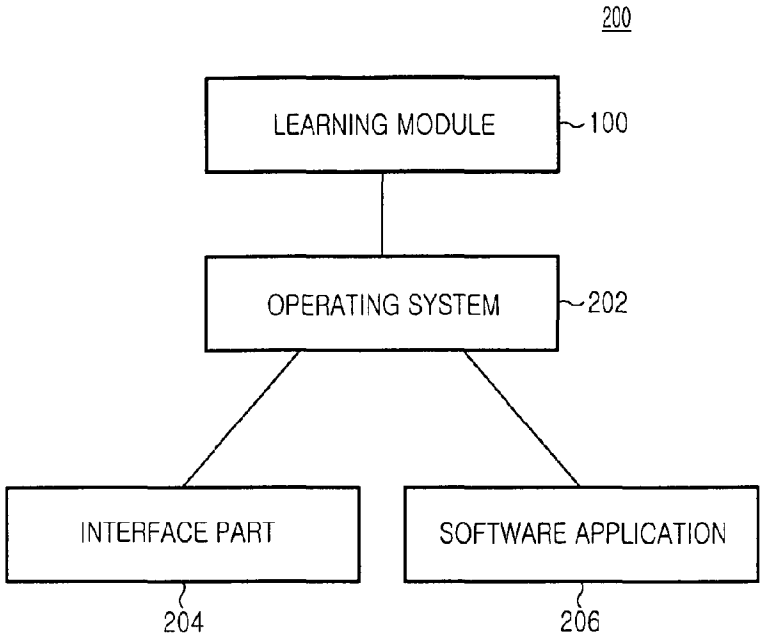
(74) Agent: **LEE, Keon-Joo**; Mihwa Bldg. 110-2, Myon-gryun-dong 4-ga, Chongro-gu, Seoul 110-524 (KR).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,

[Continued on next page]

(54) Title: PROACTIVE USER INTERFACE INCLUDING EVOLVING AGENT



(57) Abstract: A proactive user interface, installed in (or otherwise control and/or be associated with) any type of computational device. The proactive user interface actively makes suggestions to the user, based upon prior experience with a particular user and/or various preprogrammed patterns from which the computational device could select, depending upon user behavior. These suggestions can be made by altering the appearance of at least a portion of the display, for example by changing a menu or a portion thereof; providing different menus for display; and/or altering touch screen functionality. The suggestions can also be made audibly.

WO 2005/024649 A1

WO 2005/024649 A1

ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Published:

— *with international search report*

PROACTIVE USER INTERFACE INCLUDING EVOLVING AGENT

BACKGROUND OF INVENTION

1. Field of the Invention

The present invention relates to a proactive user interface including an evolving agent, and systems and methods thereof, particularly for use with mobile information devices.

1. Description of Related Art

The use of mobile and portable wireless devices has expanded dramatically in recent years. Many such devices having varying functions, internal resources, and capabilities now exist, and include, but are not limited to, mobile telephones, personal digital assistants, medical and laboratory instrumentation, smart cards, and set-top boxes. All such devices can be refered to are mobile information devices. The devices tend to be special purpose, limited-function devices, rather than the general-purpose personal computer. Many of these devices are connected to the Internet, and are used for a variety of applications.

One example of such a mobile information device is the cellular telephone. Cellular telephones are fast becoming ubiquitous; and the use of cellular telephones is even surpassing that of traditional PSTN (public switched telephone network) telephones or "land line" telephones. Cellular telephones themselves are becoming more sophisticated, and in fact are actually computational devices with embedded operating systems.

As cellular telephones become more sophisticated, the range of functions that they offer is also potentially becoming more extensive. However, currently available functions are typically related to extensions of functions already present in regular (land line) telephones, and/or the merging of certain functions of personal digital assistants (PDAs) with those of cellular telephones. The user interface provided with cellular telephones is similarly non-sophisticated, typically featuring a keypad for scrolling through a few simple menus. Customization, although clearly desired by customers who have spent significant amounts of money on personalized ring tones and other cellular telephone accessories, is still limited to a very few functions of the cellular telephone. Furthermore, cellular telephones currently lack any automatic personalization, for example the user interface and custom/tailored functionalities that are required for

better use of the mobile information device, and/or the ability to react according to the behavior of the user.

This lack of sophistication, however, is also seen with user interfaces for personal (desk top or laptop) computers and other computational devices. These computational devices can also only be customized in very simple ways. Such customization must be performed by the user, who may not understand computer functions and/or may not feel comfortable with performing such customization tasks. Currently, computational devices do not learn patterns of user behavior and adjust their own behavior accordingly, as adaptive systems for the user interface. If the user cannot manually adjust the computer, then the user must adjust his/her behavior to accommodate the computer, rather than vice versa.

Software which is capable of learning has been developed, albeit only for specialized laboratory functions. For example, "artificial intelligence" (AI) software has been developed. The term "AI" has been given a number of definitions. "AI is the study of the computations that make it possible to perceive, reason, and act."(Artificial Intelligence A Modern Approach (second edition) by Stuart Russell , Peter Norvig (Prentice Hall, Pearson Education Inc, 2003). AI software combines several different concepts, such as perception, which provides an interface to the world in which the AI software is required to reason and act. Examples include but are not limited to, natural language processing – communicating, understanding document content and context of natural language; computer vision – perceive objects from imagery source; and sensor systems – perception of objects and features of perceived objects analyzing sensory data, etc.

Another important concept is that of the knowledge base. Knowledge representation is responsible for representing, extracting and storing knowledge. This discipline also provides techniques to generalize knowledge, feature extraction and enumeration, object state construction and definitions. The implementation itself may be performed by commonly using known data structures, such as graphs, vectors, tables, etc.

Yet another important concept is that of reasoning. Automated reasoning combines the algorithms that use the knowledge representation and perception to draw new conclusions, infer questions and answers, and achieve the agent goals. The following conceptual frameworks are examples of AI reasoning: rule bases – system rules are evaluated against the knowledge base and perceived state for reasoning; search systems – the use of well known data structures for searching for an intelligent conclusion according to the perceived state, the available knowledge and goal (examples include decision trees, state graphs, minimax

decision etc); classifiers – the target of the classifier reasoning system is to classify a perceived state represented as an experiment that has no classification tag. According to a pre-classified knowledge base the classifier will infer the classification of the new experiment (examples include vector distance heuristics, Support Vector Machine, Classifier Neural Network etc).

Another important concept is for learning. The target of learning is improving the potential performance of the AI reasoning system by generalization over experiences. The input of a learning algorithm will be the experiment and the output would be modifications of the knowledge base according to the results (examples include Reinforcement learning, Batch learning, Support Vector Machine etc).

Work has also been done for genetic algorithms and evolution algorithms for software. One example of such software is described in "Evolving Virtual Creatures", by Karl Sims (*Computer Graphics*, SIGGRAPH '94 Proceedings, July 1994, pp. 15-22). This reference described software "creatures" which could move through a three-dimensional virtual world, which is a simulated version of the actual physical world. The creatures could learn and evolve by using genetic algorithms, thereby changing their behaviors without directed external input. These genetic algorithms therefore delineated a hyperspace of potential behaviors having different "fitness" or rewards in the virtual world. The algorithms themselves were implemented by using directed graphs, which describe both the genotypes (components) of the creatures, and their behavior.

At the start of the simulation, many different creatures with different genotypes are simulated. The creatures are allowed to alter their behavior in response to different stimuli in the virtual world. At each "generation", only certain creatures are allowed to survive, either according to a relative or absolute cut-off score, with the score being determined according to the fitness of the behavior of the creatures. Mutations are permitted to occur, which may increase the fitness (and hence survivability) of the mutated creatures, or vice versa. Mutations are also performed through the directed graph, for example by randomly changing a value associated with a node, and/or adding or deleting nodes. Similarly, "mating" between creatures may result in changes to the directed graph.

The results described in the reference showed that in fact virtual creatures could change and evolve. However, the creatures could only operate within their virtual world, and had no point of reference or contact with the actual physical world, and/or with human computer operators.

SUMMARY OF THE INVENTION

The background art does not teach or suggest a system or method for enabling intelligent software at least for mobile information devices to learn and evolve specifically for interacting with human users. The background art also does not teach or suggest an intelligent agent for a mobile information device, which is capable of interacting with a human user through an avatar. The background art also does not teach or suggest a proactive user interface for a mobile device, in which the proactive user interface learns the behavior of the user and is then able to actively suggest options for evolution of the agent to the user. The background art also does not teach or suggest an agent for a mobile information device, which uses an avatar to interact with another avatar of another mobile information device or the user thereof.

The background art does not teach or suggest a system or method for enabling intelligent software at least for mobile information devices to express an emotion specifically for interacting with human users. The background art also does not teach or suggest a proactive user interface for a computational device, in which the proactive user interface learns the behavior of the user and is then able to actively suggest options to the user and express an emotion according to a reaction of the user to the suggestion. The background art also does not teach or suggest an intelligent agent for a mobile information device, which can perform interaction with a human user through an avatar, said interaction including emotional expression.

The present invention overcomes these deficiencies of the background art by providing a proactive user interface, which could optionally be installed in (or otherwise control and/or be associated with) any type of computational device. The proactive user interface would actively suggest options for evolution of the agent to the user, based upon prior experience with a particular user and/or various preprogrammed patterns from which the computational device could select, depending upon user behavior. In addition, the proactive user interface would actively make suggestions to the user, and/or otherwise engage in non-deterministic or unexpected behavior, based upon prior experience (interaction) with a particular user and/or various preprogrammed patterns from which the computational device could select, depending upon user behavior. These suggestions could optionally be made by altering the appearance of at least a portion of the display, for example by changing a menu or a portion thereof;

providing different menus for display; and/or altering touch screen functionality. The suggestions could also optionally be made audibly. Other types of suggestions or delivery mechanisms are possible. The present invention features the expression of an emotion of the agent according to a reaction of the user to such a suggestion.

By "suggestion" it should be noted that the system could actually execute the action automatically and express a corresponding emotion of the agent, given certain user preferences and also depending upon whether the system state allows the specific execution of the action. The present invention is characterized in that the emotional expression of the agent depends upon whether or not the user makes a selection in response to a "suggestion" of the agent or upon the user's rewards to the suggestion of the agent.

Generally, it is important to emphasize that the proactive user interface preferably at least appears to be intelligent and interactive, and is preferably capable of at least somewhat "free" (e.g. non-scripted or partially scripted) communication with the user. An intelligent appearance is important in the sense that the expectations of the user are preferably fulfilled for interactions with an "intelligent" agent/device. These expectations may optionally be shaped by such factors as the ability to communicate, the optional appearance of the interface, the use of anthropomorphic attribute(s) and so forth, which are preferably used to increase the sense of intelligence in the interactions between the user and the proactive user interface. In terms of communication received from the user, the proactive user interface is preferably able to sense how the user wants to interact with the mobile information device. Optionally, communication may be in only one direction; for example, the interface may optionally present messages or information to the user, but not receive information from the user, or alternatively the opposite may be implemented. Preferably, communication is bi-directional for preferred interactions with the user.

For communication to the user, the proactive interface is capable of displaying or demonstrating simulated emotions for interactions with the user, as part of communication with the user. As described in greater detail below, these emotions are simulated for presentation by an intelligent agent, represented by an avatar or creature. The emotions are created through an emotional system, which may optionally be at least partially controlled according to at least one user preference. The emotional system is used in order for the reactions and communications of the intelligent agent to be believable in terms of the

perception of the user; for example, if the intelligent agent is presented as a dog-like creature, the emotional system enables the emotions to be consistent with the expectations of the user with regard to "dog-like" behavior.

Similarly, the intelligent agent at least appears to be intelligent to the user. The intelligence may optionally be provided through a completely deterministic mechanism; however, the basis for at least the appearance of intelligence includes at least one or more random or semi-random elements. Again, such elements are present in order to be consistent with the expectations of the user concerning intelligence with regard to the representation of the intelligent agent.

Adaptiveness is preferably present, in order for the intelligent agent to be able to alter behavior at least somewhat for satisfying the request or other communication of the user. Even if the proactive user interface does not include an intelligent agent for communicating with the user, adaptiveness enables the interface to be proactive. Observation of the interaction of the user with the mobile information device enables such adaptiveness to be performed, although the reaction of the proactive user interface to such observation may be guided by a knowledge base and/or a rule base.

As a specific, non-limiting but preferred example of such adaptiveness, particularly for a mobile information device which includes a plurality of menus, such adaptiveness may include the ability to alter at least one aspect of the menu. For example, one or more shortcuts may be provided, enabling the user to directly reach a menu choice while by-passing at least one (and more preferably all) of the previous menus or sub-menus which are higher in the menu hierarchy than the final choice. Optionally (alternatively or additionally), one or more menus may be rearranged according to adaptiveness of the proactive user interface, for example according to frequency of use. Such a rearrangement may include moving a part of a menu, such as a menu choice and/or a sub-menu, to a new location that is higher in the menu hierarchy than the current location. Sub-menus which are higher in a menu hierarchy are reached more quickly, through the selection of fewer menu choices, than those which are located in a lower (further down) location in the hierarchy.

Adaptiveness and/or emotions are assisted through the use of rewards for learning by the proactive user interface. Suggestions or actions of which the user approves provide a reward, or a positive incentive, to the proactive interface to continue with such suggestions or actions; disapproval by the user causes a disincentive to the proactive user interface to continue such behavior(s). Providing positive or negative incentives/disincentives to the proactive user

interface preferably enables the behavior of the interface to be more nuanced, rather than a more "black or white" approach, in which a behavior would either be permitted or forbidden. Such nuances are also preferred to enable opposing or contradictory behaviors to be handled, when such behaviors are collectively approved/disapproved by the user to at least some extent.

According to an embodiment of the present invention, a model of the user is constructed through the interaction of the proactive user interface with the user. Such a model would integrate AI knowledge bases determined from the behavior of the user and/or preprogrammed. Furthermore, the model would also enable the proactive user interface to gauge the reaction of the user to particular suggestions made by the user interface, thereby adapting to the implicit preferences of the user.

Non-limiting examples of such computational devices include automated teller machines (ATM's) (this also has security implications, as certain patterns of user behavior could set off an alarm, for example), regular computers of any type (such as desktop, laptop, thin clients, wearable computers and so forth), mobile information devices such as cellular telephones, pager devices, other wireless communication devices, regular telephones having an operating system, PDA's and wireless PDA's, and consumer appliances having an operating system. Hereinafter, the term "computational device" includes any electronic device having an operating system and being capable of performing computations. The operating system may be an embedded system and/or another type of software and/or hardware run time environment. Hereinafter, the term "mobile information device" includes, but is not limited to, any type of wireless communication device, including, but not limited to, cellular telephones, wireless pagers, wireless PDA's and the like.

The present invention is implemented in order to provide an enhanced user experience and interaction with the computational device, as well as to change the current generic, non-flexible user interface of such devices into a flexible, truly user friendly interface. More preferably, the present invention is implemented to provide an enhanced emotional experience of the user with the computational device, for example according to the optional but preferred embodiment of constructing the user interface in the form of an avatar which would interact with the user. The present invention is therefore capable of providing a "living device" experience, particularly for mobile information devices such as cellular telephones, for example. According to this embodiment, the user may even form an emotional attachment to the "living device".

According to another embodiment of the present invention, there is provided a mobile information device which includes an adaptive system. Like the user interface above, it also relies upon prior experience with a user and/or preprogrammed patterns. However, the adaptive system is more restricted to operating within the functions and environment of a mobile information device.

Either or both of the mobile information device adaptive system and proactive user interfaces may be implemented with genetic algorithms, artificial intelligence (AI) algorithms, machine learning (ML) algorithms, learned behavior, and software/computational devices which are capable of evolution. Either or both may also provide an advanced level of voice commands, touch screen commands, and keyboard 'short-cuts'.

According to another preferred embodiment of the present invention, there is provided one or more intelligent agents for use with a mobile information device over a mobile information device network, preferably including an avatar (or "creature"; hereinafter these terms are used interchangeably) through which the agent may communicate with the human user. The avatar can provide a user interface for interacting with the user. The intelligent agent can also include an agent for controlling at least one interaction of the mobile information device over the network. This embodiment may include a plurality of such intelligent agents being connected over the mobile information device network, thereby forming a network of such agents. Various applications may also be provided through this embodiment, including but not limited to teaching in general and/or for learning how to use the mobile information device in particular, teaching languages, communication applications, community applications, games, entertainment, shopping (getting coupons, etc), locating a shop or another place, filtering advertisements and other non-solicited messages, role-playing or other interactive games over the cell phone network, "chat" and meeting functions, the ability to buy "presents" for the intelligent agents and otherwise accessorize the character, and so forth. In theory, the agents themselves could be given "pets" as accessories.

The intelligent agents could also assist in providing various business/promotional opportunities for the cell phone operators. The agents could also assist with installing and operating software on cell phones, which is a new area of commerce. For example, the agents could assist with the determination of the proper type of mobile information device and other details that are essential for correctly downloading and operating software.

Therefore, a number of different interactions are possible according to the

various embodiments of the present invention. These interactions include any one or more of an interaction between the user of the device and an avatar or other character or personification of the device; an interaction between the user of the device and the device, for operating the device, through the avatar or other character or personification; interactions between two users through their respective devices, by communicating through the avatar, or other character or personification of the device; and interactions between two devices through their respective intelligent agents, and can be done without any communication between users or even between the agent and the user. The interaction or interactions that are possible are determined according to the embodiment of the present invention, as described in greater detail below.

The present invention benefits from the relatively restricted environment of a computational device and/or a mobile information device, such as a cellular telephone for example, because the parameters of such an environment are known in advance. Even if such devices are communicating through a network, such as a cellular telephone network for example, the parameters of the environment can still be predetermined. The current computational devices only provide a generic interface, with little or no customization permitted by even manual, direct intervention by the user.

It should be noted that the term "software" may also optionally include firmware or instructions operated by hardware.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is herein described, by way of example only, with reference to the accompanying drawings, wherein:

FIG. 1 is a block diagram of an exemplary learning module according to the present invention;

FIG. 2 is a block diagram of an exemplary system according to the present invention for using the proactive user interface;

FIG. 3 shows an exemplary implementation of a proactive user interface system according to the present invention;

FIG. 4 is a block diagram of an exemplary implementation of the adaptive system according to the present invention;

FIGS. 5A and 5B are a block diagram and a sequence diagram, respectively, of an exemplary application management system according to the present invention;

FIGS. 6A and 6B show an exemplary infrastructure required for the adaptive system according to the present invention to perform one or more actions through the operating system of the mobile information device and an exemplary sequence diagram thereof according to the present invention;

FIGS. 7A, 7B and 7C show exemplary events, and how they are handled by interactions between the mobile information device (through the operating system of the device) and the system of the present invention;

FIGS. 8A and 8B describe an exemplary structure of the intelligent agent and also includes an exemplary sequence diagram for the operation of the intelligent agent;

FIGS. 9A and 9B show two exemplary methods for selecting an action according to the present invention;

FIG. 10 shows a sequence diagram of an exemplary action execution method according to the present invention;

FIGS. 11A, 11B and 11C are diagrams for describing an exemplary, illustrative implementation of an emotional system according to the present invention;

FIG. 12A shows an exemplary sequence diagram for textual communication according to the present invention;

FIG. 12B shows a non-limitting example of an emotional expression "I am happy" that the agent performs in a mobile phone;

FIGS. 13A, 13B and 13C show an exemplary evolution class diagram, and an exemplary mutation and an exemplary hybrid sequence diagram, respectively, according to the present invention;FIG. 14 shows an exemplary hybridization sequence between intelligent agents on two mobile information devices;

FIGS. 15, 16, 17, 18, 19, 20 and 21 show exemplary screenshots of an avatar or creature according to different embodiments of the present invention;

FIG. 22 is a block diagram of an exemplary intelligent agent system according to the present invention;

FIG. 23 shows the system of Figure 23 in more detail;

FIG. 24 is a block diagram of an exemplary implementation of an action selection system according to the present invention; and

FIGS. 25A and 25B show exemplary screenshots of the avatar according to the present invention on the screen of the mobile information device.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Preferred embodiments of the present invention will be described herein below with reference to the accompanying drawings. In the following description, well-known functions or constructions are not described in detail since they would obscure the invention in unnecessary detail.

The present invention is of a proactive user interface, which could be installed in (or otherwise control and/or be associated with) any type of computational device. The proactive user interface actively makes suggestions to the user, based upon prior experience with a particular user and/or various preprogrammed patterns from which the computational device could select, depending upon user behavior. These suggestions could optionally be made by altering the appearance of at least a portion of the display, for example by changing a menu or a portion thereof; providing different menus for display; and/or altering touch screen functionality. The suggestions could also be made audibly. The present invention features the expression of an emotion of the agent according to a reaction of the user to such a suggestion.

The proactive user interface is preferably implemented for a computational device, as previously described, which includes an operating system. The interface can include a user interface for communicating between the user and the operating system. The interface can also include a learning module for detecting at least one pattern of interaction of the user with the user interface and for actively suggesting options for evolution of at least one function of the user interface to the user, according to the detected pattern. Therefore, the proactive user interface can anticipate the requests of the user and thereby assist the user in selecting a desired function of the computational device.

At least one pattern can be selected from the group consisting of a pattern determined according to at least one previous interaction of the user with the user interface, and a predetermined pattern, or a combination thereof. The first type of pattern represents learned behavior, while the second type of pattern may be preprogrammed or otherwise predetermined, particularly for assisting the user when a particular computational device is first being operated by the user. A third type of pattern could combine these two aspects, and would enable the pattern to be at least partially determined according to the user behavior, but not completely; for example, the pattern selection may be guided according to a plurality of rules, and/or according to a restrictive definition of the possible world environment state and/or the state of the device and/or user interface. The pattern includes a pattern of the user's preferences for the appearance, function or characteristic of the intelligent agent.

The user interface preferably features a graphical display, such that at least one function of the graphical display is proactively altered according to the pattern. For example, at least a portion of the graphical display may be altered, for example by selecting a menu for display according to the detected pattern; and displaying the menu. The menu may be selected by constructing a menu from a plurality of menu options, for example in order to create a menu "on the fly".

The user interface may feature an audio display, such that altering at least one function of the user interface involves altering at least one audible sound produced by the computational device.

The proactive user interface could be implemented according to a method of the present invention, which is preferably implemented for a proactive interaction between a user and a computational device through a user interface. The method preferably includes detecting a pattern of user behavior according to at least one interaction of the user with the user interface; and proactively altering at least one function of the user interface according to the pattern. The pattern includes a pattern of user preferences for the appearance, function or characteristic of the intelligent agent.

According to another embodiment of the present invention, there is provided a mobile information device which includes an adaptive system. Like the user interface above, it also relies upon prior experience with a user and/or preprogrammed patterns. However, the adaptive system can be more restricted to operating within the functions and environment of a mobile information device, such as a cellular telephone for example, which currently may also include certain basic functions from a PDA.

The adaptive system preferably operates with a mobile information device featuring an operating system. The operating system can comprise an embedded system. The mobile information device can comprise a cellular telephone.

The adaptive system is preferably able to analyze the user behavior by analyzing a plurality of user interactions with the mobile information device, after which more preferably the adaptive system compares the plurality of user interactions to at least one predetermined pattern, to see whether the predetermined pattern is associated with altering at least one function of the user interface. The analysis may also include comparing the plurality of user interactions to at least one pattern of previously detected user behavior, wherein the pattern of previously detected user behavior is associated with altering at least one function of the user interface.

The adaptive system may be operated by the mobile information device itself. Alternatively, if the mobile information device is connected to a network, the adaptive system may be operated at least partially according to commands sent from the network to the mobile information device. For this implementation, data associated with at least one operation of the adaptive system is stored at a location other than the mobile information device, in which the location is accessible through the network.

According to preferred embodiments of the present invention, the adaptive system also includes a learning module for performing the analysis according to received input information and previously obtained knowledge. Such knowledge may have been previously obtained from the behavior of the user, and/or may have been communicated from another adaptive system in communication with the adaptive system of the particular mobile information device. The adaptive system can adapt to user behavior according to any one or more of an AI algorithm, a machine learning algorithm, or a genetic algorithm.

According to another optional but preferred embodiment of the present invention, there is provided one or more intelligent agents for use with a mobile information device over a mobile information device network, preferably including an avatar through which the agent may communicate with the human user. The avatar can therefore provide a user interface for interacting with the user. The intelligent agent can also include an agent for controlling at least one interaction of the mobile information device over the network. This embodiment may include a plurality of such avatars being connected over the mobile information device network.

According to preferred embodiments of the present invention, at least one characteristic of an appearance of the avatar can be altered, for example according to a user command. A plurality of characteristics of an appearance of avatar can be altered according to a predefined avatar skin. The skin can be predefined by the user. By "skin" it is meant that a plurality of the characteristics is altered together as a set, in which the set forms the skin. If this embodiment is combined with the previous embodiment of having at least a portion of the data related to the avatar being stored at a network-accessible location, then the user could move the same avatar onto different phones, and/or customize the appearance of the avatar for different reasons, for example for special occasions such as a party or other celebration. Of course, these are only intended as examples and are not meant to be limiting in any way.

According to other embodiments of the present invention, at least one

characteristic of an appearance of the avatar can be altered according to an automated evolutionary algorithm, for example a genetic algorithm. The evolutionary algorithm is one non-limiting example of a method for providing personalization of the avatar for the user. Personalization may also be performed through direct user selection of one or more characteristics or skins (groups of characteristics). Such personalization is desirable at least in part because it enhances the emotional experience of the user with the avatar and hence with the mobile information device.

In terms of technical implementation, the present invention is preferably capable of operating on a limited system (in terms of memory, data processing capacity, screen display size and resolution, and so forth) in a device which is also very personal to the user. For example, the device is a mobile information device, such as a cellular telephone, which by necessity is adapted for portability and ease of use, and therefore may have one or more, or all, of the above limitations. The implementation aspects of the present invention are preferably geared to this combination of characteristics. Therefore, in order to overcome the limitations of the device itself while still maintaining the desirable personalization and "personal feel" for the user, various solutions are proposed below. It should be noted that these solutions are examples only, and are not meant to be limiting in any way.

EXAMPLE 1: PROACTIVE INTERFACE – General

The proactive user interface of the present invention is preferably able to control and/or be associated with any type of computational device, in order to actively make suggestions to the user, based upon prior experience with a particular user and/or various preprogrammed patterns from which the computational device could select, depending upon user behavior. These suggestions could be made by altering the appearance of at least a portion of the display, for example by changing a menu or a portion thereof; providing different menus for display; and/or altering touch screen functionality. The suggestions could also be made audibly.

The proactive user interface is preferably implemented for a computational device, as previously described, which includes an operating system. The interface can include a user interface for communicating between the user and the operating system. The interface is preferably able to detect at least one pattern of interaction of the user with the user interface, for example through operation of

a learning module and would therefore be able to proactively alter at least one function of the user interface according to the detected pattern. The proactive user interface can anticipate the requests of the user and thereby assist the user in selecting a desired function of the computational device.

This type of proactive behavior, particularly with regard to learning the behavior and desires of the user, requires some type of learning capability on the part of the proactive interface. Such learning capabilities may be provided through algorithms and methodologies which are known in the art, relating to learning (by the software) and interactions of a software object with the environment. Software can be said to be learning when it can improve its actions over a period of time. Artificial Intelligence needs to demonstrate intelligent action selection (reasoning), such that the software has the ability to explore its environment (its "world") and to discover action possibilities. The software would also have the ability to represent the world's state and its own internal state. The software would then be able to select an intelligent action (using the knowledge above) and to act.

Learning, for example by the learning module of the interface, can be reinforced by rewards, in which the learning module is rewarded for taking particular actions according to the state of the environment. This type of learning actually involves training the learning module to behave in a certain manner. If more than one behavior is allowed, then the learning process is non-deterministic and can create different behaviors. With regard to the proactive user interface, for example, the reward includes causing the learning module to detect when an offered choice leads to a user selection, as opposed to when an offered choice causes the user to seek a different set of one or more selections, for example by selecting a different menu than the one offered by the proactive user interface. Clearly, the proactive user interface should seek to maximize the percentage of offerings which lead to a direct user selection from that offering, as this shows that the interface has correctly understood the user behavior.

Figure 1 is a block diagram of an exemplary learning module according to the present invention for reactive learning. As shown, a learning module **100** includes a Knowledge Base **102**, which acts as the memory of learning module **100**, by holding information gathered by the learning module **100** as a result of interactions with the environment. Knowledge Base **102** may be stored in non-volatile memory (not shown). Knowledge Base **102** stores information that assists the learning module **100** to select the appropriate action. This information can include values such as numerical weights for an inner neural net, or a table with

action reward values, or any other type of information.

In order for learning module **100** to be able to receive information related to the environment, the learning module **100** features a plurality of sensors **104**. The sensors **104** allow the learning module **100** to perceive its environment state. The sensors **104** are connected to the environment and output sensed values. The values can come from the program itself (for example, position on screen, energy level, etc.), or from real device values (for example, battery value and operating state, such as a flipper state for cellular telephones in which the device can be activated or an incoming call answered by opening a "flipper").

Sensors **104** clearly provide valuable information; however, this information needs to be processed before the learning module **100** can comprehend it. Therefore, the learning module **100** also includes a perception unit **106**, for processing the current output of the sensors **104** into a uniform representation of the world, called a "state". The state is then the input to a reasoning system **108**, which may be described as the "brain" of learning module **100**. This design supports the extension of the world state and the sensor mechanism, as well as supporting easy porting of the system to several host platforms (different computational devices and environments), such that the world state can be changed according to the device.

The reasoning system **108** processes the current state with the Knowledge Base **102**, thereby producing a decision as to which action to perform. The reasoning system **108** receives the current state of the world, outputs the action to be performed, and receives feedback on the action selected. Based on the feedback, the reasoning system **108** updates the Knowledge Base **102**. This is an iterative process in which learning module **100** learns to associate actions to states.

According to another embodiment of the present invention, the computational device may feature one or more biological sensors, for sensing various types of biological information about the user, such as emotional state, physical state, movement, etc. This information may then be fed to the sensors **104** for assisting the perception unit **106** in a determination of the state of the user, and hence to determine the proper state for the device. Such biological sensors may include but are not limited to sensors for body temperature, heart rate, oxygen saturation or any other type of sensor which measures biological parameters of the user.

Figure 2 shows an exemplary embodiment of a system **200** according to the present invention for providing the proactive user interface, again featuring the learning module **100**. The learning module **100** is shown communicating with

an operating system **202** of the computational device (not shown) with which the learning module **100** is associated and/or controls and/or by which the learning module **100** is operated. The operating system **202** controls the operation of an interface **204** and also at least one other software application **206** (although of course many such software applications may optionally be present).

The user communicates through interface **204**, for example by selecting a choice from a menu. The operating system **202** enables this communication to be received and translated into data. The learning module **100** then preferably receives such data, and can send a command back to the operating system **202**, for example to change some aspect of the interface **204** (for example by offering a different menu), and/or to operate the software application **206**. The user then responds through the interface **204**; from this response, the learning module **100** learns whether or not the action (command that was sent by learning module **100**) was appropriate.

Figure 3 is a block diagram showing an exemplary implementation of a proactive user interface system **300** according to the present invention. As shown, system **300** features a three level architecture, with an application layer being supported by an AI (artificial intelligence) framework, which in turn communicates with the host platform computational device (shown as "host platform").

The application layer features a plurality of different applications, of which a few non-limiting examples are shown, such as a MutateApp **302**, a PreviousApp **304** and a TeachingApp **306**.

The MutateApp **302** is invoked in order to control and/or initiate mutations in the system **300**. As noted above, the learning module can optionally change its behavior through directed or semi-directed evolution, for example through genetic algorithms. The MutateApp **302** controls and/or initiates such mutations through evolution. The embodiment of evolution is described in greater detail below.

The PreviousApp **304** enables a prior state of the system **300**, or a portion thereof (such as the state of the learning module) to be invoked in place of the current state. More specifically, the PreviousApp **304** enables the user to return to the previous evolutionary step if the present invention is being implemented with an evolutionary algorithm. More generally, the system **300** is preferably stateful and therefore can return to a previous state, as a history of such states is preferably maintained.

The TeachingApp **306** is only one non-limiting example of a generic

application which may be implemented over the AI framework layer.

The AI framework layer itself contains one or more components which enable the user interface to behave in a proactive manner. The framework can include a DeviceWorldMapper **308**, for determining the state of the computational device and also that of the virtual world, as well as the relationship between the two states. The DeviceWorldMapper **308** receives input, for example from various events from an EventHandler **310**, in order to determine the state of the virtual world and that of the device.

The DeviceWorldMapper **308** also communicates with an AI/ML (machine learning) module **312** for analyzing input data. The AI/ML module **312** also determines the behavior of the system **300** in response to various stimuli, and also enables the system **300** to learn, for example from the response of the user to different types of user interface actions. The behavior of the system **300** may also be improved according to an evolution module **314**.

The embodiment of evolution is particularly preferred with regard to the use of an intelligent agent on a mobile information device (see below for an example), but may also be used with any proactive user interface for a computational device. This embodiment is used when the proactive user interface also features or is used in combination with an avatar.

Evolution can be simulated by a set of genetic algorithms. The basis of these algorithms is describing the properties of the proactive interface (and particularly the avatar's appearance) in term of genes, chromosomes, and phenotypes. The gene is a discrete property that has a level of expression for example a leg of a certain type. The level of the expression can be the number of these legs.

A phenotype is the external expression of a gene; for example the leg gene can have different phenotypes in term of leg length or size.

The gene can go though a mutation process. This process (preferably according to a certain probability) changes one or more parameter of the gene, thereby producing different new phenotypes.

A chromosome is a set of genes that function together. The chromosome can hybridize (cross-breed) with the same type of chromosome from a different creature, thus creating a new chromosome that is a combination of its genetic parent chromosomes.

This methodology helps in creating a generic infrastructure to simulate visual evolution (for example of the appearance of the avatar) and/or evolution of the behavior of the proactive user interface. These algorithms may also be used

for determining non-visual behavioral characteristics, such as dexterity, stamina and so on. The effect could result for example in a faster creature, or a more efficient creature. These algorithms may be used for any such characteristics that can be described according to the previously mentioned gene/genotype/phenotype structure, such that for example behavioral genes could optionally determine the behavior of AI algorithms used by the present invention.

The algorithm output preferably provides a variety of possible descendant avatars and/or proactive user interfaces.

The genetic algorithms use a natural selection process to decide which of the genetic children will continue as the next generation. The selection process can be decided by the user or can be predefined. In this way the creature can display interesting evolutional behavior. The generic algorithm framework can be used to evolve genes that encode other non visual properties of the creature, such as goals or character.

The Evolution module **314** is a non-limiting example of the application for managing the evolutions of the intelligent agent. The evolution module **314** supports and also preferably manages such evolution, for example through the operation of the MutateApp **302**.

Between these different AI-type applications and the EventHandler **310**, one or more different low level managers preferably support the receipt and handling of different events, and also the performance of different actions by the system **300**. These managers may include but are not limited to, an ActionManager **316**, a UIManager **318**, a StorageManager **320** and an ApplicationManager **322**.

The ActionManager **316** is described in greater detail below, but briefly enables the system **300** to determine which action should be taken, for example through the operation of the AI/ML module **312**.

The UIManager **318** manages the appearance and functions of the user interface, for example by directing changes to that interface as previously described.

The StorageManager **320** manages the storage and handling of data, for example with regard to the knowledge base of the system **300** (not shown).

The ApplicationManager **322** handles communications with the previously described applications in the application layer.

All of these different managers receive events from the EventHandler **310**.

Within the AI framework layer, an AI infrastructure **324** supports communication with the host platform. The host platform itself features a host

platform interface **326**, which may be provided through the operating system of the host platform for example.

The AI infrastructure **324** can include an I/O module **328**, for receiving inputs from the host platform interface **326** and also for sending commands to the host platform interface **326**. A screen module **330** handles the display of the user interface on the screen of the host platform computational device. A resources module **332** enables the system **300** to access various host platform resources, such as data storage and so forth.

Of course, the above Figures represent only one optional configuration for the learning module. For example, the learning module may also be represented as a set of individual agents, in which each agent has a simple goal. The learning module chooses an agent to perform an action based on the current state. The appropriate mapping between the current state and agents can also be learned by the learning module with reinforcement learning.

EXAMPLE 2: ADAPTIVE SYSTEM FOR MOBILE INFORMATION DEVICE

This example relates to the illustrative implementation of an adaptive system of the present invention with a mobile information device, although it should be understood that this implementation is preferred but optional, and is not intended to be limiting in any way.

The adaptive system may optionally include any of the functionality described above in Example 1, and may also be implemented as previously described. This Example focuses more on the actual architecture of the adaptive system with regard to the mobile information device operation. Also, this Example describes an optional but preferred implementation of the creature or avatar according to the present invention.

The next sections describe optional but preferred embodiments of specific technical implementations of various aspects of the adaptive system according to the present invention. For the purpose of description only and without any intention of being limiting, these embodiments are based upon the optional but preferred embodiment of an adaptive system interacting with the user through an intelligent agent, optionally visually represented as an avatar or "creature".

Section 1: Event Driven System

This Section describes a preferred embodiment of an event driven system according to the present invention, including but not limited to an application

manager, and interactions between the device itself and the system of the present invention as it is operated by the device.

Figure 4 is a block diagram of an exemplary adaptive system **400** according to the present invention, and interactions of the system **400** with a mobile information device **402**. Also as shown, both the system **400** and the mobile information device **402** interact with a user **404**.

The mobile information device **402** has a number of standard functions, which are shown divided into two categories for the purpose of explanation only: data and mechanisms. Mechanisms may include but are not limited to such functions as a UI (user interface) system **406** (screen, keypad or touchscreen input, etc); incoming and outgoing call function **408**; messaging function **410** for example for SMS; sound **412** and/or vibration **414** for alerting user **404** of an incoming call or message, and/or alarm, etc; and storage **416**.

Data may include such information as an address (telephone) book **418**; incoming or outgoing call information **420**; the location of the mobile information device **402**, shown as location **422**; message information **424**; cached Internet data **426**; and data related to the user **404**, shown as owner data **428**.

It should be noted that mobile information device **402** may include any one or more of the above data/mechanisms, but does not necessarily need to include all of them, and/or may include additional data/mechanisms that are not shown. These are simply intended as non-limiting examples with regard to the mobile information device **402**, particularly for cellular telephones.

The adaptive system **400** according to the present invention preferably interacts with the data/mechanisms of the mobile information device **402** in order to be able to provide an adaptive (and also preferably proactive) user interface, thereby increasing the ease and efficiency with which the user **404** interacts with the mobile information device **402**.

The adaptive system **400** features logic **430**, which functions in a similar manner as the previously described learning module, and which also operates according to the previously described AI and machine learning algorithms.

The logic **430** is able to communicate with the knowledge base **102** as described with regard to Figure 1 (components featuring the same reference numbers have either identical or similar functionality, unless otherwise stated). The information storage **432** includes data about the actions of the mobile information device **402**, user information and so forth, and preferably supplements the data in the knowledge base **102**.

Preferably, the adaptive system **400** is capable of evolution, through an

evolution logic **434**, which may optionally combine the previously described functionality of the evolution module **314** and the MutateApp **302** of Figure 3.

The adaptive system **400** is capable of communicating directly with the user **404** through text and/or audible language, as supported by a language module **436**.

Particularly as described with regard to the embodiment of the present invention in Example 3 below, but also for the adaptive system **400**, the user **404** may be presented with an avatar (not shown) for the user interface. If present, such an avatar may be created through a 3D graphics model **438** and an animation module **440**. The avatar may be personalized for the user **404**, thereby providing an enhanced emotional experience for the user **404** when interacting with the mobile information device **402**.Figure 5A shows a block diagram of an exemplary application management system **500**, which is a core infrastructure for supporting the adaptive system of the present invention. The system **500** may also be used for supporting such embodiments as a teaching application, as previously described and also as described in greater detail below. The system **500** features an application manager **502** for managing the different types of applications which are part of the adaptive system according to the present invention. The application manager **502** communicates with an application interface called a BaseApp **504**, which is implemented by all applications in the system **500**. Both the application manager **502** and the BaseApp **504** communicate events through an EventHandler **506**.

The application manager **502** is responsible for managing and providing runtime for the execution of the system applications (applications which are part of the system **500**). The life cycle of each such application is defined in the BaseApp **504**, which allows the application manager **502** to start, pause, resume and exit (stop) each such application. The application manager **502** manages the runtime execution through the step method of the interface of BaseApp **504**. It should be noted that the step method is used for execution, since the system **500** is stateful, such that each step preferably corresponds (approximately) to one or more states. However, execution could also be based upon threads and/or any type of execution method.

The application manager **502** receives a timer event from the mobile information device. The mobile information device features an operating system, such that the timer event is received from the operating system layer. When a timer is invoked, the application manager **502** invokes the step of the current application being executed. The application manager **502** switches from one

application to another application when the user activates a different application, for example when using the menu system.

Some non-limiting examples of the system applications are shown, including but not limited to, a TeachingMachineApp **508**, a MutateApp **510**, a GeneStudioApp **514**, a TWizardApp **516**, a FloatingAgentApp **518**, a TCWorldApp **522** and a HybridApp **520**. These applications are also described in greater detail below with regard to Example 3.

The MutateApp **510** is invoked in order to control and/or initiate mutations in the adaptive system, and/or in the appearance of an avatar representing the adaptive system as a user interface. As noted above with regard to Example 1, the adaptive system of the present invention can change its behavior through directed or semi-directed evolution, for example through genetic algorithms. The MutateApp **510** controls and/or initiates such mutations.

The GeneStudioApp **514** enables the user to perform directed and/or semi-directed mutations through one or more manual commands. For example, the user may wish to direct the adaptive system (through the application management system **500**) to perform a particular task sequence upon receiving a particular input. Alternatively, the user may wish to directly change part of the appearance of an avatar, if present. According to the preferred embodiments of the present invention, these different aspects of the adaptive system are implemented by distinct "genes", which can then be altered by the user.

The HybridApp **520** may be invoked if the user wishes to receive information from an external source, such as the adaptive system of another mobile information device, and to merge this information with existing information on the user's mobile information device. For example, the user may wish to create an avatar having a hybrid appearance with the avatar of another mobile information device. The HybridApp **520** also provides the main control of the user on the entire evolutionary state of the avatar. The HybridApp **520** may be used to instruct the user on the "life" properties of with the avatar, which may have a name, personality, behavior and appearance.

The TeachingMachineApp **508** is an illustrative, non-limiting example of an application which may relate to providing instruction on the use of the device itself, but provides instruction on a subject which is not related to the direct operation of the device itself. Therefore, the TeachingMachineApp **508** represents an example of an application which is provided on the mobile information device for a purpose other than the use of the device itself.

The TCWorldApp **522** is an application which runs the intelligent agent,

controlling both the intelligent aspects of the agent and also the graphical display of the creature or avatar.

The TWizardApp **516** is another type of application which provides information to the user. It is described with regard to the Start Wizard application in Example 4 below. Briefly, this application contains the user preferences and configuration of the AI framework, such as the character of the intelligent agent, particularly with regard to the emotional system, and also with regard to setting goal priorities.

The FloatingAgentApp **518** controls the appearance of the user interface, particularly with regard to the appearance of an avatar (if present). The FloatingAgentApp **518** enables the visual display aspects of the user interface to be displayed independently of the display of the avatar, which may therefore appear to "float" over the user interface, for example. The FloatingAgentApp **518** is the default application being operated when no other application is running.

Figure 5B shows an exemplary sequence diagram for the operations of the application manager according to the present invention. As shown, an EventHandler **506** dispatches a notification of an event to the application manager **502**, as shown in arrow 1. If the event is a timer event, then the application manager **502** invokes the step (action) of the relevant application that was already invoked, as shown in arrow 1.1.1. If the event is to initiate the execution of an application, then the application manager **502** invokes the relevant application, as shown in arrow 1.2.1. If a currently running application is to be paused, then the application manager **502** sends the pause command to the application, as shown in arrow 1.3.1. If a previously paused application is to be resumed, then the application manager **502** sends the resume command to the application, as shown in arrow 1.4.1. In any case, successful execution of the step is returned to the application manager **502**, as shown by the relevant return arrows above. The application manager **502** then notifies the EventHandler **506** of the successful execution, or alternatively of the failure.

These different applications are important for enabling the adaptive system to control various aspects of the operation of the mobile information device. However, the adaptive system also needs to be able to communicate directly with various mobile information device components, through the operating system of the mobile information device. Such communication may be performed through a communication system **600**, shown with regard to Figure 6, preferably with the action algorithms described below.

Figures 6A and 6B show an exemplary implementation of the

infrastructure required for the adaptive system according to the present invention to perform one or more actions through the operating system of the mobile information device, as well as a sequence diagram for operation of the communication system **600**. According to embodiments of the present invention, this infrastructure is an example of a more general concept of "AI wrappers", or the ability to "wrap" an existing UI (user interface) system with innovative AI and machine learning capabilities.

The communication system **600** is capable of handling various types of events, with a base class event **602** that communicates with the EventHandler **506** as previously described. The EventDispatcher **604** then routes the event to the correct object within the system of the present invention. Routing is determined by registration of the object with the EventDispatcher **604** for a particular event. The EventDispatcher **604** preferably manages a registry of handlers that implement the EventHandler **506** interface for such notification.

Specific events for which particular handlers are implemented include a flipper event handler **606** for cellular telephones in which the device can be activated or an incoming call answered by opening a "flipper"; when the flipper is opened or closed, this event occurs. Applications being operated according to the present invention may send events to each other, which are handled by an InterAppEvent handler **608**. An event related to the evolution (change) of the creature or avatar is handled by an EvolutionEvent handler **610**. An incoming or outgoing telephone call is handled by a CallEvent handler **612**, which in turn has two further handlers, a CallStartedEvent handler **614** for starting a telephone call and a CallEndedEvent handler **616** for ending a telephone call.

An SMS event (incoming or outgoing message) is handled by an SMSEvent handler **618**. Parameters which may be included in the event comprise parameters related to hybridization of the creature or avatar of one mobile information device with the creature or avatar of another mobile information device, as described in greater detail below.

Events related to operation of the keys are preferably handled by a KeyEvent handler **620** and/or a KeyCodeEvent handler **622**. For example, if the user depresses a key on the mobile information device, the KeyEvent handler **620** preferably handles this event, which relates to incoming information for the operation of the system according to the present invention. In the sequence diagram, the key_event is an object from class KeyEvent, which represents the key event message object. The KeyEvent handler **620** handles the key_event itself, while the KeyCodeEvent handler **622** listens for input code (both input

events are obtained through a hook into the operating system).

A BatteryEvent handler **624** handles events related to the battery, such as a low battery, or alternatively switching from a low power consumption mode to a high power consumption mode.

DayTimeEvent handler **626** relates to alarm, calendar or reminder/appointment diary events.

Figure 6B is an exemplary sequence diagram, which shows how events are handled between the mobile information device operating system or other control structure and the system of the present invention. In this example, the mobile information device has an operating system, although a similar operation flow could be implemented for devices that lack such an operating system. If present, the operating system handles the input and output to/from the device, and manages the state and events which occur for the device. The sequence diagram in Figure 6B is an abstraction for facilitating the handling of, and the relation to, these events.

An operating system module (os_module) **628** causes or relates to an event; a plurality of such modules may be present, but only one is shown for the purposes of clarity and without intending to be limiting in any way. The operating system module **628** is part of the operating system of the mobile information device. The operating system module **628** sends a notification of an event, whether received or created by operating system module **628**, to a hook **630**. The hook **630** is part of the system according to the present invention, and is used to permit communication between the operating system and the system according to the present invention. The hook **630** listens for relevant events from the operating system. The hook **630** is capable of interpreting the event from the operating system, and of constructing the event in a message which is comprehensible to the event **602**. Hook **630** also dispatches the event to the EventDispatcher **604**, which communicates with each handler for the event, shown as the EventHandler **506** (although there may be a plurality of such handlers). The EventDispatcher **604** then reports to the hook **630**, which reports to the operating system module **628** about the handling of the event.

Figures 7A, 7B and 7C show exemplary events, and how they are handled by interactions between the mobile information device (through the operating system of the device) and the system of the present invention. It should be noted that some events may be handled within the system of the present invention, without reference to the mobile information device.

Figure 7A shows an exemplary key event sequence diagram, described

according to a mobile information device that has the DMSS operating system infrastructure from Qualcomm Inc., for their MSM (messaging state machine) CDMA (code division multiple access) mobile platform. This operating system provides operating system services such as user interface service, I/O services and interactive input by using the telephone keys (keypad). This example shows how an input event from a key is generated and handled by the system of the present invention. Other events are sent to the system in almost an identical manner, although the function of the hook **630** alters according to the operating system module which is sending the event; a plurality of such hooks is present, such that each hook has a different function with regard to interacting with the operating system.

As shown in Figure 7A, a ui_do_event module **700** is a component of the operating system and is periodically invoked. When a key on the mobile device is pressed, the user interface (UI) structure which transfers information to the ui_do_event module **700** contains the value of the key. The hook **630** then receives the key value, identifies the event as a key event (particularly if the ui_do_event module **700** dispatches a global event) and generates a key event **702**. The key event **702** is then dispatched to the EventDispatcher **604**. The event is then sent to an application **704** which has requested to receive notification of such an event, preferably through an event handler (not shown) as previously described. Notification of success (or failure) in handling the event is then preferably returned to the EventDispatcher **604** and hence to the hook **630** and the ui_do_event module **700**.

Figure 7B shows a second illustrative example of a sequence diagram for handling an event; in this case, the event is passed from the system of the present invention to the operating system, and is related to drawing on the screen of the mobile information device. Information is passed through the screen access method of the operating system, in which the screen is (typically) represented by a frame buffer. The frame buffer is a memory segment that is copied by using the screen driver (driver for the screen hardware) and displayed by the screen. The system of the present invention produces the necessary information for controlling drawing on the screen to the operating system.

Turning now to Figure 7B, as shown by arrow "1", the operating system (through scrn_update_main module **710**) first updates the frame buffer for the screen. This updating may involve drawing the background for example, which may be displayed on every part of the screen to which data is not drawn from the information provided by the system of the present invention. The presence of

such a background supports the use of semi-transparent windows, which may be used for the creature or agent as described in greater detail below.

The Scrn_update_main module **710** then sends a request for updated data to a screen module **712**, which is part of the system of the present invention and which features a hook for communicating with the operating system. The screen module **712** then sends a request to each application window, shown as an agentWindow **714**, of which a plurality may be present, for updated information about what should be drawn to the screen. If a change has occurred, such that an update is required, then the agentWindow **714** notifies the screen module **712** that the update is required. The screen module **712** then asks for the location and size of the changed portion, preferably in two separate requests (shown as arrows 2.1.2.1 and 2.1.2.2 respectively), for which answers are sent by the agentWindow **714**.

The screen module **712** returns the information to the operating system through the scrn_update_main **710** in the form of an updated rectangle, as follows. The scrn_update_main **710** responds to the notification about the presence of an update by copying the frame buffer to a pre-buffer (process 3.1). The screen module **712** then draws the changes for each window into the pre-buffer, shown as arrow 3.2.1. The pre-buffer is then copied to the frame buffer and hence to the screen (arrow 3.3).

Figure 7C shows the class architecture for the system of the present invention for drawing on the screen. The screen module **712** and the agentWindow **714** are both shown. The class agentWindow **714** also communicates with three other window classes, which provide information regarding updating (changes to) windows: BackScreenWindow **716**, BufferedWindow **718** and DirectAccessWindow **720**. The BufferedWindow **718** has two further window classes with which it communicates: TransBufferedWindow **722** and PreBufferedWindow **724**.

Section 2: Action Selection System

This Section describes a preferred embodiment of an action selection system according to the present invention, including but not limited to a description of optional action selection according to incentive(s)/disincentive(s), and so forth. In order to assist in explaining how the actions of the intelligent agent are selected, an initial explanation is provided with regard to the structure of the intelligent agent, and the interactions of the intelligent agent with the virtual environment which is provided by the system of the present invention.

Figure 8A describes an exemplary structure of the intelligent agent and Figure 8B includes an exemplary sequence diagram for the operation of the intelligent agent. As shown with regard to Figure 8A, an intelligent agent **800** includes a plurality of classes. The main class is an AICreature **802**, which includes information about the intelligent agent such as its state, personality, goals etc, and also information about the appearance of the creature which visually represents the agent, such as location, color, whether it is currently visible and so forth.

The AICreature **802** communicates with World **804**, which is the base class for the virtual environment for the intelligent agent. The World **804** in turn communicates with the classes which comprise the virtual environment, of which some non-limiting examples are shown. World **804** preferably communicates with various instances of a WorldObject **806**, which represents an object that is found in the virtual environment and with which the intelligent agent may interact. The World **804** manages these different objects and also receives information about their characteristics, including their properties such as location and so forth. The World **804** also manages the properties of the virtual environment itself, such as size, visibility and so forth. The visual representation of the WorldObject **806** may use two dimensional or three dimensional graphics, or a mixture thereof, and may also use other capabilities of the mobile information device, such as sound production and so forth.

The WorldObject **806** itself may represent an object which belongs to one of several classes. This abstraction enables different object classes to be added to or removed from the virtual environment. For example, the object may be a "ball" which for example may start as part of a menu and then be "removed" by the creature in order to play with it, as represented by a MenuBallObject **808**. A GoodAnimalObject **810** also communicates with the WorldObject **806**; in turn, classes such as a FoodObject **812** (representing food for the creature), a BadAnimalObject **814** (an animal which may annoy the creature and cause them to fight for example) and a HouseObject **816** (a house for the creature) preferably communicate with the GoodAnimalObject **810**. The GoodAnimalObject **810** includes the functionality to be able to draw objects on the screen and so forth, which is why other classes and objects preferably communicate with the GoodAnimalObject **810**. Of course, many other classes and objects are possible in this system, since other toys may optionally be provided to the creature, for example.

The WorldObject **806** may also relate to the state of the intelligent agent,

for example by providing a graded input to the state. This input is graded in the sense that it provides an incentive to the intelligent agent or a disincentive to the intelligent agent; it may also have a neutral influence. The aggregation of a plurality of such graded inputs enables the state of the intelligent agent to be determined. As described with regard to the sequence diagram of Figure 8B, and also the graph search strategy and action selection strategy diagrams of Figures 9A and 9B respectively, the graded inputs are preferably aggregated in order to maximize the reward returned to the intelligent agent from the virtual environment.

These graded inputs may also include input from the user in the form of encouraging or discouraging feedback, so that the intelligent agent has an incentive or disincentive, respectively, to continue the behavior for which feedback has been provided. The calculation of the world state with respect to feedback from the user is performed as follows:

Grade = (weighting_factor * feedback_reward) + ((1-weighting_factor) * world_reward)

In which the feedback_reward results from the feedback provided by the user and the world_reward is the aggregated total reward from the virtual environment as described above; weighting_factor is a value between 0 and 1, which indicates the weight of the user feedback as opposed to the virtual environment (world) feedback.

Non-limiting examples of such reward for the agent's action include positive or negative feedback on the agent's suggestion; provision of a world object such as a ball or food to the agent; telephone usage duration; user teaching duration; and the like. Each of these examples can be assigned a predetermined score, and the agent's action can be restricted or expanded according to a corresponding accumulated score. For example, positive and negative feedback provided by the user may be assigned positive and negative point values, respectively; encountering an enemy or bad animal: -20 points; obtaining a food, toy or house object: +5 points; low battery alarm: -1 point; correct and incorrect answers, when the agent teaches the user: +1 point and -1 point, respectively; inactivity for 20 minutes: -1 point; wrong dialing: -1 point; SMS use: +1 point; and the like. The above examples may be applied in other ways.

Figure 8B shows an illustrative sequence diagram for an exemplary set of interactions between the virtual world and the intelligent agent of the present invention. The sequence starts with a request from a virtual world module **818** to the AICreature **802** for an update on the status of the intelligent agent. A

virtual world module **818** controls and manages the entire virtual environment, including the intelligent agent itself.

The intelligent agent then considers an action to perform, as shown by arrow 1.1.1. The action is preferably selected through a search (arrow 1.1.1.1) through all world objects, and then recursively through all actions for each object, by interacting with the World **804** and the WorldObject **806**. The potential reward for each action is evaluated (arrow 1.1.1.1.1.1) and graded (arrow 1.1.1.1.1.1.2). The action with the highest reward is selected. The overall grade for the intelligent agent is then determined and the AICreature **802** performs the selected action.

The Virtual_world **818** then updates the location and status of all objects in the world, by communicating with the World **804** and the WorldObject **806**.

The search through various potential actions may optionally be performed according to one or more of a number of different methods. Figures 9A and 9B show two exemplary methods for selecting an action according to the present invention.

Figure 9A shows an exemplary method for action selection, termed herein a rule based strategy for selecting an action. In stage 1, the status of the virtual environment is determined by the World state. A World Event occurs, after which the State Handler which is appropriate for that event is invoked in stage 2. The State Handler preferably queries a knowledge base in stage 3. The knowledge base may be divided into separate sections and/or separate knowledge bases according to the State Handler which has been invoked. In stage 4, a response is returned to the State Handler.

In stage 5, rule base validation is performed, in which the response (and hence the suggested action which in turn brings the intelligent agent into a specific state) is compared against the rules. If the action is not valid, then the process returns to stage 1. If the action is valid, then in stage 6 the action is generated. The priority for the action is then determined in stage 7; more preferably, the priority is determined according to a plurality of inputs, including but not limited to, an action probability, an action utility and a user preference. In stage 8, the action is placed in a queue for the action manager. In stage 9, the action manager retrieves the highest priority action, which is then performed by the intelligent agent in stage 10.

Figure 9B shows an exemplary action selection method according to a graph search strategy. Again, in stage 1 the process begins by determining the state of the world (virtual environment), including the state of the intelligent agent

and of the objects in the world. In stage 2, the intelligent agent is queried. In stage 3, the intelligent agent obtains a set of legal (permitted or possible) actions for each world object; preferably each world object is queried as shown.

The method now branches into two parts. A first part, shown on the right, is performed for each action path. In stage 4, an action to be performed is simulated. In stage 5, the effect of the simulation is determined for the world, and is preferably determined for each world object in stage 6. In stage 7, a grade is determined for the effect of each action.

In stage 8, the state of the objects and hence of the world is determined, as is the overall accumulated reward of an action. In stage 9, the effect of the action is simulated on the intelligent agent; preferably the effect between the intelligent agent and each world object is also considered in stage 10.

Turning now to the left branch of the method, in stage 11, all of this information is preferably used to determine the action path with the highest reward. In stage 12, the action is generated. In stage 13, the action priority is set, preferably according to the action grade or reward. In stage 14, the action is placed in a queue at the action manager, as in Figure 9A. In stage 15, the action is considered by the action manager according to priority; the highest priority action is selected, and is executed in stage 16.

Next, a description is provided of an exemplary action execution method and structure. Figure 10 shows a sequence diagram of an exemplary action execution method according to the present invention. A handler **1000** send a goal for an action to an action module **1002** in arrow 1, which features a base action interface. The base action interface enables the action module **1002** to communicate with the handler **1000** and also with other objects in the system, which are able to generate and post actions for later execution by the intelligent agent, shown here as a FloatingAgentApp **1006**. These actions are managed by an action manager **1004**.

The action manager **1004** has two queues containing action objects. One queue is the ready for execution queue, while the other queue is the pending for execution queue. The latter queue may be used for example if an action has been generated, but the internal state of the action is pending so that the action is not ready for execution. When the action state matures to be ready for execution, the action is preferably moved to the ready for execution queue.

An application manager **1008** interacts with the FloatingAgentApp **1006** for executing an action, as shown in arrow 2. The FloatingAgentApp **1006** then requests the next action from the action manager **1004** (arrow 2.1); the action

itself is provided by the action module **1002** (arrow 2.2.1). Actions are enqueued from the handler **1000** to the action manager **1004** (arrow 3). Goals (and hence at least a part of the priority) are set for each action by communication between the handler **1000** and the action module **1002** (arrow 4). Arrows 5 and 6 show the harakiri () method, described in greater detail below.

As previously described, the actions are queued in priority order. The priority is determined through querying the interface of the action module **1002** by the action manager **1004**. The priority of the action is determined according to a calculation which includes a plurality of parameters. For example, the parameters may include the priority as derived or inferred by the generating object, more preferably based upon the predicted probability for the success of the action; the persistent priority for this type of action, which is determined according to past experience with this type of action (for example according to user acceptance and action success); and the goal priority, which is determined according to the user preferences.

One optional calculation for managing the above parameters is as follows:

P(all) = P(action probability) * ((P(persistent priority) + P(action goal)/10))/2)

Complementary for the priority based action execution, each action referably has a Time To Live (ttl) period; this ttl value stands for the amount of execution time passed between the time when the action was posted in the ready queue and the expiration time of this action. If an action is ready but does not receive a priority for execution until its ttl has expired, the action manager **1004** preferably invokes the method harakiri(), which notifies the action that it will not be executed. Each such invocation of harakiri() preferably decreases the priority of the action until a threshold is reached. After this threshold has been reached, the persistent priority starts to increase. This model operates to handle actions that were proposed or executed but failed since the user aborted the action. The persistent priority decreases by incorporating the past experience in the action priority calculation.

This method shows how actions that were suggested or executed adapt to the specific user's implicit preferences in realtime.

This model is not complete without the harakiri() mechanism since if an action persistent priority reduces, so the action does not run, it needs to be allowed to either be removed or else possibly run again, for example if the user preferences change. After several executions of harakiri(), the action may regain the priority to run.

The previous Sections provide infrastructure, which enables various actions and mechanisms to be performed through the adaptive system of the present invention. These actions and mechanisms are described in greater detail below.

Section 3: Emotional System

This Section describes a preferred embodiment of an emotional system according to the present invention, including but not limited to a description of specific emotions and their intensity, which are combinable to form an overall mood. The emotional system can also includes a mechanism for allowing moods to change as well as for controlling one or more aspects of such a change, such as the rate of change for example.

Figures 11A-11C feature diagrams for describing an exemplary, illustrative implementation of an emotional system according to the present invention. Figure 11A shows an exemplary class diagram for the emotional system, while Figures 11B and 11C show exemplary sequence diagrams for the operation of the emotional system according to the present invention.

As shown with regard to an emotional system **1100** according to the present invention, the goal class (goal **1102**) represents an abstract goal of the intelligent agent. A goal is something which the intelligent agent performs as an action to achieve. The goal **1102** is responsible for creating emotions based on certain events that are related to the state of the goal and its chances of fulfillment.

The goal **1102** interacts with the AICreature **802** (previously described with regard to Figure 8). Briefly, the intelligent agent seeks to fulfill goals, so the interactions between the AICreature **802** are required in order to determine whether or not goals have been fulfilled, which in turn impact the emotional state of the intelligent agent.

The emotional state itself is handled by the class EmotionalState **1104**, which in turn is connected to the class Emotion **1106**. The Emotion **1106** is itself preferably connected to classes for specific emotions such as the anger class AngerEmotion **1108** and the joy class JoyEmotion **1110**. The EmotionalState **1104** is also preferably connected to a class which determines the pattern of behavior, the BehavioralPatternMapper **1112**.

The proactive user interface creates emotions of the agent through the emotional system when the likelihood of success (LOS) of the abstract goal of the intelligent agent increases or decreases and when the likelihood of failure (LOF) thereof increases or decreases. When LOS increases, then the hope emotion is

generated. When LOS decreases, the despair emotion is generated. When LOF increases, the fear emotion is preferably generated, and when LOF decreases, then the joy emotion is generated.

Success or failure of a goal has a significant effect on the goal state and generated emotions. When a goal fails, despair is generated, and if the likelihood of success was high, frustration is also generated (since expectation of success was high).

When a goal succeeds, joy is generated, and if expectation and accumulated success were high, then pride is generated.

The Emotion **1106** is a structure that has two properties, which are major and minor types. The major type describes the high level group to which the minor emotion belongs, preferably including POSITIVE_EMOTION and NEGATIVE_EMOTION. Minor types preferably include JOY, HOPE, GLOAT, PRIDE, LIKE, ANGER, HATE, FEAR, FRUSTRATION, DISTRESS, DISAPPOINTMENT. Other properties of the emotion are the intensity given when generated, and the decay policy (i.e. the rate of change of the emotion).

The next phase after emotion generation is performed by the EmotionalState class **1104** that accumulates emotions which were generated over time by the intelligent agent. This class represents the collection of emotion instances that defines the current emotional state of the intelligent agent. The current emotional state is defined by maintaining a hierarchy of emotion types, which are then generalized by aggregation and correlation. For example, the minor emotions are aggregated into a score for POSITIVE_EMOTION and a score for NEGATIVE_EMOTION; these two categories are then correlated to GOOD/BAD MOOD, which describes the overall mood of the intelligent agent.

The EmotionalState class **1104** is queried by the intelligent agent floating application; whenever the dominant behavior pattern changes (by emotions generated, decayed and generalized in the previously described model), the intelligent agent expresses its emotional state and behaves according to that behavioral pattern. The intelligent agent can expresses its emotional state using one or more of the text communication engine (described in greater detail below), three dimensional animation, facial expressions, two dimensional animated effects and sounds.

Figure 11B is an exemplary sequence diagram for generation of an emotion by the emotional system according to the present invention. As shown, the application manager **502** sends a step to the FloatingAgentApp **1006** in arrow 1. The FloatingAgentApp **1006** then determines the LOF (likelihood of failure)

by querying the goal class **1102** in arrow 1.1. The goal **1102** then determines the LOF; if the new LOF is greater than the previously determined LOF, fear is preferably generated by a request to emotion class **1106** in arrow 1.1.1.1. The fear emotion is also added to the emotional state by communication with EmotionalState **1104** in arrow 1.1.1.2.

Next, the application manager **502** sends another step (arrow 2) to the FloatingAgentApp **1006**, which determines the LOS (likelihood of success) by again querying the goal **1102** in arrow 2.1. The goal **1102** then determines the LOS; if the new LOS is greater than the previously determined LOS, hope is preferably generated by a request to emotion class **1106** in arrow 2.1.1.1. The hope emotion is also added to the emotional state by communication with EmotionalState **1104** in arrow 2.1.1.2.

Arrow 3 shows the application manager **502** sending another step to the FloatingAgentApp **1006**, which requests determination of emotion according to the actual outcome of an action. If the action has failed and the last LOS was greater than some factor, such as 0.5, which indicated that success was expected, then the FloatingAgentApp **1006** causes the Goal **1102** to have despair generated by the Emotion **1106** in arrow 3.1.1.1. The despair emotion is also added to the emotional state by communication with the EmotionalState **1104** in arrow 3.1.1.2. Also, if the action failed (regardless of the expectation of success), distress is generated by the Emotion **1106** in arrow 3.1.2. The distress emotion is also added to the emotional state by communication with the EmotionalState **1104** in arrow 3.1.3.

Next, the application manager **502** sends another step (arrow 4) to the FloatingAgentApp **1006**, which updates emotions based on actual success by sending a message to goal **1102** in arrow 4.1. The goal **1102** then causes joy to preferably be generated by a request to the emotion class **1106** in arrow 4.1.1. The joy emotion is also added to the emotional state by communication with the EmotionalState **1104** in arrow 4.1.2.

If actual success is greater than predicted, then the goal **1102** preferably causes pride to be generated by a request to the emotion class **1106** in arrow 4.1.3.1. The pride emotion is also added to the emotional state by communication with the EmotionalState **1104** in arrow 4.1.3.2.

Figure 11C is an exemplary sequence diagram for expressing an emotion by the emotional system according to the present invention. Such expression is governed by the user preferences. The application manager **502** initiates emotional expression by sending a step (arrow 1) to the FloatingAgentApp **1006**,

which queries the bp_mapper **1108** as to the behavioral pattern of the intelligent agent in arrow 1.1. If the dominant behavior has changed, then the FloatingAgentApp **1006** sends a request to the bp_display **1110** to set the behavioral pattern (arrow 1.2.1). The bp_display **1110** controls the actual display of emotion. The FloatingAgentApp **1006** then requests an action to be enqueued in a message to action manager **1004** (arrow 1.2.2).

The application manager **502** sends another step (arrow 2) to the FloatingAgentApp **1006**, which requests that the action be removed from the queue (arrow 2.1) to the action manager **1004**, and that the action be performed by the bp_display **1110**.

The following table shows a non-limiting example of the definition of actions that can be taken by the agent according to the levels of reward points obtained by the agent.

[Table 1]

Level	Basic Actions	Special Actions
5 (high)	sitting sleeping sniffing barking being sad being happy	1st-type dancing, running, 2nd-type jumping, 2nd-type flying
4		2nd-type dancing, running, jumping, flying
3		2nd-type happy, thinking, running, jumping
2		thinking, 1st-type waiting, 2nd-type angry, 2nd-type waiting
1 (low)		2nd-type sad, thinking, angry, 2nd-type waiting

In this example, the agent can perform the basic actions such as sitting, sleeping, sniffing, barking, being sad, and being happy, irrespective of reward points obtained by the agent. The level of the agent increases by one level as the reward points increase by +200 points, whereas it decreases by one level as the reward points decrease by +200 points (i.e., increase by -200 reward points). A different number of reward points may be set as a basis for the level change. In case the basic level of the agent is set to the 3rd level, the agent can take the basic actions and additional special actions corresponding to the 3rd level, such as 2nd-type happy, thinking, running, and jumping, as shown in Table 1. In this

case, if the agent obtains +200 reward points, the level of the agent becomes the 4th level, so that the agent can take the basic actions and additional special actions corresponding to the 4th level, such as 2nd-type dancing, running, jumping, and flying. The action selection system described above determines which action should be taken from among actions available at each level, while the emotional system controls emotional expressions corresponding respectively to the actions. In addition to such actions and emotional expressions, the agent performs an emotional expression based on the following communication with the user.

Section 4: Communication with the User

This Section describes a preferred embodiment of a communication system for communication with the user according to the present invention, including but not limited to textual communication, audio communication and graphical communication. For the purpose of description only and without any intention of being limiting, textual communication is described as an example of these types of communication. The communication with the user described in this section can be used for (but is not limited to) the agent's suggestions, provision of the information of the user, or the agent's emotional expressions.

Figure 12A is an exemplary sequence diagram for textual communication according to the present invention. A text engine **1200** is responsible for generating text that is relevant to a certain event and which can be communicated by the intelligent agent. The text engine **1200** includes a natural language generation of sentences or short phrases according to templates that are predefined and contain place holders for fillers. Combining the templates and the fillers together enable the text engine **1200** to generate a large number of phrases, which are relevant to the event to which the template belongs.

This framework can be extensible for many new and/or changing events or subjects because additional templates can also be added, as can additional fillers.

As shown in Figure 12A, the FloatingAgentApp **1006** communicates with the text engine **1200** by first sending a request to generate text, preferably for a particular event (arrow 1). The text engine **1200** selects a template, preferably from a plurality of templates that are suitable for this event (arrow 1.1). The text engine **1200** also selects a filler for the template, preferably from a plurality of fillers that are suitable for this event (arrow 1.2.1). The filled template is then returned to the FloatingAgentApp **1006**.

The following provides an example of generation of text for a mood

change event, which is that the intelligent agent is now happy, with some exemplary, non-limiting templates and fillers. Examples of the templates are as follows:

Happy template 1: "%noun1 is %happy_adj2"

Happy template 2: "%self_f_pronoun %happy_adj1"

Examples of the fillers are as follows:

%noun1 = {"the world", "everything", "life", “this day”, “the spirit”}

%happy_adj1 = {"happy", "joyful", "glad", "pleased", "cheerful", "in high spirits", "blissful", "exultant", "delighted", "cheery", "jovial", "on cloud nine" }

%happy_adj2 = {"nice", "beautiful", "great", "happy", "joyful", "good", "fun"}

%self_f_pronoun = {"I am", "I'm", "your intelligent agent", "your agent friend"}

Examples of some resultant text communication phrases from combinations of templates and fillers as follows:

I'm cheerful
the spirit is joyful
I am exultant
life is beautiful
life is good
I'm pleased
I'm jovial
I am joyful
the world is joyful
I'm glad
the spirit is joyful
the spirit is happy
the world is nice
I am happy

As another non-limiting example, a missed call template could be constructed as follows:

%user missed a call from %missed %reaction

In this example, the user's name is used for %user; the name or other identifier (such as telephone number for example) is entered

to %missed; %reaction is optional and is used for the reaction of the intelligent agent, such as expressing disappointment for example (e.g. "I'm sad").

As shown by these examples, the text engine **1200** can generate relevant sentences for many events, from missed call events to low battery events, making the user's interaction with the mobile information device richer and more understandable.

Figure 12B shows a non-limitting example of an emotional expression "I am happy" that the agent performs in a mobile phone.

EXAMPLE 3: EVOLUTION SYSTEM FOR AN INTELLIGENT AGENT

This example describes a preferred embodiment of an evolution system according to the present invention, including but not limited to a description of DNA (DeoxyriboNucleic Acid) for the creature or avatar according to a preferred embodiment of the present invention, and also a description of an optional gene studio according to the present invention. The evolution system enables the creature or avatar to "evolve", that is, to alter at least one aspect of the behavior and/or appearance of the creature. This example is described as being operative with the intelligent agent described in example 2, but this description is for the purposes of illustration only and is not meant to be limiting in any way. In other words, the evolution system for the intelligent agent described in this example may be used (but not necessarily) in conjunction with the learning module and the action selection system described above, thereby making it possible to implement a system that can determine the user's preferences and actively evolve without requesting the user's behavior.

Evolution (change) of the intelligent agent is described herein with regard to both tangible features of the agent, which are displayed by the avatar or creature, and non-tangible features of the agent, which affect the behavior of the avatar or creature.

Figure 13A shows an exemplary evolution class diagram **1800**. The genetic model described in the class diagram allows for various properties of the intelligent agent to be changed, including visual as well as functional properties. The model includes a CreatureDNA class **1802** that represents the DNA structure. The DNA structure is a vector of available genes and can preferably be extended to incorporate new genes. A gene is a parameter with a range of possible values (i.e. genotype). The gene is interpreted by the system according to the present

invention, such that the expression of the data in the gene is its genotype. For example the head gene is located as the first gene in the DNA, and its value is expressed as the visual structure of the creature's head, although preferably the color of the head is encoded in another gene.

In order to evolve the intelligent agent to achieve a specific DNA instance that pleases the user, the genetic model according to the present invention implements hybrid and mutate genetic operations that modify the DNA. The CreatureProxy class **1804** is responsible for providing an interface to the DNA and to the genetic operations for the system classes. CreatureProxy **1804** holds other non-genetic information about the intelligent agent (i.e. name, birth date, and so forth).

The EvolutionMGR class **1806** manages the evolutions of the intelligent agent and provides an interface to the CreatureProxy **1804** of the intelligent agent and its genetic operations to applications.

The EvolutionEngine class **1808** listens to evolution events that may be generated from time to time, for indicating that a certain genetic operation should be invoked and performed on the intelligent agent DNA. The DNA structure is given below.

The CreatureDNA **1802** preferably listens to such evolution events from the EvolutionEvent **1810**. The following is an algorithm defining an examplory DNA structure.

DNA structure

```
#ifndef __CREATURE_DNA__
#define __CREATURE_DNA__

#include "CreatureDefs.h"
#include "CommSerializable.h"

#define GENE_COUNT 19
#define BASE_COLOR_GENE 8

typedef  struct internal_dna
{
  unsigned char gender;
  unsigned char head;
```

```
    unsigned char head_color;
    unsigned char head_scale;
    unsigned char body;
    unsigned char body_color;
    unsigned char body_scale;
    unsigned char hand;
    unsigned char hand_color;
    unsigned char hand_scale;
    unsigned char tail;
    unsigned char tail_color;
    unsigned char tail_scale;
    unsigned char leg;
    unsigned char leg_color;
    unsigned char leg_scale;
    unsigned char dexterity;
    unsigned char efficiancy;
    unsigned char interactive;
    unsigned char base_color;
} internal_dna;

typedef internal_dna p_internalDna;

/**
 * This class represents the Creature DNA structure.
 * The DNA holds all the data about the Creature body
parts and some
 * personality and functional qualities
 */
class CreatureDNA /*: public CommSerializable*/
{

public:
   static const int  gene_count;

   /**
    * defualt constructor, DNA is initialized to zero
    */
```

```
    CreatureDNA();

    /*
     * Copy constructor
     * @param other - the DNA to copy
     */
    CreatureDNA(const CreatureDNA &other);

    /**
     * Initialization function, should be called if the
constructor was not
     * called.
     */
    void init();

    /**
     * Randomizes the DNA data
     *
     */
    void randomizeDna();


    /**
     * The DNA actual data
     */
    union {

      internal_dna genes;
      unsigned char data[GENE_COUNT];
    };

    /**
     * Range of gender gene
     */
    static const int GENDER_RANGE;
    /**
    * Range of type gene
```

```
    */
   static const int TYPE_RANGE;
   /**
    * Range of color gene
    */
   static const int COLOR_RANGE;
   /**
    * Range of scale gene
    */
   static const int SCALE_RANGE;
   /**
    * Range of character genes
    */
   static const int CHARECTER_RANGE;

   static const int BASE_COLOR_RANGE;
private:

   /**
    * Location of scale gene in the type,color,scale
triplet
    */
   static const int SCALE_LOCATION;
};


#endif /*__CREATURE_DNA__*/
```

Intelligent agent DNA construction is preferably performed as follows. When providing a version of a "living" mobile phone, the DNA is preferably composed from a Gene for each Building Block of the intelligent agent. The building block can be a visual part of the agent, preferably including color or scale (size of the building block), and also can include a non visual property that relate to the functionality and behavior of the intelligent agent. This model of DNA composition can be extended as more building blocks can be added and the expression levels of each building block can increase.

The construction of an intelligent agent from the DNA structure is

performed with respect to each gene and its value. Each gene (building block) value (expression level) describes a different genotype expressed in the composed agent. The basic building blocks of the visual agent are modeled as prototypes, hence the amount of prototypes dictate the range of each visual gene. It is also possible to generate in runtime values of expressed genes not relaying on prototypes, for example color gene expression levels can be computed as indexes in the host platform color table, or scale also can be computed with respect to the host screen size, to obtain genotypes that are independent of predefined prototypes. The prototype models are decomposed and then a non-prototype agent is recomposed according to the gene values of each building block.

The following example provides an illustrative non-limiting explanation of this process. For simplicity and clarity, color and scale, and other non visual genes, are not included, but the same process also applies to these genes.

Without taking into consideration the gender gene, a 16 prototype and 5 building block version of DNA may optionally be given as follows:

DNA_0={[head,0:15] , [body,0:15] , [legs, 0:15] , [hands,0:15] , [tail, 0:15]}

Each of the 5 building blocks has 16 different possible genotypes according to the building block gene values that are derived from the number of prototype models. When composing the intelligent agent, the right building block is taken according to the value of that building block in the DNA , which is the value of its respective gene.

For example a specific instance of the DNA scheme described above can be:

DNA = { [3],[5],[10],[13],[0] }

The variety of possible intelligent agent compositions in this simple DNA version is:

$V_0 = (16) * (16) * (16) * (16) * (16) = (16)^5 = 1048576$

If a base color gene for describing the general color of the intelligent agent (i.e. green, blue, and so forth) is added, with expression level of possible 16 base colors, the following variety is obtained:

DNA_1 =

{[head,0:15] , [body,0:15] , [legs, 0:15] , [hands,0:15] , [tail, 0:15], [bs_color,0:15]}

The variety then becomes:
$V_1 = V_0 * 16 = (16)^6 = 16777216$

If an intensity gene for the base color gene (i.e from light color to dark color) is added to this DNA version, with an expression level of possible 16 intensities of the base color, the following variety is preferably obtained:

$DNA_2 =$
{[head,0:15] , [body,0:15] , [legs, 0:15] , [hands,0:15] , [tail, 0:15], [bs_color,0:15] , [intensity,0:15]}

The variety calculation is:
$V_2 = V_1 * 16 = (16)^7 = 268435456$

The present invention can express a variety of agent combination types as described above without storing the information of each of the completed combination types. According to the present invention, only with both the information of building blocks of the combination types and the information of a method for combining the building blocks is it possible to make a variety of agent combination types as described above. Accordingly, in case the agent is used with a portable computational device, it is possible for each of the computational device users to hold a substantially-unique type of agent, thanks to diversity in the combination methods.

On the other hand, according to another embodiment, 16 prototype and 5 building block version of the above DNA may optionally be given as follows:

DNA_0={[$head_0$,0:15] , [$body_0$,0:15] , [$legs_0$, 0:15] , [$hands_0$,0:15] , [$tail_0$, 0:15]}, when 0<Tg<Tth, and

DNA_1={[$head_1$,0:15] , [$body_1$,0:15] , [$legs_1$, 0:15] , [$hands_1$,0:15] , [$tail_1$, 0:15]}, when Tg≥Tth,

(where "Tg" denotes a growth time of the agent and "Tth" denotes a threshold time).

In this example, the threshold time Tth is set to 2 weeks, but may also be set differently. The growth time Tg of the agent indicates a time period from when the computational device user resets the agent or starts using the agent for the first time, to the current time. In this case, a trait expressed by the DNA_0

may be selected from a combination of first building blocks if the growth time of the agent is less than 2 weeks, whereas a trait expressed by the DNA_1 may be selected from a combination of second building blocks if the growth time is 2 weeks or more. If the first building-block combination is set to represent the appearance of a younger agent, and the second building-block combination is set to represent the appearance of a more grown-up agent, it is possible to implement the appearance of an agent with the same genes that automatically grows as time goes on. After the user starts using the agent for the first time, the growth of the agent occurs only with the lapse of time. In case two threshold times are set, the growth of the agent is composed of three steps. It is also possible to set more than two threshold times.

A variety of genetic operations may be performed on the DNA, as described with regard to Figures 13B and 13C, which show a mutation sequence diagram and a hybridization sequence diagram, respectively.

As shown in Figure 13B, the basic mutation operation randomly selects a gene from the gene set that can be mutated, which may be the entire DNA, and then change the value of the selected gene within that gene's possible range (expression levels). The basic operation can be performed numerous times.

A mutate application **1812** sends a request to the EvolutionMGR **1806** (arrow 1.1) to create a mutant. The EvolutionMGR class **1806** passes this request to the CreatureProxy **1804**, for a number of mutants (this value may be given in the function call; arrow 1.1.1). For each such mutant, the CreatureProxy **1804** preferably selects a random gene (arrow 1.1.1.1.1) and changes it to a value that is still within the gene's range (arrow 1.1.1.1.2). The mutant(s) are then returned to the mutate application **1812**, and are preferably displayed to the user, as described in greater detail below with regard to Example 4.

If the user approves of a mutant, then the mutate application **1812** sends a command to replace the existing implementation of the agent with the new mutant (arrow 2.1) to the EvolutionMGR **1806**. The EvolutionMGR **1806** then sets the DNA for the creature at the CreatureProxy **1804** (arrow 2.1.1), which preferably then updates the history of the agent at the agent_history **1814** (arrow 2.1.1.1).

Figure 13C shows an exemplary sequence diagram for the basic hybrid operation (or cross-over operation), which occurs when two candidate DNAs are aligned one to the other. Both the two candidate DNAs may be obtained from the intelligent agent system. One of the two candidate DNAs may also be

obtained from an intelligent agent system for another mobile information device. For example, in the case of an intelligent agent for a networked mobile information device in Example 5 described below, one of the two candidate DNAs may be obtained from an intelligent agent for a second mobile information device of a second user via a short message service (SMS).

Taking into consideration the gender gene, the above DNA may be represented as follows:

DNA_0={[gender, 0:1], [head,0:15] , [body,0:15] , [legs, 0:15] , [hands,0:15] , [tail, 0:15]}

The gender gene determines whether the hybrid operation is allowed. Preferably, the hybrid operation is allowed only between different gender genes. However, if the gender gene is not taken into consideration, the hybrid operation may be allowed in any case. For the hybrid operation, one or more cross over points located on the DNA vector are preferably selected (the cross-over points number can vary from 1 to the number of genes in the DNA; this number may be randomly selected). The operation of selecting the crossover points is called get_cut_index. At each cross over point, the value for the DNA is selected from one of the existing DNA values. This may be performed randomly or according to a count called a cutting_index. The gender-gene hybrid operation is performed by selecting one of the corresponding two genes. The result is a mix between the two candidate DNAs. The basic hybrid operation can be performed numerous times with numerous candidates.

As shown, a HybridApp **1816** sends a command to the EvolutionMGR **1806** to begin the process of hybridization. The process is optionally performed until the user approves of the hybrid agent or aborts the process. The EvolutionMGR **1806** starts hybridization by sending a command to obtain target DNA (arrow 2.1.1) from the CreatureProxy **1804**, with a number of cross-overs (hybridizations) to be performed. As shown, a cutting_index is maintained to indicate when to do a cross-over between the values of the two DNAs.

The hybrid agent is returned, and if the user approves, then the current agent is replaced with the hybrid agent, as described above with regard to the mutant process. In the end, the history of the agent at the agent_history **1814** is updated.

Hybridization may be performed with agent DNA that is sent from a source external to the mobile information device, for example in a SMS message, through infrared, BlueTooth or the Internet, or any other source. For the purpose of description only and without any intention of being limiting, this process is

illustrated with regard to receiving such hybrid DNA through an SMS message. The SMS message preferably contains the data for the DNA in a MIME type. More, the system of the present invention has a hook for this MIME type, so that this type of SMS message is automatically parsed for hybridization without requiring manual intervention by the user.

Figure 14 shows an exemplary sequence diagram of such a process. As shown, User 1 sends a request to hybridize the intelligent agent of User 1 with that of User 2 through Handset 1. User 2 can optionally approve or reject the request through Handset 2. If User 2 approves, the hybrid operation is performed between the DNA from both agents on Handset 1. The result is displayed to the requesting party (User 1), who may save this hybrid as a replacement for the current agent. If the hybrid is used as the replacement, then User 2 receives a notice and saves to the hybrid to the hybrid results collection on Handset 2.

EXAMPLE 4: USER INTERACTIONS WITH THE PRESENT INVENTION

This Example is described with regard to a plurality of representative, non-limiting, illustrative screenshots, in order to provide an optional but preferred embodiment of the system of the present invention as it interacts with the user.

Figure 15 shows an exemplary screenshot of the "floating agent", which is the creature or avatar (visual expression of the intelligent agent).

Figure 16 shows an exemplary screenshot of a menu for selecting objects for the intelligent agent's virtual world.

Figure 17A shows the Start Wizard application, which allows the user to configure and modify the agent settings, as well as user preferences.

Figure 17B-17F show exemplary screenshots of an initial setting mode for an agent after the start wizard is activated, where Figure 17B shows a screenshot of a setting mode for selecting the type of the agent; Figure 17C for selecting a color thereof; Figure 17D for selecting a name thereof; Figure 17E for selecting a personality thereof; and Figure 17F for indicating the completion of the agent setting.

One example of an action to be performed with the wizard is to Set Personality, to determine settings for the emotional system of the intelligent agent. Here, the user can configure the creature's personality and tendencies.

The user can determine the creature's setting by pressing the right arrow key in order to increase the level of the characteristic and in order to do the opposite and decrease the level of the various characteristics such as Enthusiasm,

Sociability, Anti_social behavior, Temper (level of patience), Melancholy, Egoistic behavior, and so forth.

The user is also able to set User Preferences, for example to determine how quickly to receive help. Some other non-limiting examples of these preferences include: communication (extent to which the agent communicates); entertain_user (controls agent playing with the user); entertain_self (controls agent playing alone); preserve_battery (extends battery life); and transparency_level (the level of the transparency of the creature).

The user also sets User Details with the start wizard, including but not limited to, user name, birthday (according to an optional embodiment of the present invention, this value is important in Hybrid SMS since it will define the "konghup" possibility between users, which is the ability to create a hybrid with a favorable astrology pattern; the konghup option is built according to suitable tables of horsocopes and dates), and gender. Here, the "konghup" (also called "goong-hap") is a Korean word used to describe marital harmony as predicted by a fortuneteller, and the konghup possibility can be defined as the possibility of a favorable astrology pattern for inter-personal relationship.

The user can also preferably set Creature Details.

Figure 18 shows an exemplary menu for performing hybridization through the hybrid application as previously described.

Figure 19A shows an exemplary screenshot for viewing a new creature and generating again, by pressing on the Generate button, which enables the user to generate a creature randomly. Figure 19B shows the resultant creature in a screenshot with a Hybrid button: pressing on this button confirms the user's creature selection and passes to the creature preview window.

The preview window allows the user to see the newly generated creature in three dimensions, and optionally to animate the creature by using the following options:

1. Navigation UP key: Zoom In and minimizes the size of the creature.
2. Navigation DOWN key: Zoom Out and maximizes the size of the creature.
3. Navigation LEFT key: Switch between the "Ok" and "Back" buttons.
4. Navigation RIGHT key: Switch between the "Ok" and "Back" buttons.
5. Ok key (OK): Confirm selection.
6. Clear key (CLR): Exit the creature preview window to Living Mobile Menu.

7. End key: Exit the creature preview window to the main menu.
8. '0' key: Lighting and shading operation on the creature.
9. '1' key: Circling the creature to the left with the clock direction.
10. '2' key: Circling the creature in the 3D.
11. '3' key: Circling the creature to the right against the clock direction.
12. '5' Key: Circling the creature in the 3D.
13. '6' key: Animates the creature in many ways. Every new pressing on this key changes the animation type.

The animations that the creature can perform include but are not limited to, walking, sitting, smelling, flying, and jumping.

Figure 20 shows an exemplary screenshot of the hybrid history, which enables the user to review and explore the history of the creature's changes in the generations. The user can see the current creature and its parents, and also the parents of the parents. Preferably, for every creature there can be at most 2 parents. On the other hand, if a current DNA different from a first DNA (DNA 1) is created from the first DNA (DNA1), the creation can be set to indicate that mutation has occurred.

Figure 21 shows an exemplary screen shot of the Gene studio, with the DNA Sequence of the current creature. The gene studio also preferably gives the opportunity for the user to change and modify the agent's DNA sequence. The agent's DNA sequence displayed on the gene studio screen is preferably composed of a sequence of four letters A, G, C and T. The four letters represent the four bases constituting biological DNA. The present invention introduces the four letters so that the user becomes more familiar with the agent DNA. In the present invention, the four letters correspond to four numbers required to express the quaternary numbers. For example, if A=0, G=1, C=2, and T=3, then AA=0, AG=1, ..., and TT=15. If the agent has a DNA sequence composed of two letters for each building block as in the above Example 3, it is possible to express 16 (=4×4) different genotypes for each building block. Accordingly, if the agent has 5 building blocks, the agent can be defined as a DNA sequence composed of 10 (=2×5) letters, and the number of possible combinations of the agent is 16^5. Similarly, if the agent has 10 building blocks and a DNA sequence composed of 4 letters for each building block, it is possible to express 256 (=4×4×4×4) different genotypes for each building block, a DNA sequence required to define the agent is composed of 40 (=4×10) letters, and the number of possible combinations of the agent is 256^{10}. However, the present invention is not limited to these examples.

A description will now be given of an evolving agent system operating in conjunction with the learning module and the action selection system. As described above, the hybrid history or the information as to whether a mutation is selected is stored in the agent system. The learning module can determine preferences or tendencies of the user on the basis of the stored information, and the action selection system can provide an evolution event, according to the determined user preferences or tendencies, to the evolution class diagram. Information as to whether the user selects the result of the performance of the provided evolution event is stored in the agent system, so that the stored information is referred to when the next evolution event is provided.

EXAMPLE 5 –INTELLIGENT AGENT FOR A NETWORKED MOBILE INFORMATION DEVICE

This example relates to the use of an intelligent agent on a networked mobile information device, preferably a cellular telephone. The intelligent agent comprises an avatar for interacting with the user, and an agent for interacting with other components on the network, such as other mobile information devices, and/or the network itself. The avatar forms the user interface (or a portion thereof) and also has an appearance, which is more preferably three-dimensional. This appearance may be humanoid but may alternatively be based upon any type of character or creature, whether real or imaginary. The agent then handles the communication between the avatar and the mobile information device, and/or other components on the network, and/or other avatars on other mobile information devices. It should also be noted that although this implementation is described with regard to mobile information devices such as cellular telephones, the avatar aspect of the implementation (or even the agent itself) may be implemented with the adaptive system (Example 2) and/or proactive user interface (Example 1) as previously described.

The intelligent agent of the present invention is targeted at creating enhanced emotional experience by applying the concept of a "Living Device". This concept includes both emphases upon the uniqueness of the intelligent agent, as every living creature is unique and special in appearance and behavior, while also providing variety, such as a variety of avatar appearances to enhance the user's interaction with the living device. The avatar preferably has compelling visual properties, with suitable supplementary objects and surrounding environment.

The intelligent agent preferably displays intelligent decision making, with unexpected behavior that indicates its self-existence and independent learning. Such independent behavior is an important aspect of the present invention, as it has not been previously demonstrated for any type of user interface or interaction for a user and a computational device of any type, and has certainly not been used for an intelligent agent for a mobile information device. The intelligent agent can also evolve with time, as all living things, displaying visual changes. This is one of the most important "Living Device" properties.

The evolution step initiates an emotional response from the user of surprise and anticipation for the next evolution step.

Evolution is a visual change of the creature with respect to time. The time frame may be set to a year for example, as this is the lifecycle of midrange cellular telephone in the market. During the year or quarter, periodic changes preferably occur through evolution. The evolutionary path (adaptation to the environment) is a result of natural selection. The natural selection can be user driven (i.e. user decides if the next generation is better), although another option is a predefined natural selection process by developing some criteria for automatic selection.

The intelligent agent may be implemented for functioning in two "worlds" or different environments: the telephone world and the virtual creature world. The telephone (mobile information device) world enables the intelligent agent to control different functions of the telephone and to suggest various function selections to the user, as previously described. Preferably the intelligent agent is able to operate on the basis of one or more telephone usage processes that are modeled for the agent to follow. Another important aspect of the telephone world is emotional expressions that can be either graphic expressions such as breaking the screen or free drawing or facial and text expressions one or two relevant words for the specific case.

The virtual world is preferably a visual display and playground area, where objects other than the avatar can be inserted and the user can observe the avatar learning and interacting with them. The objects that are entered into the world can be predefined, with different behaviors resulting from the learning process. The user can give rewards or disincentives and be part of the learning process. In this respect, the intelligent agent (through the appearance of the avatar) may act as a type of virtual pet or companion (for example, act as a running puppy or a laughing person).

Some preferred aspects of the intelligent agent include but are not limited to, a 3D graphic infrastructure (with regard to the appearance of the avatar); the use of AI and machine learning mechanisms to support both adaptive and proactive behavior; the provision of gaming capabilities; the ability to enhance the usability of the mobile information device and also to provide specific user assistance; and provision of a host platform abstraction layer. Together, these features provide a robust, compelling and innovative content platform to support a plurality of AI applications all generically defined to be running on the mobile information device.

The avatar also preferably has a number of important visual aspects. For example, the outer clip size may optionally be up to 60 x 70 pixels, although a different resolution may be selected according to the characteristics of the screen display of the mobile information device. The avatar is preferably represented as a 3D polygonal object with several colors, but in any case preferably has a plurality of different 3D visual characteristics, such as shades, textures, animation support and so forth. These capabilities may be provided through previously created visual building blocks that are stored on the mobile information device. The visual appearance of the avatar is preferably composed in runtime.

The avatar may start "living" after a launch wizard, taking user preferences into account (user introduction to the living device). In addition to evolution, the avatar may display small visual changes that represent mutations (color change / movement of some key vertices in a random step). Visual evolution step is preferably performed by addition / replacement of a building block. The avatar can preferably move in all directions and rotate, and more is a fully animated 3D character.

The avatar is preferably shown as floating over the mobile information device display with the mobile information device user interface in the background, but may also be dismissed upon a request by the user. The avatar is preferably able to understand the current user's normal interaction with the mobile information device and tries to minimize forced hiding/dismissal by the user.

According to optional but preferred embodiments of the present invention, the avatar can be programmed to "move" on the screen in a more natural, physically realistic manner. For example, various types of algorithms and parameters are available which attempt to describe physically realistic behavior and movement for controlling the movement of robots. Examples of such algorithms and parameters are described in "Automatic Generation of Kinematic

Models for the Conversion of Human Motion Capture Data into Humanoid Robot Motion", A. Ude et al., *Proc. First IEEE-RAS Int. Conf. Humanoid Robots (Humanoids 2000)*, Cambridge, MA, USA, September 2000 (hereby incorporated by reference as if fully set forth herein). This reference describes various human motion capture techniques, and methods for automatically translating the captured data into humanoid robot kinetic parameters. Briefly, both human and robotic motion are modeled, and the models are used for translating actual human movement data into data that can be used for controlling the motions of humanoid robots.

This type of reference is useful as it provides information on how to model the movement of the humanoid robot. Although the present invention is concerned with realistic movement of an avatar (virtual character being depicted three-dimensionally), similar models could optionally be used for the avatar as for the humanoid robot. Furthermore, a model could also be constructed for modeling animal movements, thereby permitting more realistic movement of an animal or animal-like avatar. More generally, the system can handle any given set of 3D character data generically.

These models could also be used to permit the movement of the avatar to evolve, since different parameters of the model could be altered during the evolutionary process, thereby changing how the avatar moves. Such models are also useful for describing non-deterministic movement of the avatar, and also for enabling non-deterministic movements to evolve. Such non-deterministic behavior also helps to maintain the interest of the user.

In order to implement these different functions of the avatar and/or intelligent agent, the intelligent agent may be constructed as described below with regard to Figures 7A-12B, although it should be noted that these figures only represent one exemplary implementation and that many different implementations are possible. Again, the implementation of the intelligent agent may incorporate or rely upon the implementations described in Examples 1 and 2 above.

Figure 22 is a block diagram of an intelligent agent system **2700** according to the present invention. As shown, a first user **2702** controls a first mobile information device **2704**, which for the purpose of this example may be implemented as a cellular telephone for illustration only and without any intention of being limiting. A second user **2706** controls a second mobile information device **2708**. The first mobile information device **2704** and the second mobile information device **2708** preferably communicate through a network **2710**, for example through messaging.

Each of the first mobile information device **2704** and the second mobile information device **2708** preferably features an intelligent agent, for interacting with their respective users **2702** and **2706** and also for interacting with the other intelligent agent. Therefore, as shown, the system **2700** enables a community of such intelligent agents to interact with each other, and/or to obtain information for their respective users through the network **2710**, for example.

The interactions of the users **2702** and **2706** with their respective mobile information devices **2704**, **2708** preferably include the regular operation of the mobile information device, but also add the new exciting functionalities of "living mobile phone". These functionalities can include the intelligent agent but also the use of an avatar for providing a user interface and also more preferably for providing an enhanced user emotional experience.

The intelligent agent preferably features an "aware" and intelligent software framework. The inner operation of such a system preferably involve several algorithmic tools, including but not limited to AI and ML algorithms.

The system **2700** may involve interactions between multiple users as shown. Such interactions increase the usability and enjoyment of using the mobile information device for the end-user.

Figure 23 shows the intelligent agent system of Figure 20 in more detail. As shown, a first intelligent agent **2800** is able to operate according to scenario data **2802** (such as the previously described knowledge base) in order to be able to take actions, learn and make decisions as to the operation of the mobile information device. The learning and development process of the first intelligent agent **2800** is supported by an evolution module **2804** for evolving as previously described. If the first intelligent agent **2800** communicates with the user through an avatar, according to a preferred embodiment of the present invention, then an animation module **2806** is used to support the appearance of the avatar.

The first intelligent agent **2800** may also communicate through the network (not shown) with a backend server **2808** and/or another network resource such as a computer **2810**, for example for obtaining information for the user.

The first intelligent agent **2800** may also communicate with a second intelligent agent **2812** as shown.

Figure 24 shows a block diagram of an exemplary implementation of an action selection system **2900** according to the present invention, which provides the infrastructure for enabling the intelligent agent to select an action.

The action selection system **2900** preferably features an ActionManager

2902 (see also Figure 10 for a description), which actually executes the action. A BaseAction interface **2904** provides the interface for all actions executed by the ActionManager **2902**.

Actions may use device and application capabilities denoted as an AnimationManager **2906** and a SoundManager **2908** that are necessary to perform the specific action. Each action aggregates the appropriate managers for the correct right execution.

The AnimationManager **2906** may also control a ChangeUIAction **2910**, which changes the appearance of the visual display of the user interface. In addition or alternatively, if an avatar is used to represent the intelligent agent to the user, the AnimationManager **2906** may also control a GoAwayFromObjectAction **2912** and a GoTowardObjectAction **2914**, which enables the avatar to interact with virtual objects in the virtual world of the avatar.

Figures 25A and 25B show two exemplary, illustrative non-limiting screenshots of the avatar according to the present invention on the screen of the mobile information device. Figure 25A shows an exemplary screenshot of the user interface for adjusting the ring tone volume through an interaction with the avatar. Figure 25B shows an exemplary screenshot of the user interface for receiving a message through an interaction with the avatar.

While the invention has been described with respect to a limited number of embodiments, it will be appreciated that many variations, modifications and other applications of the invention may be made.

WHAT IS CLAIMED IS:

1. A proactive user interface for a computational device having an operating system, the proactive user interface comprising:

(a) an interface unit for communication between a user of the proactive user interface and said operating system, said interface unit including an evolving agent enabling communication with the user; and

(b) a learning module for detecting at least one pattern of interaction of the user with said interface unit, and actively suggesting, to the user, options for evolving at least one function of the user interface according to said at least one pattern.

2. The proactive user interface of claim 1, wherein said at least one pattern is selected from the group consisting of a pattern determined according to at least one previous interaction of the user with said interface unit, a predetermined pattern, and a combination thereof.

3. The proactive user interface of claim 1, wherein said agent includes an avatar, and at least one characteristic of an appearance of said avatar is optionally alterable according to a user command.

4. The proactive user interface of claim 2, wherein said agent includes an avatar, and at least one characteristic of an appearance of said avatar is optionally alterable according to a user command.

5. The proactive user interface of claim 3, wherein at least one characteristic of an appearance of said avatar is alterable according to an automated evolutionary algorithm.

6. The proactive user interface of claim 4, wherein at least one characteristic of an appearance of said avatar is alterable according to an automated evolutionary algorithm.

7. The proactive user interface of claim 1, wherein said computational device is selected from a group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

8. The proactive user interface of claim 2, wherein said computational device is selected from a group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

9. The proactive user interface of claim 3, wherein said computational device is selected from a group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

10. The proactive user interface of claim 4, wherein said computational device is selected from a group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

11. The proactive user interface of claim 5, wherein said computational device is selected from a group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

12. The proactive user interface of claim 6, wherein said computational device is selected from a group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

13. The proactive user interface of claim 1, wherein said learning module comprises a knowledge base for storing information gathered as a result of interactions with the user and/or the operating system.

14. The proactive user interface of claim 13, wherein said knowledge base comprises a plurality of integrated knowledge items determined from the behavior of the user and from preprogrammed information.

15. The proactive user interface of claim 13, wherein said learning module further comprises a plurality of sensors for perceiving a state of the operating system.

16. The proactive user interface of claim 15, wherein said learning module further comprises a perception unit for processing output from said sensors to determine a state of the operating system and a state of said interface unit.

17. The proactive user interface of claim 13, wherein said learning module further comprises a reasoning system for updating said knowledge base and for learning an association between an alteration of said interface unit and a state of the operating system.

18. The proactive user interface of claim 13, wherein said learning module maximizes a percentage of proactive alterations leading to a direct user selection from said active suggestion.

19. The proactive user interface of claim 14, wherein said learning module maximizes a percentage of proactive alterations leading to a direct user selection from said active suggestion.

20. The proactive user interface of claim 14, wherein said learning module maximizes a percentage of proactive alterations leading to a direct user selection from said active suggestion.

21. The proactive user interface of claim 15, wherein said learning module maximizes a percentage of proactive alterations leading to a direct user selection from said active suggestion.

21. The proactive user interface of claim 16, wherein said learning module maximizes a percentage of proactive alterations leading to a direct user selection from said active suggestion.

22. The proactive user interface of claim 17, wherein said learning module maximizes a percentage of proactive alterations leading to a direct user selection from said active suggestion.

23. The proactive user interface of claim 18, wherein said maximization is performed through learning reinforcement.

24. The proactive user interface of claim 19, wherein said maximization is performed through learning reinforcement.

25. The proactive user interface of claim 20, wherein said maximization is performed through learning reinforcement.

26. The proactive user interface of claim 21, wherein said maximization is performed through learning reinforcement.

27. The proactive user interface of claim 22, wherein said maximization is performed through learning reinforcement.

28. The proactive user interface of claim 23, wherein said learning reinforcement is performed through an iterative learning process.

29. The proactive user interface of claim 24, wherein said learning reinforcement is performed through an iterative learning process.

30. The proactive user interface of claim 25, wherein said learning reinforcement is performed through an iterative learning process.

31. The proactive user interface of claim 26, wherein said learning reinforcement is performed through an iterative learning process.

32. The proactive user interface of claim 27, wherein said learning reinforcement is performed through an iterative learning process.

33. The proactive user interface of claim 28, wherein each iteration of said learning process is performed after said evolution has been performed.

34. The proactive user interface of claim 29, wherein each iteration of said learning process is performed after said evolution has been performed.

35. The proactive user interface of claim 30, wherein each iteration of said learning process is performed after said evolution has been performed.

36. The proactive user interface of claim 31, wherein each iteration

of said learning process is performed after said evolution has been performed.

37. The proactive user interface of claim 32, wherein each iteration of said learning process is performed after said evolution has been performed.

38. An adaptive system with an evolving agent for a computational device having an operating system, the adaptive system comprising:

(a) a user interface including an avatar for communicating between a user of the adaptive system and said operating system;

(b) at least one software application controlled by said operating system; and

(c) an artificial intelligence (AI) framework for supporting said at least one software application, and communicating with a host platform having the operating system,

wherein said AI framework includes an evolution module for supporting and/or managing an evolution of the avatar.

39. The adaptive system of claim 38, further comprising a knowledge base for storing information selected from the group consisting of a pattern determined according to at least one previous interaction of the user with said user interface, a predetermined pattern, and a combination thereof.

40. The adaptive system of claim 39, wherein said AI framework comprises:

an artificial intelligence/machine learning (AI/ML) module;

an application manager for communicating with said at least one software application;

a storage manager for managing storage and handling of data with regard to the knowledge base;

an action manager for enabling the adaptive system to determine which action to take through an operation of the AI/ML module;

a user interface (UI) manager for managing an appearance and a function of the user interface by directing changes to the user interface; and

a device world mapper for determining a state of the computational device, a state of a virtual world, and relationship between said two states.

41. The adaptive system of claim 40, wherein said AI/ML module

determines a behavior of the adaptive system in response to various stimuli, and enables the adaptive system to learn from a response of the user to different types of actions of the adaptive system.

42. The adaptive system of claim 40, wherein said AI framework further comprises an event handler, and between said at least one software application and said event handler, a plurality of different low level managers support receipt and handling of different events, said low level managers including the action manager, the UI manager, the storage manager, and the application manager.

43. The adaptive system of claim 41, wherein said AI framework further comprises an event handler, and between said at least one software application and said event handler, a plurality of different low level managers support receipt and handling of different events, said low level managers including the action manager, the UI manager, the storage manager, and the application manager.

44. The adaptive system of claim 38, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

45. The adaptive system of claim 39, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

46. The adaptive system of claim 39, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

47. The adaptive system of claim 40, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

48. The adaptive system of claim 41, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

49. The adaptive system of claim 42, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

50. The adaptive system of claim 43, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

51. The adaptive system of claim 38, wherein said at least one software application includes an application for optionally returning the user interface, which has been evolved through an evolutionary algorithm, to a previous state, and storing and managing information of states of the user interface.

52. The adaptive system of claim 38, wherein said at least one software application includes a mutation application for controlling mutations in the adaptive system.

53. The adaptive system of claim 38, wherein said at least one software application includes a gene studio application for enabling the user to perform mutations through at least one manual command.

54. The adaptive system of claim 38, wherein said at least one software application includes a hybrid application for enabling the user to create an avatar having a hybrid appearance with an avatar of another computational device.

55. The adaptive system of claim 51, wherein said application manager is capable of starting, pausing, resuming and stopping each of said at

least one software application.

56. The adaptive system of claim 52, wherein said application manager is capable of starting, pausing, resuming and stopping each of said at least one software application.

57. The adaptive system of claim 53, wherein said application manager is capable of starting, pausing, resuming and stopping each of said at least one software application.

58. The adaptive system of claim 54, wherein said application manager is capable of starting, pausing, resuming and stopping each of said at least one software application.

59. The adaptive system of claim 38, wherein said evolution module manages a gene, a chromosome, and a phenotype of the user interface implemented in the form of an avatar, and actively suggests options for evolving at least one function of the user interface based on the gene, chromosome, and phenotype.

60. The adaptive system of claim 59, wherein said evolution module changes at least one parameter of the gene to create different new phenotypes.

61. The adaptive system of claim 59, wherein said evolution module enables the chromosome of the user interface to hybridize or crossbreed with the same type of chromosome from a different creature to create a new chromosome as a combination of genetic parent chromosomes of the new chromosome.

62. The adaptive system of claim 38, wherein said evolution module performs a natural selection process to decide which genetic children will continue as a next generation.

63. The adaptive system of claim 62, wherein said natural selection process can be decided by the user or can be predefined.

64. An evolution system for an intelligent agent for a computational device having an operating system, said evolution system comprising:

(a) a user interface including an evolving avatar for communicating between a user of the evolution system and said operating system;

(b) at least one software application controlled by said operating system; and

(c) an evolution class for enabling visual or functional properties of said intelligent agent to be changed using algerithmic DNA (DeoxyriboNucleic Acid).

65. The evolution system of claim 64, wherein said evolution class comprises:

a creature DNA class for representing a DNA structure of the DNA;

a creature proxy class for providing an interface to the DNA and to genetic operations for classes of said evolution system;

an evolution MGR class for managing evolutions of said intelligent agent and providing an interface to the creature proxy class and genetic operations of the creature proxy class to said at least one software application; and

an evolution engine class for listening to evolution events generated for indicating that a certain genetic operation should be invoked and performed on the DNA of the intelligent agent.

66. The evolution system of claim 64, wherein the DNA is a vector of available genes.

67. The evolution system of claim 65, wherein the DNA is a vector of available genes.

68. The evolution system of claim 66, wherein the gene is a parameter with a range of possible values, and expression of data of the parameter is a genotype of the intelligent agent.

69. The evolution system of claim 67, wherein the gene is a parameter with a range of possible values, and expression of data of the parameter is a genotype of the intelligent agent.

70. The evolution system of claim 66, wherein said vector includes at least one of a head, a body, a hand, a tail, a leg, head color, body color, hand color, tail color, leg color, head scale, body scale, hand scale, tail scale, leg scale,

dexterity, efficiency, interactive, and base color.

71. The evolution system of claim 67, wherein said vector includes at least one of a head, a body, a hand, a tail, a leg, head color, body color, hand color, tail color, leg color, head scale, body scale, hand scale, tail scale, leg scale, dexterity, efficiency, interactive, and base color.

72. The evolution system of claim 66, wherein said intelligent agent is constructed using a gene included in the DNA structure and a value of said gene.

73. The evolution system of claim 67, wherein said intelligent agent is constructed using a gene included in the DNA structure and a value of said gene.

74. The evolution system of claim 66, wherein, for the same DNA, said intelligent agent expresses a trait corresponding to a combination of first building blocks if a growth time of said intelligent agent is less than a predetermined time interval, and a trait corresponding to a combination of second building blocks if the growth time is equal to or greater than the predetermined time interval so that said intelligent agent automatically grows as time goes on.

75. The evolution system of claim 67, wherein, for the same DNA, said intelligent agent expresses a trait corresponding to a combination of first building blocks if a growth time of said intelligent agent is less than a predetermined time interval, and a trait corresponding to a combination of second building blocks if the growth time is equal to or greater than the predetermined time interval so that said intelligent agent automatically grows as time goes on.

76. The evolution system of claim 74, wherein said predetermined time interval is at least 2 weeks.

77. The evolution system of claim 75, wherein said predetermined time interval is at least 2 weeks.

78. The evolution system of claim 66, wherein said intelligent agent randomly selects a gene from a gene set, and changes a value of the selected gene

within a range thereof to implement mutation.

79. The evolution system of claim 67, wherein said intelligent agent randomly selects a gene from a gene set, and changes a value of the selected gene within a range thereof to implement mutation.

80. The evolution system of claim 78, wherein said mutation is implemented by replacing an existing agent with a mutant after displaying the mutant to the user and obtaining the user's approval of the mutant.

81. The evolution system of claim 79, wherein said mutation is implemented by replacing an existing agent with a mutant after displaying the mutant to the user and obtaining the user's approval of the mutant.

82. The evolution system of claim 66, wherein said intelligent agent performs a hybridization operation on two candidate DNAs by selecting at least one crossover point located on each of the two candidate DNAs and selecting a DNA value from among existing DNA values at each crossover point.

83. The evolution system of claim 67, wherein said intelligent agent performs a hybridization operation on two candidate DNAs by selecting at least one crossover point located on each of the two candidate DNAs and selecting a DNA value from among existing DNA values at each crossover point.

84. The evolution system of claim 82, wherein said hybridization operation is performed between different gender genes.

85. The evolution system of claim 83, wherein said hybridization operation is performed between different gender genes.

86. The evolution system of claim 82, wherein said intelligent agent replaces a current agent with a hybrid agent generated by the hybridization operation after displaying the hybrid agent to the user and obtaining the user's approval of the hybrid agent.

87. The evolution system of claim 83, wherein said intelligent agent replaces a current agent with a hybrid agent generated by the hybridization

operation after displaying the hybrid agent to the user and obtaining the user's approval of the hybrid agent.

88. The evolution system of claim 82, wherein said hybridization operation is performed using an agent DNA included in an SMS message received from an external source through at least one of infrared, BlueTooth, and the Internet.

89. The evolution system of claim 83, wherein said hybridization operation is performed using an agent DNA included in an SMS message received from an external source through at least one of infrared, BlueTooth, and the Internet.

90. The evolution system of claim 66, wherein a combination of letters A, G, C and T expressing the DNA is displayed on a screen.

91. The evolution system of claim 67, wherein a combination of letters A, G, C and T expressing the DNA is displayed on a screen.

92. The evolution system of claim 64, wherein the user can review a history of changes of a creature represented by said avatar.

93. The evolution system of claim 64, wherein, based on user details set by the user, said intelligent agent creates a hybrid with a konghup possibility between the user of the evolution system corresponding to one mobile phone and a user of another mobile phone.

94. The evolution system of claim 65, wherein, based on user details set by the user, said intelligent agent creates a hybrid with a konghup possibility between the user of the evolution system corresponding to one mobile phone and a user of another mobile phone.

95. The evolution system of claim 64, wherein said avatar is controlled to act as one of a virtual pet and a virtual companion.

96. The evolution system of claim 65, wherein said avatar is controlled to act as one of a virtual pet and a virtual companion.

97. The evolution system of claim 64, wherein said avatar is represented as a 3D polygonal object with more than one color.

98. The evolution system of claim 65, wherein said avatar is represented as a 3D polygonal object with more than one color.

99. A mobile telephone, comprising:

(a) a user interface including an evolving avatar for communicating between a user of the telephone and said operating system;

(b) at least one software application controlled by said operating system; and

(c) an evolution class for enabling properties of an intelligent agent to be changed using algorithmic DNA,

wherein said avatar has a plurality of different 3D visual characteristics with attributes of shades, textures and animation.

100. A proactive user interface for a computational device having an operating system, the proactive user interface comprising:

(a) an interface unit for communicating between a user of the proactive user interface and said operating system, said interface unit including an emotional agent for communicating with the user;

(b) at least one software application controlled by the operating system;

(c) an artificial intelligence (AI) framework for supporting said at least one software application, communicating with a host platform having the operating system, detecting at least one pattern of interaction of the user with said interface unit, and actively suggesting, to the user, options for altering at least one function of the user interface according to said detected pattern,

wherein said agent expresses at least one emotion according to a reaction of the user to said suggestion.

101. The proactive user interface of claim 100, further comprising a knowledge base for storing information selected from the group consisting of a pattern determined according to at least one previous interaction of the user with said interface unit, a predetermined pattern, and a combination thereof.

102. The proactive user interface of claim 101, wherein said AI framework comprises:

an artificial intelligence/machine learning (AI/ML) module;

an application manager for communicating with said at least one software application;

a storage manager for managing storage and handling of data with regard to the knowledge base;

an action manager for enabling the proactive user interface to determine which action to take through an operation of the AI/ML module;

a user interface (UI) manager for managing an appearance and a function of the user interface by directing changes to the user interface; and

a device world mapper for determining a state of the computational device, a state of a virtual world, and relationship between said two states.

103. The proactive user interface of claim 102, wherein said AI/ML module determines a behavior of the proactive user interface in response to various stimuli, and enables the proactive user interface to learn from a response of the user to different types of actions of the proactive user interface.

104. The proactive user interface of claim 102, wherein said AI framework further comprises an event handler, and between said at least one software application and said event handler, a plurality of different low level managers support receipt and handling of different events, said low level managers including the action manager, the UI manager, the storage manager, and the application manager.

105. The proactive user interface of claim 103, wherein said AI framework further comprises an event handler, and between said at least one software application and said event handler, a plurality of different low level managers support receipt and handling of different events, said low level managers including the action manager, the UI manager, the storage manager, and the application manager.

106. The proactive user interface of claim 103, wherein the application manager is capable of starting, pausing, resuming and stopping each of said at least one software application.

107. The proactive user interface of claim 100, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

108. The proactive user interface of claim 101, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

109. The proactive user interface of claim 102, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

110. The proactive user interface of claim 103, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

111. The proactive user interface of claim 104, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

112. The proactive user interface of claim 105, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

113. The proactive user interface of claim 106, wherein said computational device is selected from the group consisting of a regular computer, an ATM, mobile information devices including a cellular telephone, a PDA, and a consumer appliance having an operating system.

114. The proactive user interface of claim 107, wherein said agent is created through a 3D graphic model.

115. The proactive user interface of claim 108, wherein said agent is created through a 3D graphic model.

116. The proactive user interface of claim 109, wherein said agent is created through a 3D graphic model.

117. The proactive user interface of claim 110, wherein said agent is created through a 3D graphic model.

118. The proactive user interface of claim 111, wherein said agent is created through a 3D graphic model.

119. The proactive user interface of claim 112, wherein said agent is created through a 3D graphic model.

120. The proactive user interface of claim 113, wherein said agent is created through a 3D graphic model.

121. The proactive user interface of claim 114, wherein said proactive user interface controls the avatar to be displayed independently of visual display aspects of the user interface.

122. The proactive user interface of claim 115, wherein said proactive user interface controls the avatar to be displayed independently of visual display aspects of the user interface.

123. The proactive user interface of claim 116, wherein said proactive user interface controls the avatar to be displayed independently of visual display aspects of the user interface.

124. The proactive user interface of claim 117, wherein said proactive user interface controls the avatar to be displayed independently of visual display aspects of the user interface.

125. The proactive user interface of claim 118, wherein said proactive user interface controls the avatar to be displayed independently of visual display

aspects of the user interface.

126. The proactive user interface of claim 119, wherein said proactive user interface controls the avatar to be displayed independently of visual display aspects of the user interface.

127. The proactive user interface of claim 120, wherein said proactive user interface controls the avatar to be displayed independently of visual display aspects of the user interface.

128. The proactive user interface of claim 100, wherein said intelligent agent communicates with an object that is found in a virtual environment.

129. The proactive user interface of claim 128, wherein said object in the virtual environment includes at least one of a ball, a good animal, food, a bad animal, a house, and toys.

130. The proactive user interface of claim 129, wherein said object in the virtual environment includes a graded input to a state of said agent.

131. The proactive user interface of claim 130, wherein said object in the virtual environment becomes an incentive or disincentive for said agent to continue a behavior for which feedback has been provided.

132. The proactive user interface of claim 130, wherein said graded input includes reward points provided to the agent, and the agent responds to the reward points.

133. The proactive user interface of claim 100, wherein said suggestion is determined by a rule based strategy.

134. The proactive user interface of claim 133, wherein said rule based strategy comprises:

querying a knowledge base when an event occurs, and receiving a response therefrom;

determining whether the event is valid or not;

generating an action corresponding to the event and determining priority for the action; and

performing a highest priority action from among actions corresponding to the event.

135. The proactive user interface of claim 134, wherein the highest priority action is an action that maximizes an aggregated total reward from a virtual environment or a graded input in the form of encouraging or discouraging feedback.

136. The proactive user interface of claim 133, wherein the proactive user interface creates an emotion of the agent based on a change of at least one of a likelihood of success (LOS) and a likelihood of failure (LOF) of an abstract goal of the agent.

137. The proactive user interface of claim 134, wherein the proactive user interface creates an emotion of the agent based on a change of at least one of a likelihood of success (LOS) and a likelihood of failure (LOF) of an abstract goal of the agent.

138. The proactive user interface of claim 135, wherein the proactive user interface creates an emotion of the agent based on a change of at least one of a likelihood of success (LOS) and a likelihood of failure (LOF) of an abstract goal of the agent.

139. The proactive user interface of claim 136, wherein the created emotion is accumulated, and the agent expresses an emotional state each time a dominant behavior pattern changes.

140. The proactive user interface of claim 137, wherein the created emotion is accumulated, and the agent expresses an emotional state each time a dominant behavior pattern changes.

141. The proactive user interface of claim 138, wherein the created emotion is accumulated, and the agent expresses an emotional state each time a dominant behavior pattern changes.

142. The proactive user interface of claim 139, wherein the emotional state is expressed using at least one of textual communication, three dimensional animation, facial expressions, two dimensional animated effects and sounds.

143. The proactive user interface of claim 140, wherein the emotional state is expressed using at least one of textual communication, three dimensional animation, facial expressions, two dimensional animated effects and sounds.

144. The proactive user interface of claim 141, wherein the emotional state is expressed using at least one of textual communication, three dimensional animation, facial expressions, two dimensional animated effects and sounds.

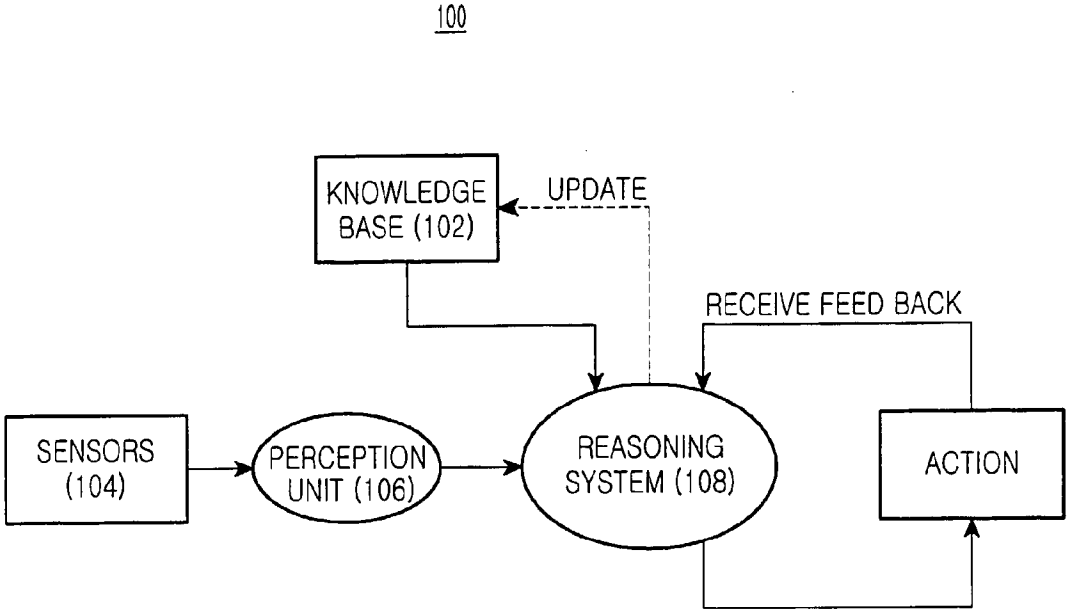

100
KNOWLEDGE BASE (102)
UPDATE
RECEIVE FEED BACK
SENSORS (104)
PERCEPTION UNIT (106)
REASONING SYSTEM (108)
ACTION

FIG.1

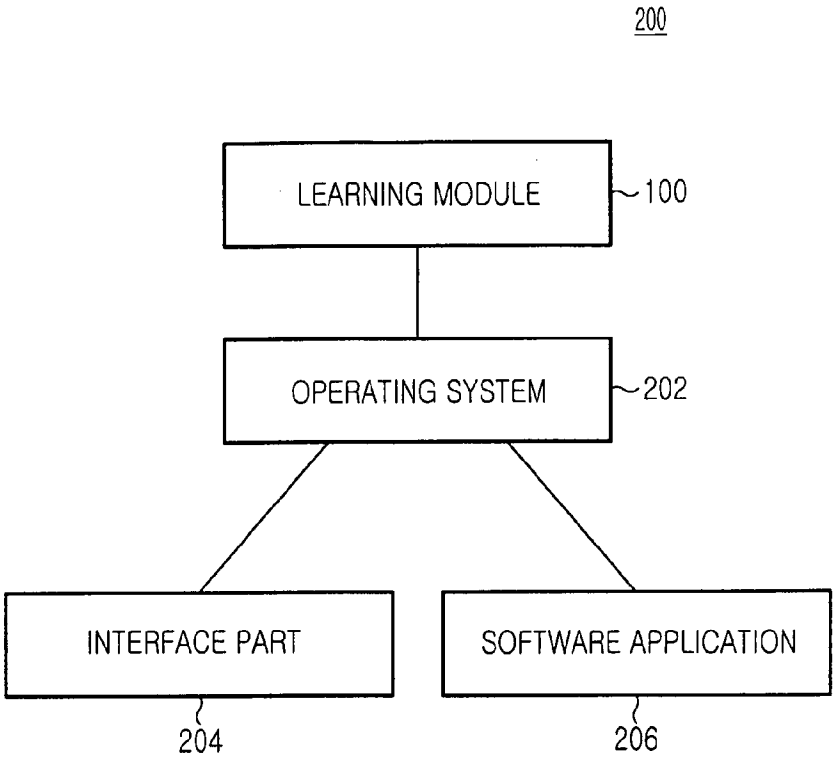

200
LEARNING MODULE
100
OPERATING SYSTEM
202
INTERFACE PART
204
SOFTWARE APPLICATION
206

FIG.2

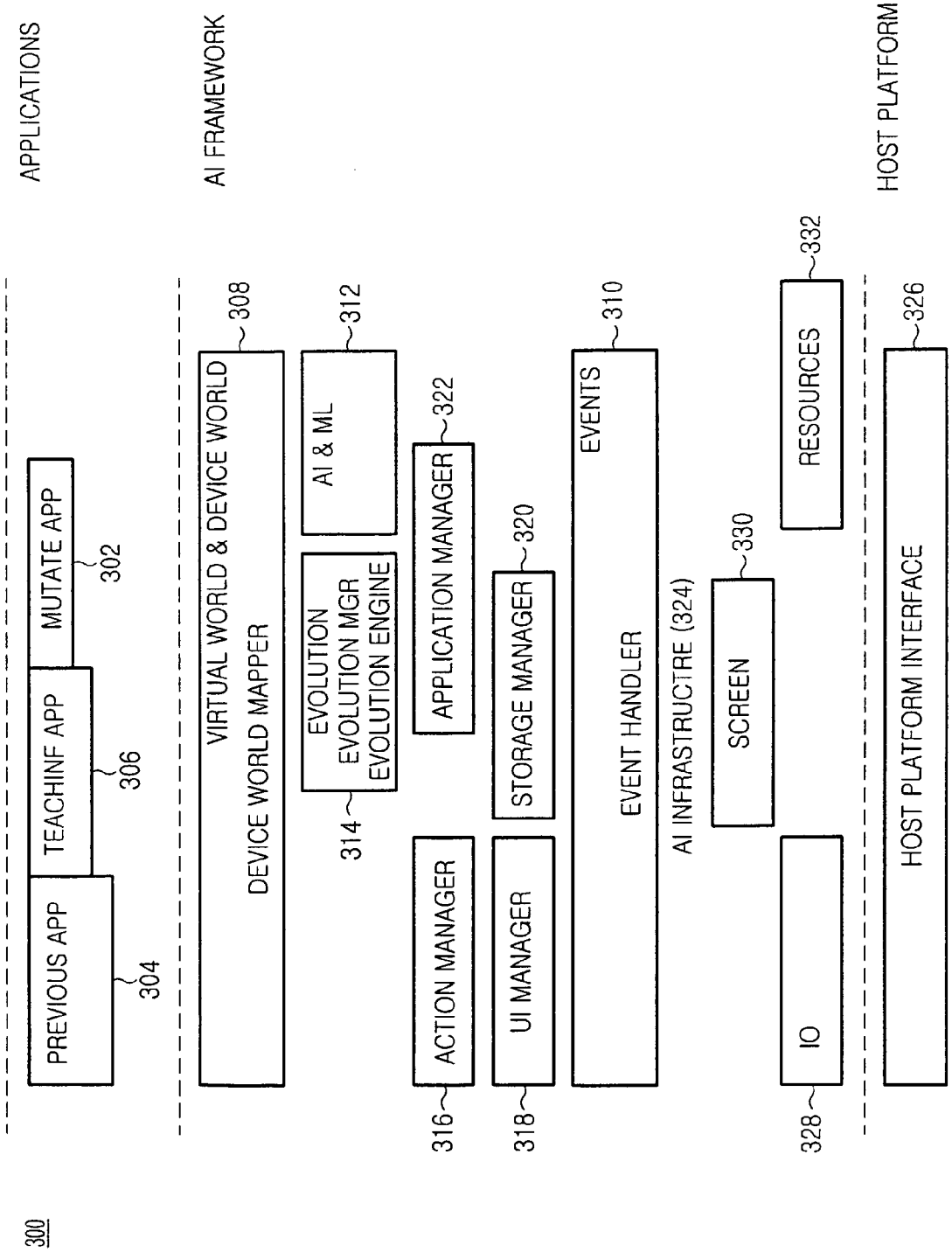

300
APPLICATIONS
AI FRAMEWORK
HOST PLATFORM
PREVIOUS APP
304
TEACHINF APP
306
MUTATE APP
302
VIRTUAL WORLD & DEVICE WORLD
DEVICE WORLD MAPPER
308
EVOLUTION
EVOLUTION MGR
EVOLUTION ENGINE
314
AI & ML
312
APPLICATION MANAGER
322
ACTION MANAGER
316
STORAGE MANAGER
320
UI MANAGER
318
EVENTS
EVENT HANDLER
310
AI INFRASTRUCTRE (324)
SCREEN
330
RESOURCES
332
IO
328
HOST PLATFORM INTERFACE
326

FIG.3

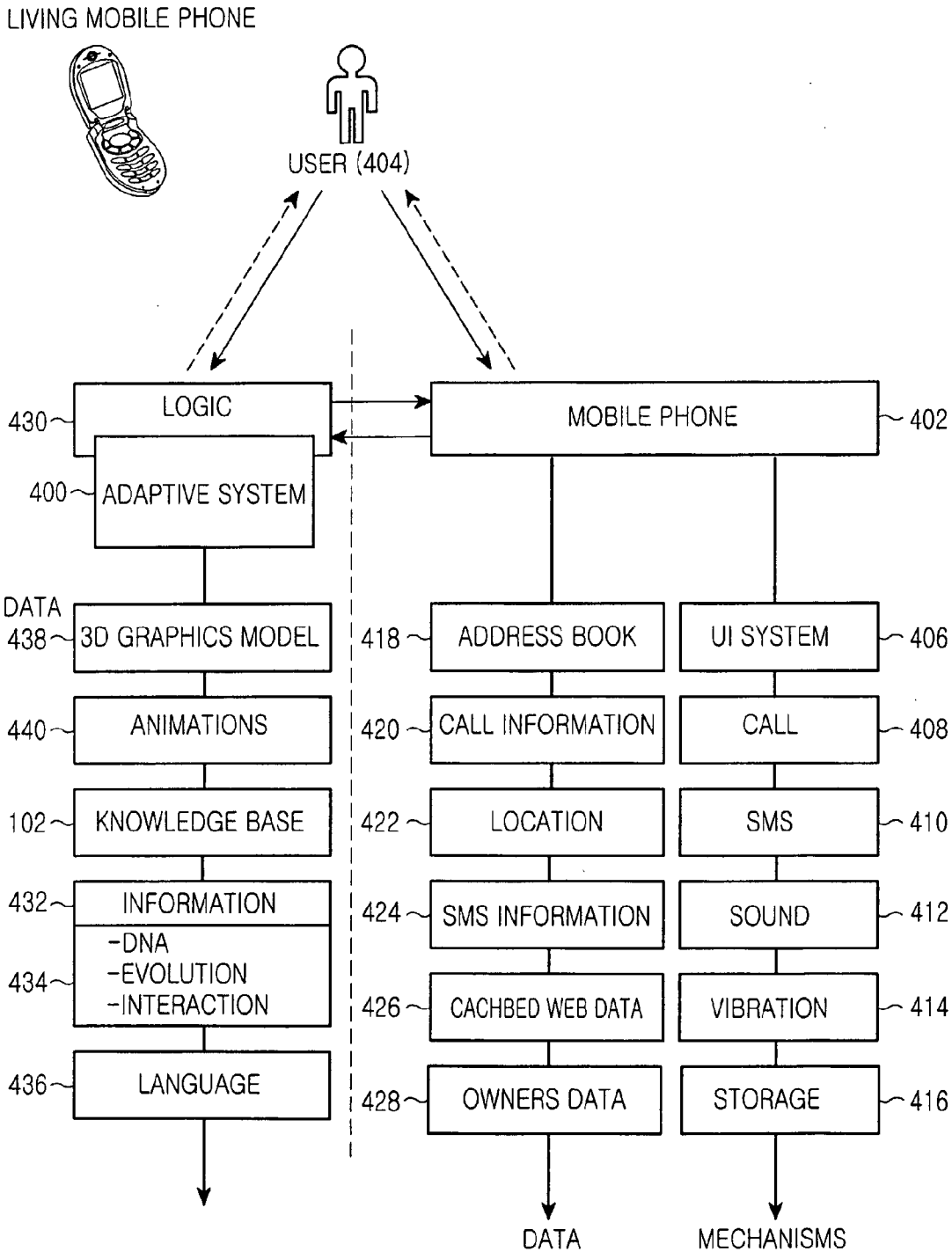

LIVING MOBILE PHONE
USER (404)
430
LOGIC
400
ADAPTIVE SYSTEM
MOBILE PHONE
402
DATA
438
3D GRAPHICS MODEL
440
ANIMATIONS
102
KNOWLEDGE BASE
432
INFORMATION
434
-DNA
-EVOLUTION
-INTERACTION
436
LANGUAGE
418
ADDRESS BOOK
420
CALL INFORMATION
422
LOCATION
424
SMS INFORMATION
426
CACHBED WEB DATA
428
OWNERS DATA
UI SYSTEM
406
CALL
408
SMS
410
SOUND
412
VIBRATION
414
STORAGE
416
DATA
MECHANISMS

FIG.4

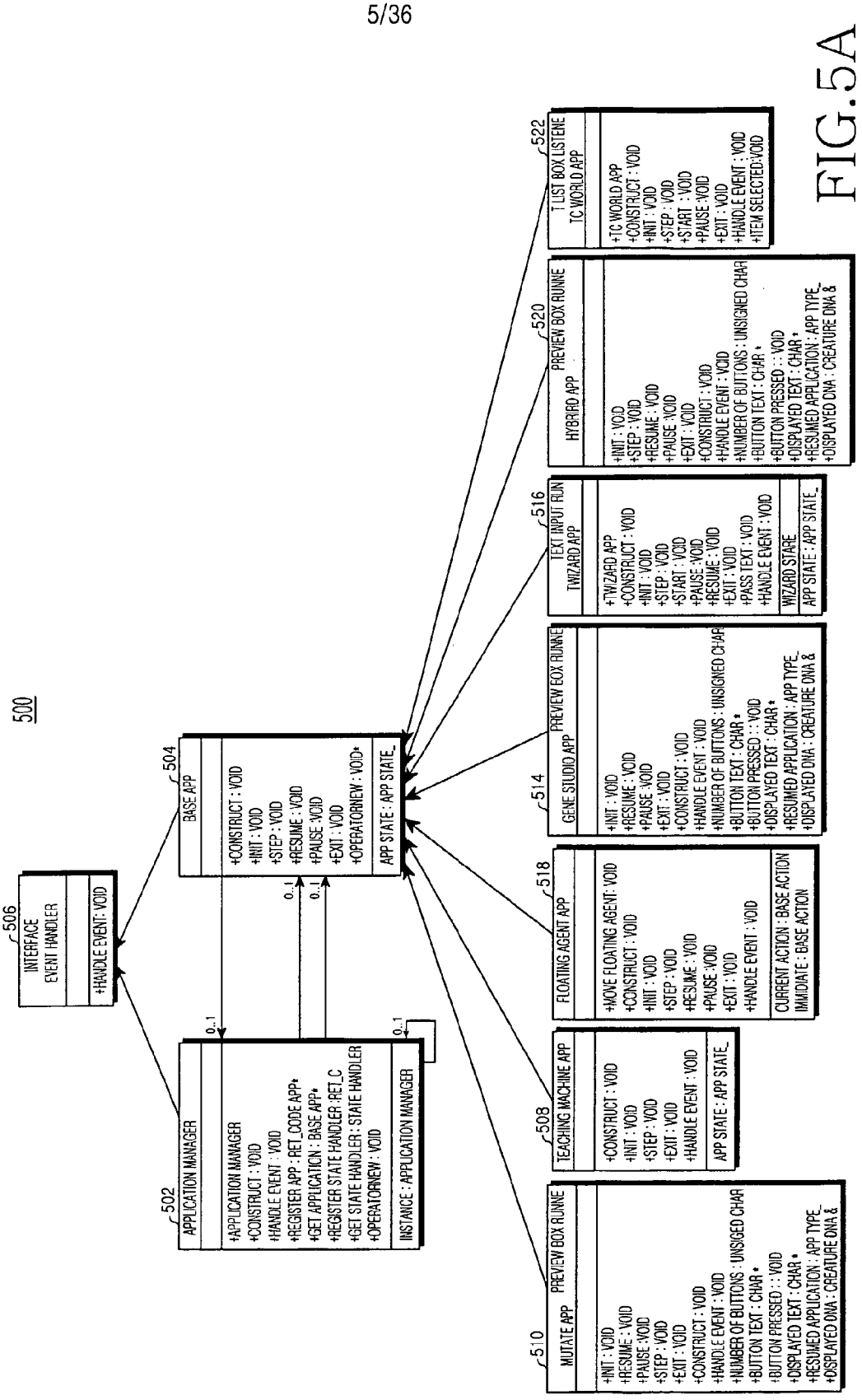

500
506
INTERFACE
EVENT HANDLER
+HANDLE EVENT: VOID
502
APPLICATION MANAGER
+APPLICATION MANAGER
+CONSTRUCT : VOID
+HANDLE EVENT : VOID
+REGISTER APP : RET_CODE APP*
+GET APPLICATION : BASE APP*
+REGISTER STATE HANDLER : RET_C
+GET STATE HANDLER : STATE HANDLER
+OPERATORNEW : VOID
INSTANCE : APPLICATION MANAGER
504
BASE APP
+CONSTRUCT : VOID
+INIT : VOID
+STEP : VOID
+RESUME : VOID
+PAUSE :VOID
+EXIT : VOID
+OPERATORNEW : VOID*
APP STATE : APP STATE_
0..1
0..1
0..1
508
TEACHING MACHINE APP
+CONSTRUCT : VOID
+INIT : VOID
+STEP : VOID
+EXIT : VOID
+HANDLE EVENT : VOID
APP STATE : APP STATE_
510
MUTATE APP PREVIEW BOX RUNNE
+INIT : VOID
+RESUME : VOID
+PAUSE :VOID
+STEP : VOID
+EXIT : VOID
+CONSTRUCT : VOID
+HANDLE EVENT : VOID
+NUMBER OF BUTTONS : UNSIGED CHAR
+BUTTON TEXT : CHAR*
+BUTTON PRESSED : : VOID
+DISPLAYED TEXT : CHAR*
+RESUMED APPLICATION : APP TYPE_
+DISPLAYED DNA : CREATURE DNA &
518
FLOATING AGENT APP
+MOVE FLOATING AGENT: VOID
+CONSTRUCT : VOID
+INIT : VOID
+STEP : VOID
+RESUME : VOID
+PAUSE :VOID
+EXIT : VOID
+HANDLE EVENT : VOID
CURRENT ACTION : BASE ACTION
IMMIDIATE : BASE ACTION
514
GENE STUDIO APP PREVIEW BOX RUNNE
+INIT : VOID
+RESUME : VOID
+PAUSE :VOID
+EXIT : VOID
+CONSTRUCT : VOID
+HANDLE EVENT : VOID
+NUMBER OF BUTTONS : UNSIGNED CHAR
+BUTTON TEXT : CHAR*
+BUTTON PRESSED : : VOID
+DISPLAYED TEXT : CHAR*
+RESUMED APPLICATION : APP TYPE_
+DISPLAYED DNA : CREATURE DNA &
516
TEXT INPUT RUN
TWIZARD APP
+TWIZARD APP
+CONSTRUCT : VOID
+INIT : VOID
+STEP : VOID
+START : VOID
+PAUSE :VOID
+RESUME : VOID
+EXIT : VOID
+PASS TEXT : VOID
+HANDLE EVENT : VOID
WIZARD STARE
APP STATE : APP STATE_
520
HYBRID APP PREVIEW BOX RUNNE
+INIT : VOID
+STEP : VOID
+RESUME : VOID
+PAUSE :VOID
+EXIT : VOID
+CONSTRUCT : VOID
+HANDLE EVENT : VOID
+NUMBER OF BUTTONS : UNSIGNED CHAR
+BUTTON TEXT : CHAR*
+BUTTON PRESSED : : VOID
+DISPLAYED TEXT : CHAR*
+RESUMED APPLICATION : APP TYPE_
+DISPLAYED DNA : CREATURE DNA &
522
T LIST BOX LISTENE
TC WORLD APP
+TC WORLD APP
+CONSTRUCT : VOID
+INIT : VOID
+STEP : VOID
+START : VOID
+PAUSE :VOID
+EXIT : VOID
+HANDLE EVENT : VOID
+ITEM SELECTED:VOID

FIG.5A

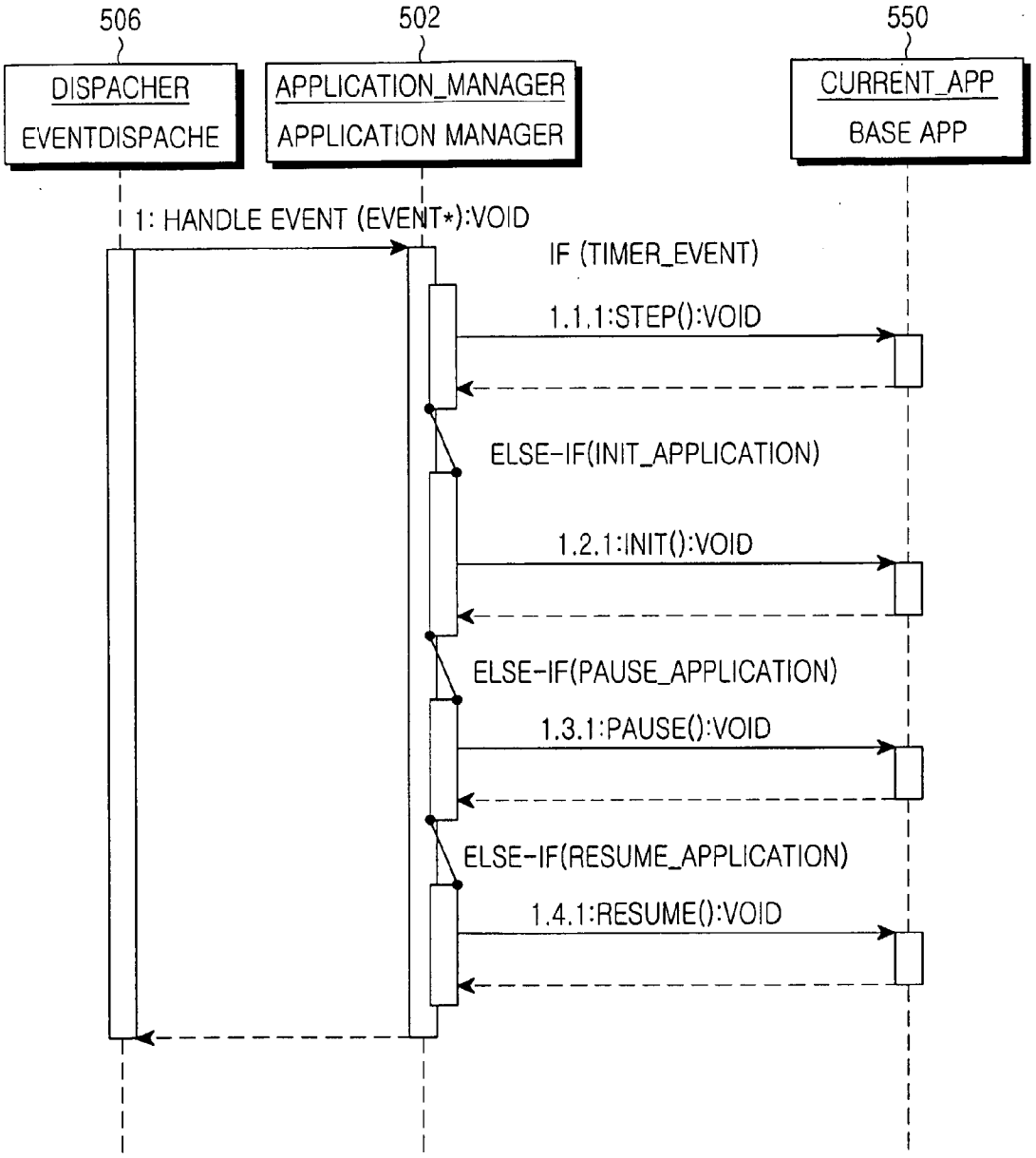

506
502
550
DISPACHER
EVENTDISPACHE
APPLICATION_MANAGER
APPLICATION MANAGER
CURRENT_APP
BASE APP
1: HANDLE EVENT (EVENT*):VOID
IF (TIMER_EVENT)
1.1.1:STEP():VOID
ELSE-IF(INIT_APPLICATION)
1.2.1:INIT():VOID
ELSE-IF(PAUSE_APPLICATION)
1.3.1:PAUSE():VOID
ELSE-IF(RESUME_APPLICATION)
1.4.1:RESUME():VOID

FIG.5B

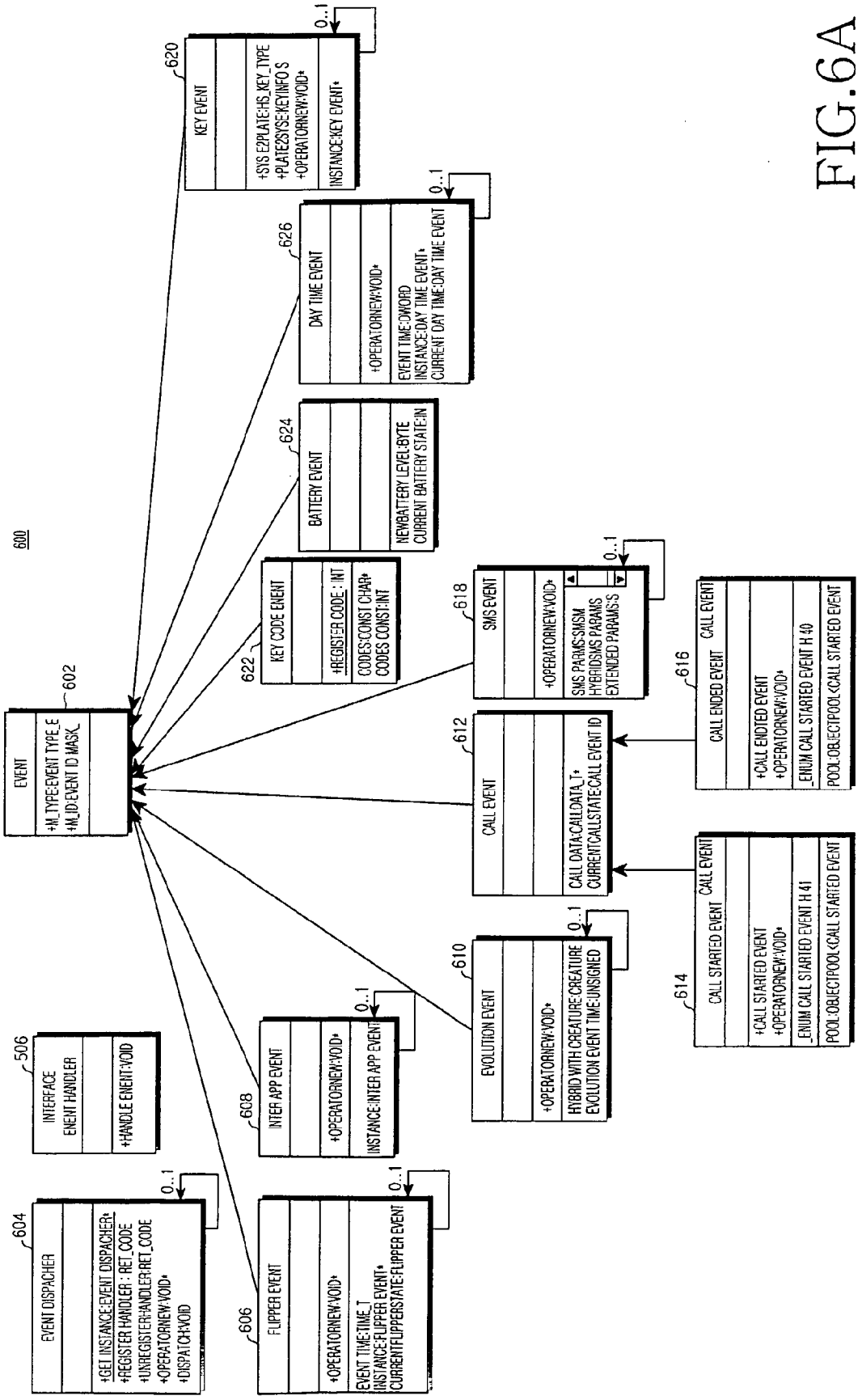

600
EVENT
+M_TYPE:EVENT TYPE_E
+M_ID:EVENT ID MASK
602
KEY EVENT
+SYS E2PLATE:HS_KEY_TYPE
+PLATE2SYS:KEYINFO S
+OPERATORNEW:VOID*
INSTANCE:KEY EVENT*
620
DAY TIME EVENT
+OPERATORNEW:VOID*
EVENT TIME:DWORD
INSTANCE:DAY TIME EVENT*
CURRENT DAY TIME:DAY TIME EVENT
626
BATTERY EVENT
NEWBATTERY LEVEL:BYTE
CURRENT BATTERY STATE:IN
624
KEY CODE EVENT
+REGISTER CODE : INT
CODES:CONST CHAR*
CODES CONST:INT
622
SMS EVENT
+OPERATORNEW:VOID*
SMS PARMS:SMSM
HYBRIDSMS PARAMS
EXTENDED PARAMS:S
618
CALL EVENT
CALL DATA:CALLDATA_T*
CURRENTCALLSTATE:CALL EVENT ID
612
CALL ENDED EVENT
+CALL ENDTED EVENT
+OPERATORNEW:VOID*
_ENUM CALL STARTED EVENT H 40
POOL:OBJECTPOOL<CALL STARTED EVENT
616
CALL STARTED EVENT
+CALL STARTED EVENT
+OPERATORNEW:VOID*
_ENUM CALL STARTED EVENT H 41
POOL:OBJECTPOOL<CALL STARTED EVENT
614
EVOLUTION EVENT
+OPERATORNEW:VOID*
HYBRID WITH CREATURE:CREATURE
EVOLUTION EVENT TIME:UNSIGNED
610
INTERFACE
EVENT HANDLER
+HANDLE EVENT:VOID
506
INTER APP EVENT
+OPERATORNEW:VOID*
INSTANCE:INTER APP EVENT*
608
EVENT DISPACHER
+GET INSTANCE:EVENT DISPACHER*
+REGISTER HANDLER : RET_CODE
+UNREGISTER:HANDLER:RET_CODE
+OPERATORNEW:VOID*
+DISPATCH:VOID
604
FLIPPER EVENT
+OPERATORNEW:VOID*
EVENT TIME:TIME_T
INSTANCE:FLIPPER EVENT*
CURRENTFLIPPERSTATE:FLIPPER EVENT
606
0..1

FIG.6A

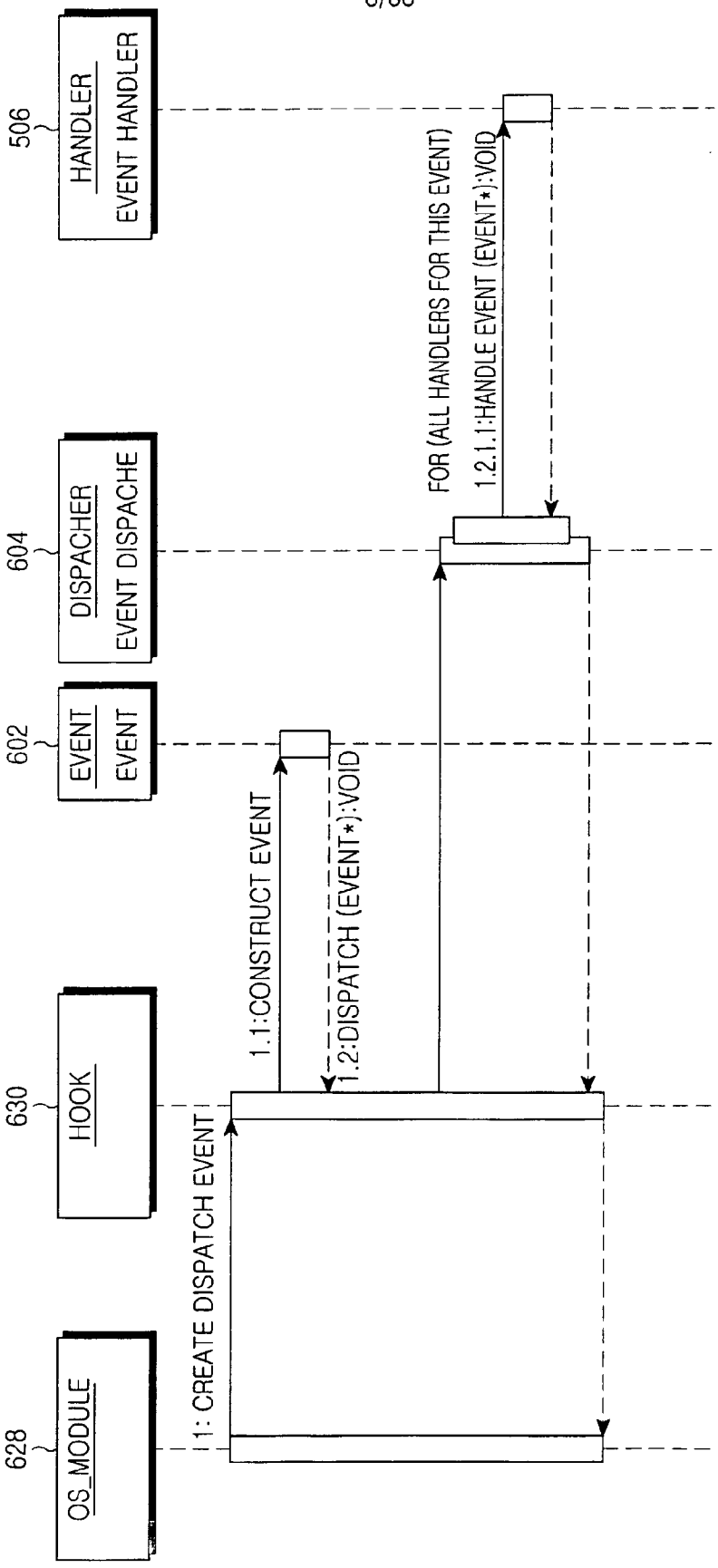

628
OS_MODULE
630
HOOK
602
EVENT
EVENT
604
DISPACHER
EVENT DISPACHE
506
HANDLER
EVENT HANDLER
1: CREATE DISPATCH EVENT
1.1:CONSTRUCT EVENT
1.2:DISPATCH (EVENT*):VOID
FOR (ALL HANDLERS FOR THIS EVENT)
1.2.1.1:HANDLE EVENT (EVENT*):VOID

FIG.6B

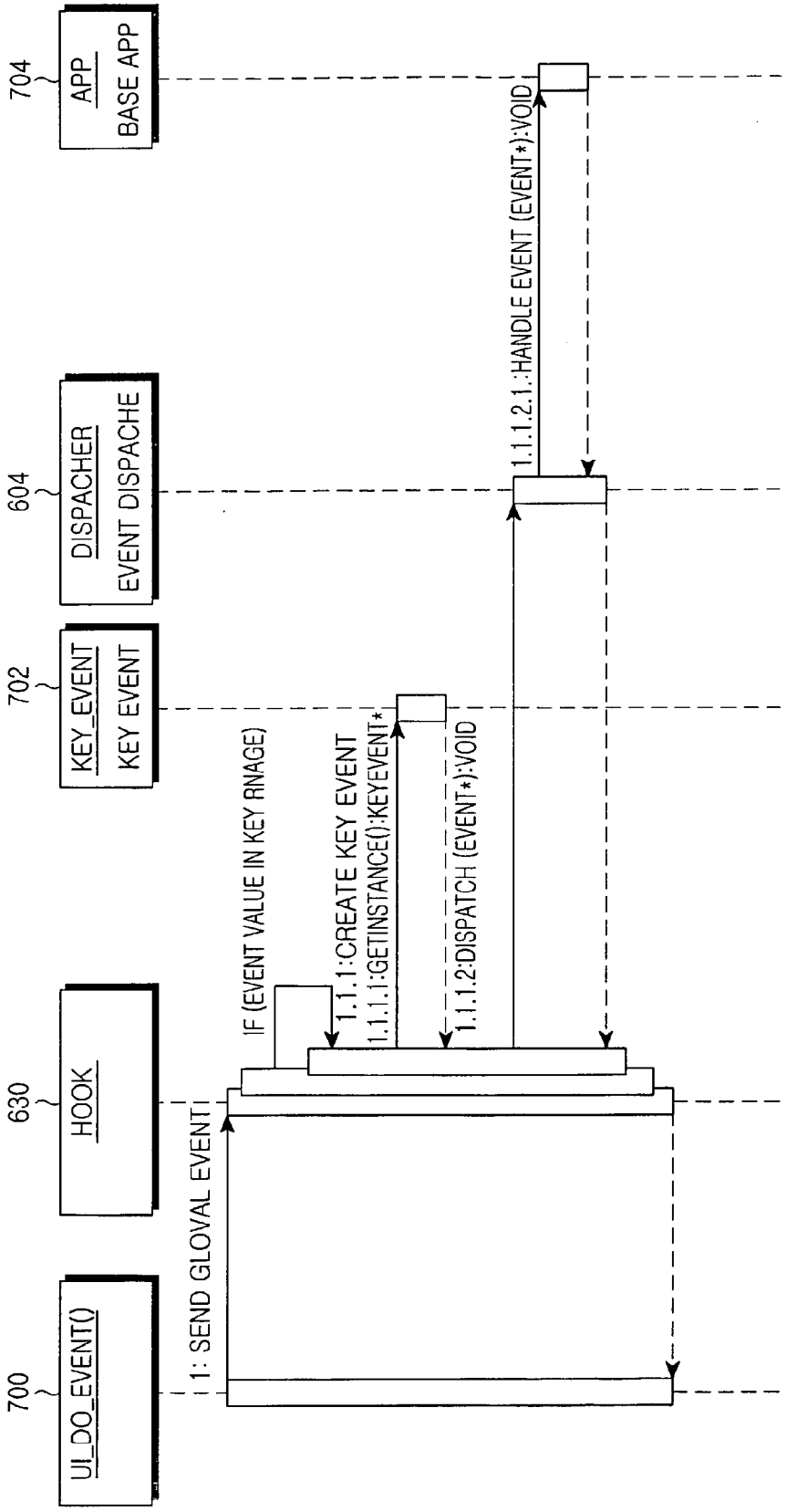

704
APP
BASE APP
604
DISPACHER
EVENT DISPACHE
702
KEY_EVENT
KEY EVENT
630
HOOK
700
UI_DO_EVENT()
1: SEND GLOVAL EVENT
IF (EVENT VALUE IN KEY RNAGE)
1.1.1:CREATE KEY EVENT
1.1.1.1:GETINSTANCE():KEYEVENT*
1.1.1.2:DISPATCH (EVENT*):VOID
1.1.1.2.1.:HANDLE EVENT (EVENT*):VOID

FIG.7A

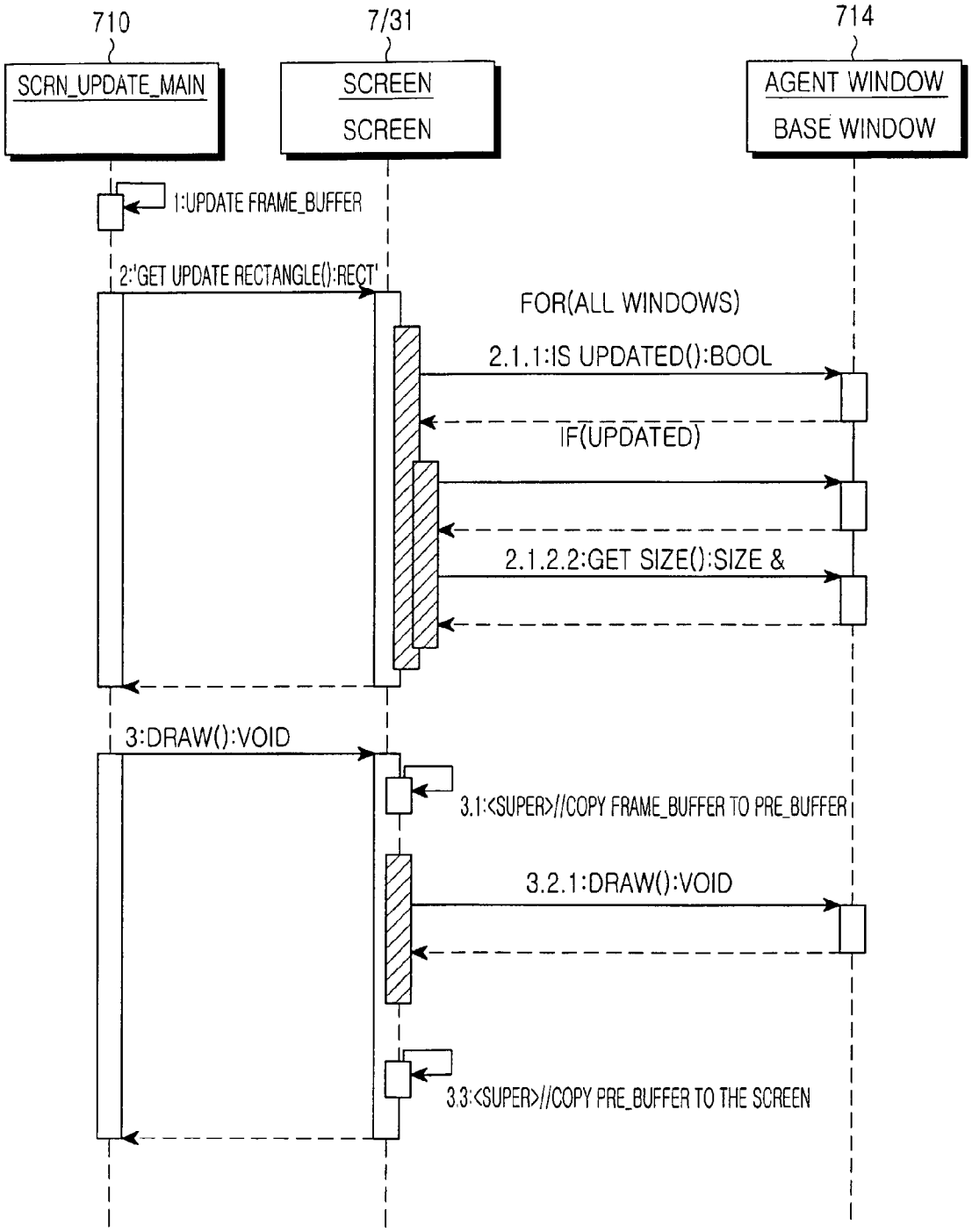

710
7/31
714
SCRN_UPDATE_MAIN
SCREEN
SCREEN
AGENT WINDOW
BASE WINDOW
1:UPDATE FRAME_BUFFER
2:'GET UPDATE RECTANGLE():RECT'
FOR(ALL WINDOWS)
2.1.1:IS UPDATED():BOOL
IF(UPDATED)
2.1.2.2:GET SIZE():SIZE &
3:DRAW():VOID
3.1:<SUPER>//COPY FRAME_BUFFER TO PRE_BUFFER
3.2.1:DRAW():VOID
3.3:<SUPER>//COPY PRE_BUFFER TO THE SCREEN

FIG.7B

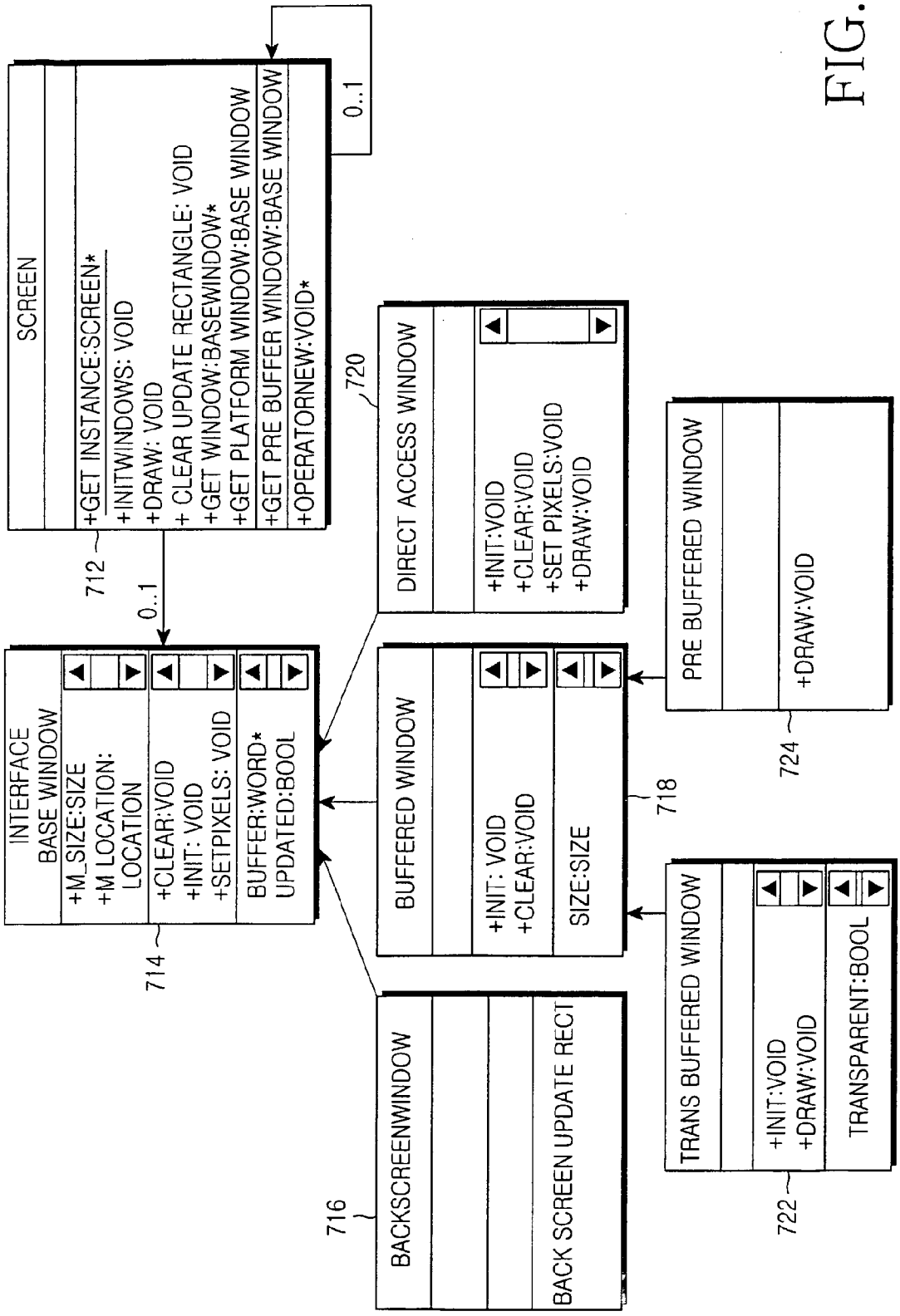

SCREEN
+GET INSTANCE:SCREEN*
+INITWINDOWS: VOID
+DRAW: VOID
+ CLEAR UPDATE RECTANGLE: VOID
+GET WINDOW:BASEWINDOW*
+GET PLATFORM WINDOW:BASE WINDOW
+GET PRE BUFFER WINDOW:BASE WINDOW
+OPERATORNEW:VOID*
0..1
712
0..1
INTERFACE BASE WINDOW
+M_SIZE:SIZE
+M LOCATION: LOCATION
+CLEAR:VOID
+INIT: VOID
+SETPIXELS: VOID
BUFFER:WORD*
UPDATED:BOOL
714
DIRECT ACCESS WINDOW
+INIT:VOID
+CLEAR:VOID
+SET PIXELS:VOID
+DRAW:VOID
720
BUFFERED WINDOW
+INIT: VOID
+CLEAR:VOID
SIZE:SIZE
718
PRE BUFFERED WINDOW
+DRAW:VOID
724
BACKSCREENWINDOW
BACK SCREEN UPDATE RECT
716
TRANS BUFFERED WINDOW
+INIT:VOID
+DRAW:VOID
TRANSPARENT:BOOL
722

FIG.7C

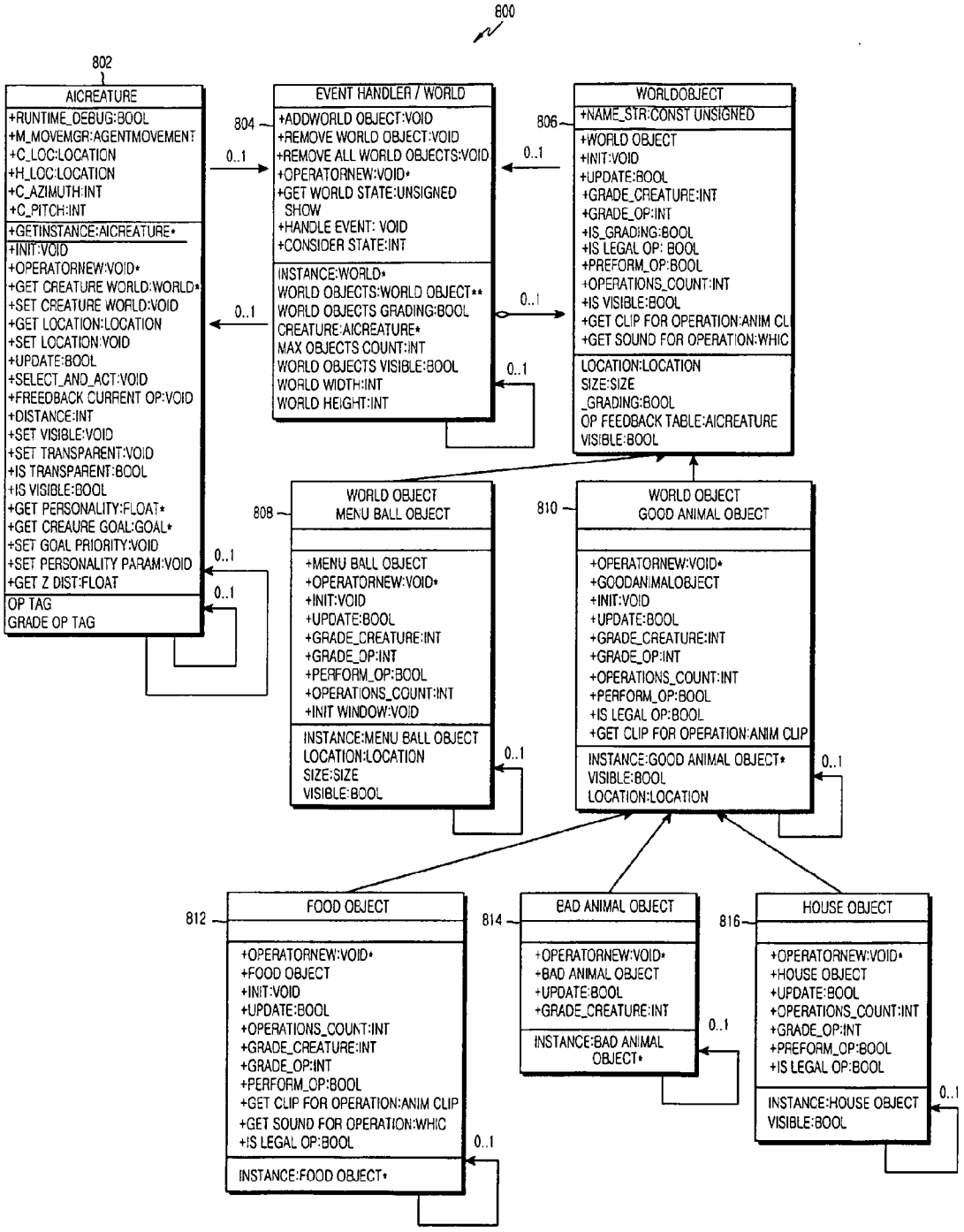

800
802
AICREATURE
+RUNTIME_DEBUG:BOOL
+M_MOVEMGR:AGENTMOVEMENT
+C_LOC:LOCATION
+H_LOC:LOCATION
+C_AZIMUTH:INT
+C_PITCH:INT
+GETINSTANCE:AICREATURE*
+INIT:VOID
+OPERATORNEW:VOID*
+GET CREATURE WORLD:WORLD*
+SET CREATURE WORLD:VOID
+GET LOCATION:LOCATION
+SET LOCATION:VOID
+UPDATE:BOOL
+SELECT_AND_ACT:VOID
+FREEDBACK CURRENT OP:VOID
+DISTANCE:INT
+SET VISIBLE:VOID
+SET TRANSPARENT:VOID
+IS TRANSPARENT:BOOL
+IS VISIBLE:BOOL
+GET PERSONALITY:FLOAT*
+GET CREAURE GOAL:GOAL*
+SET GOAL PRIORITY:VOID
+SET PERSONALITY PARAM:VOID
+GET Z DIST:FLOAT
OP TAG
GRADE OP TAG
804
EVENT HANDLER / WORLD
+ADDWORLD OBJECT:VOID
+REMOVE WORLD OBJECT:VOID
+REMOVE ALL WORLD OBJECTS:VOID
+OPERATORNEW:VOID*
+GET WORLD STATE:UNSIGNED
SHOW
+HANDLE EVENT: VOID
+CONSIDER STATE:INT
INSTANCE:WORLD*
WORLD OBJECTS:WORLD OBJECT**
WORLD OBJECTS GRADING:BOOL
CREATURE:AICREATURE*
MAX OBJECTS COUNT:INT
WORLD OBJECTS VISIBLE:BOOL
WORLD WIDTH:INT
WORLD HEIGHT:INT
806
WORLDOBJECT
+NAME_STR:CONST UNSIGNED
+WORLD OBJECT
+INIT:VOID
+UPDATE:BOOL
+GRADE_CREATURE:INT
+GRADE_OP:INT
+IS_GRADING:BOOL
+IS LEGAL OP: BOOL
+PREFORM_OP:BOOL
+OPERATIONS_COUNT:INT
+IS VISIBLE:BOOL
+GET CLIP FOR OPERATION:ANIM CL
+GET SOUND FOR OPERATION:WHIC
LOCATION:LOCATION
SIZE:SIZE
_GRADING:BOOL
OP FEEDBACK TABLE:AICREATURE
VISIBLE:BOOL
808
WORLD OBJECT
MENU BALL OBJECT
+MENU BALL OBJECT
+OPERATORNEW:VOID*
+INIT:VOID
+UPDATE:BOOL
+GRADE_CREATURE:INT
+GRADE_OP:INT
+PERFORM_OP:BOOL
+OPERATIONS_COUNT:INT
+INIT WINDOW:VOID
INSTANCE:MENU BALL OBJECT
LOCATION:LOCATION
SIZE:SIZE
VISIBLE:BOOL
810
WORLD OBJECT
GOOD ANIMAL OBJECT
+OPERATORNEW:VOID*
+GOODANIMALOBJECT
+INIT:VOID
+UPDATE:BOOL
+GRADE_CREATURE:INT
+GRADE_OP:INT
+OPERATIONS_COUNT:INT
+PERFORM_OP:BOOL
+IS LEGAL OP:BOOL
+GET CLIP FOR OPERATION:ANIM CLIP
INSTANCE:GOOD ANIMAL OBJECT*
VISIBLE:BOOL
LOCATION:LOCATION
812
FOOD OBJECT
+OPERATORNEW:VOID*
+FOOD OBJECT
+INIT:VOID
+UPDATE:BOOL
+OPERATIONS_COUNT:INT
+GRADE_CREATURE:INT
+GRADE_OP:INT
+PERFORM_OP:BOOL
+GET CLIP FOR OPERATION:ANIM CLIP
+GET SOUND FOR OPERATION:WHIC
+IS LEGAL OP:BOOL
INSTANCE:FOOD OBJECT*
814
BAD ANIMAL OBJECT
+OPERATORNEW:VOID*
+BAD ANIMAL OBJECT
+UPDATE:BOOL
+GRADE_CREATURE:INT
INSTANCE:BAD ANIMAL OBJECT*
816
HOUSE OBJECT
+OPERATORNEW:VOID*
+HOUSE OBJECT
+UPDATE:BOOL
+OPERATIONS_COUNT:INT
+GRADE_OP:INT
+PREFORM_OP:BOOL
+IS LEGAL OP:BOOL
INSTANCE:HOUSE OBJECT
VISIBLE:BOOL
0..1

FIG.8A

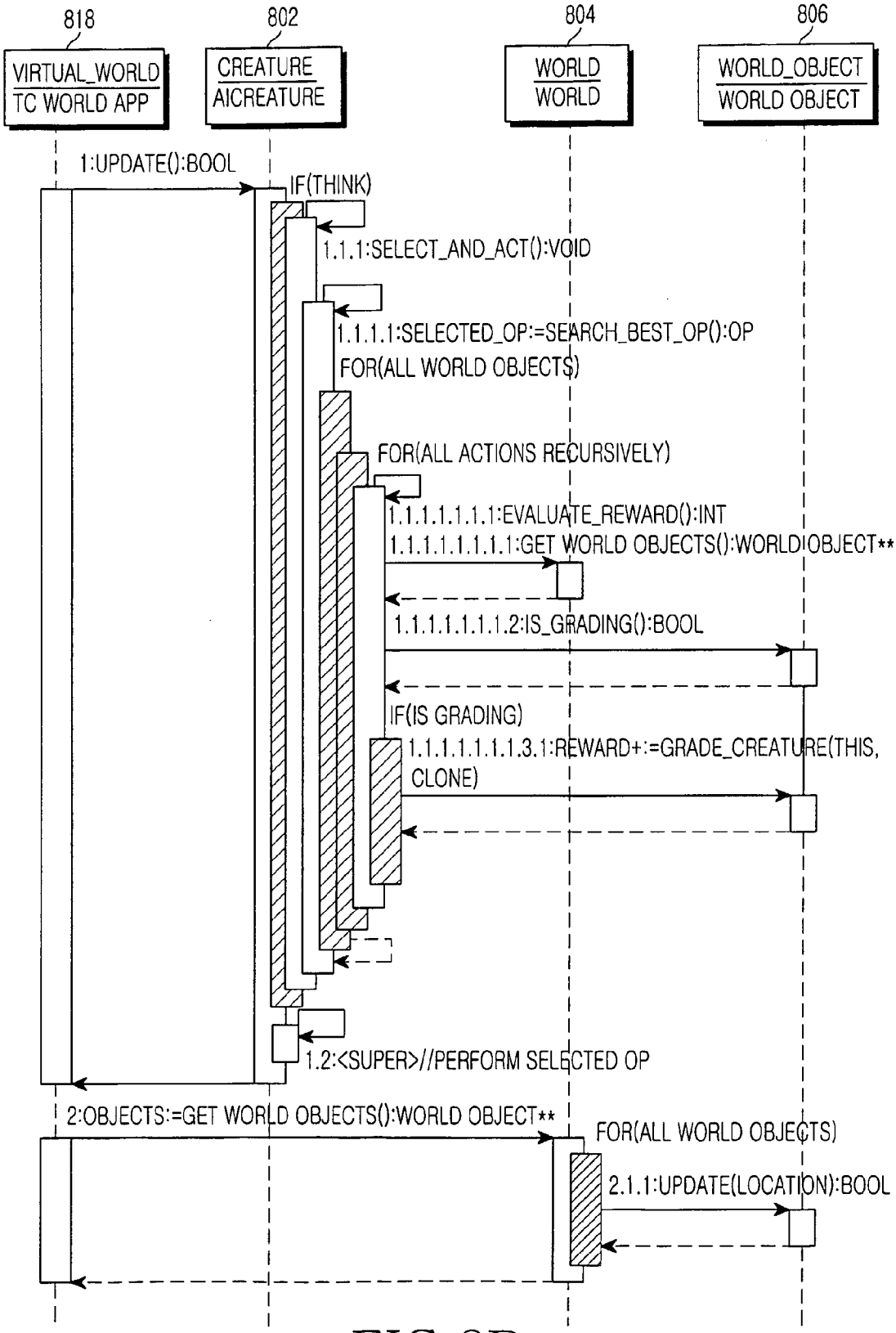

818
VIRTUAL_WORLD
TC WORLD APP
802
CREATURE
AICREATURE
804
WORLD
WORLD
806
WORLD_OBJECT
WORLD OBJECT
1:UPDATE():BOOL
IF(THINK)
1.1.1:SELECT_AND_ACT():VOID
1.1.1.1:SELECTED_OP:=SEARCH_BEST_OP():OP
FOR(ALL WORLD OBJECTS)
FOR(ALL ACTIONS RECURSIVELY)
1.1.1.1.1.1.1:EVALUATE_REWARD():INT
1.1.1.1.1.1.1.1:GET WORLD OBJECTS():WORLD OBJECT**
1.1.1.1.1.1.1.2:IS_GRADING():BOOL
IF(IS GRADING)
1.1.1.1.1.1.1.3.1:REWARD+:=GRADE_CREATURE(THIS,
CLONE)
1.2:<SUPER>//PERFORM SELECTED OP
2:OBJECTS:=GET WORLD OBJECTS():WORLD OBJECT**
FOR(ALL WORLD OBJECTS)
2.1.1:UPDATE(LOCATION):BOOL

FIG.8B

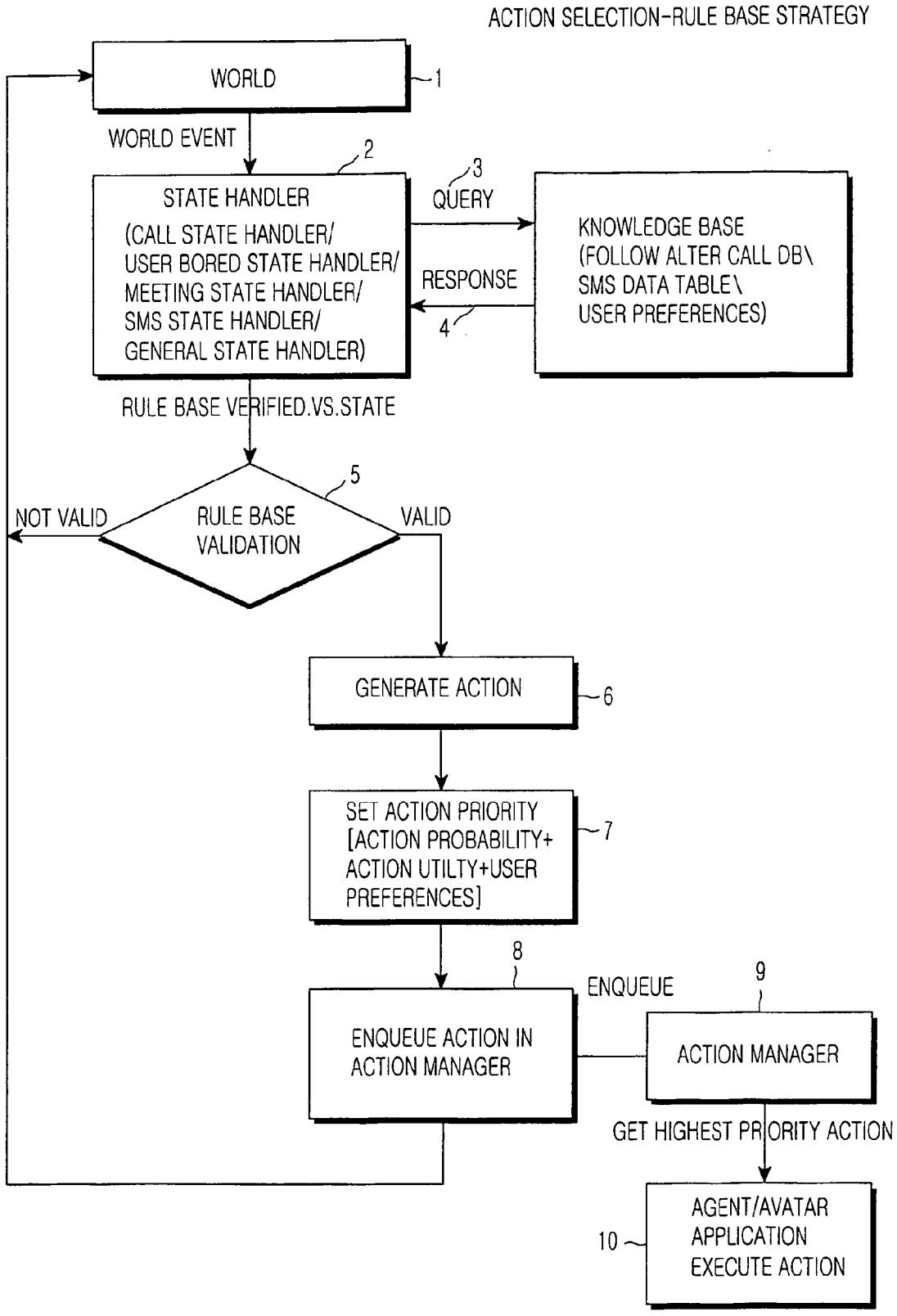

ACTION SELECTION-RULE BASE STRATEGY
WORLD
1
WORLD EVENT
2
STATE HANDLER
(CALL STATE HANDLER/
USER BORED STATE HANDLER/
MEETING STATE HANDLER/
SMS STATE HANDLER/
GENERAL STATE HANDLER)
3
QUERY
RESPONSE
4
KNOWLEDGE BASE
(FOLLOW ALTER CALL DB\
SMS DATA TABLE\
USER PREFERENCES)
RULE BASE VERIFIED.VS.STATE
5
NOT VALID
RULE BASE
VALIDATION
VALID
GENERATE ACTION
6
SET ACTION PRIORITY
[ACTION PROBABILITY+
ACTION UTILTY+USER
PREFERENCES]
7
8
ENQUEUE
9
ENQUEUE ACTION IN
ACTION MANAGER
ACTION MANAGER
GET HIGHEST PRIORITY ACTION
AGENT/AVATAR
APPLICATION
EXECUTE ACTION
10

FIG.9A

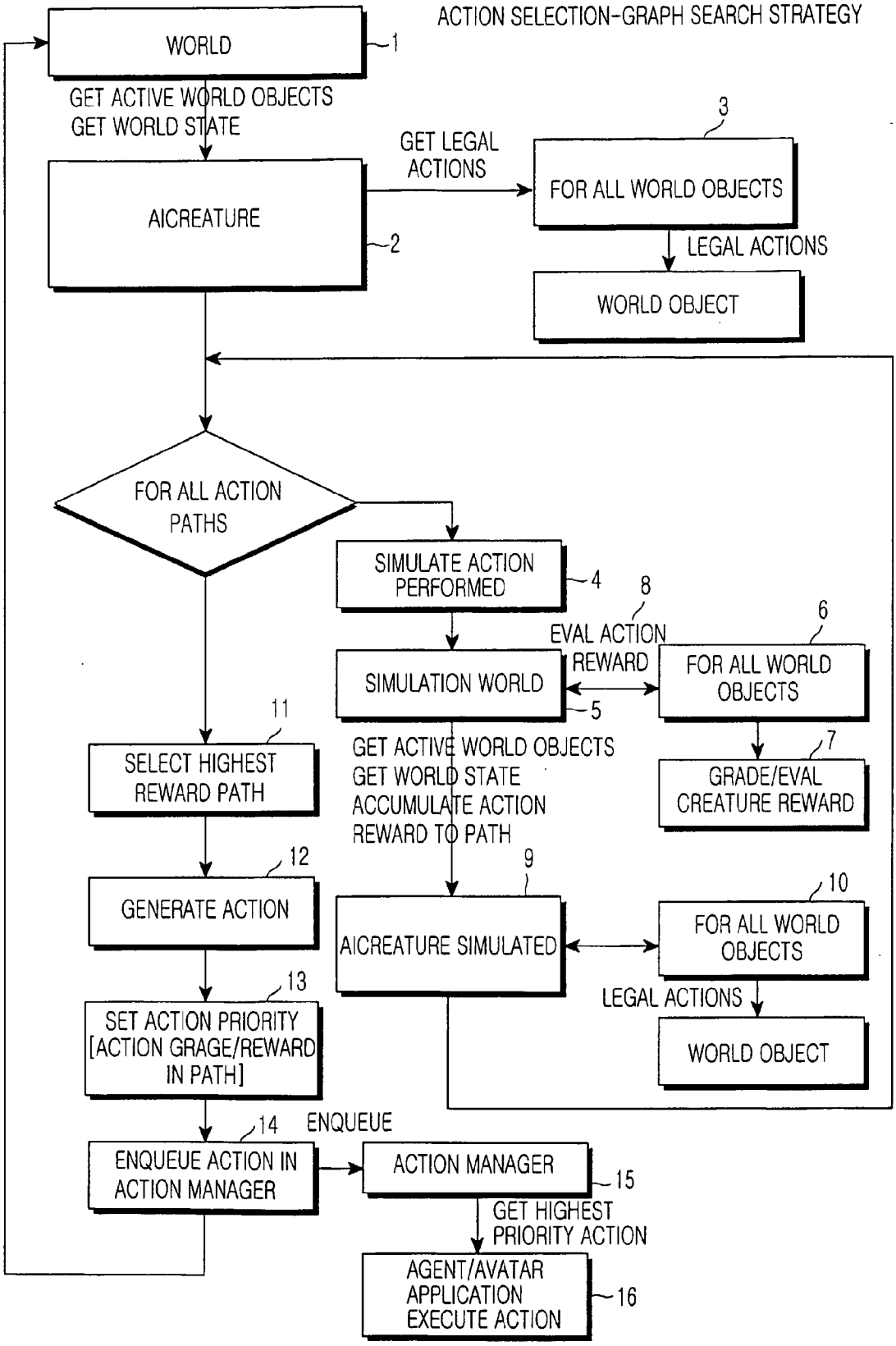

ACTION SELECTION-GRAPH SEARCH STRATEGY
WORLD
1
GET ACTIVE WORLD OBJECTS
GET WORLD STATE
AICREATURE
2
GET LEGAL ACTIONS
3
FOR ALL WORLD OBJECTS
LEGAL ACTIONS
WORLD OBJECT
FOR ALL ACTION PATHS
SIMULATE ACTION PERFORMED
4
8
EVAL ACTION REWARD
6
SIMULATION WORLD
5
FOR ALL WORLD OBJECTS
11
SELECT HIGHEST REWARD PATH
GET ACTIVE WORLD OBJECTS
GET WORLD STATE
ACCUMULATE ACTION REWARD TO PATH
7
GRADE/EVAL CREATURE REWARD
12
GENERATE ACTION
9
10
AICREATURE SIMULATED
FOR ALL WORLD OBJECTS
13
LEGAL ACTIONS
SET ACTION PRIORITY [ACTION GRAGE/REWARD IN PATH]
WORLD OBJECT
14
ENQUEUE
ENQUEUE ACTION IN ACTION MANAGER
ACTION MANAGER
15
GET HIGHEST PRIORITY ACTION
AGENT/AVATAR APPLICATION EXECUTE ACTION
16

FIG.9B

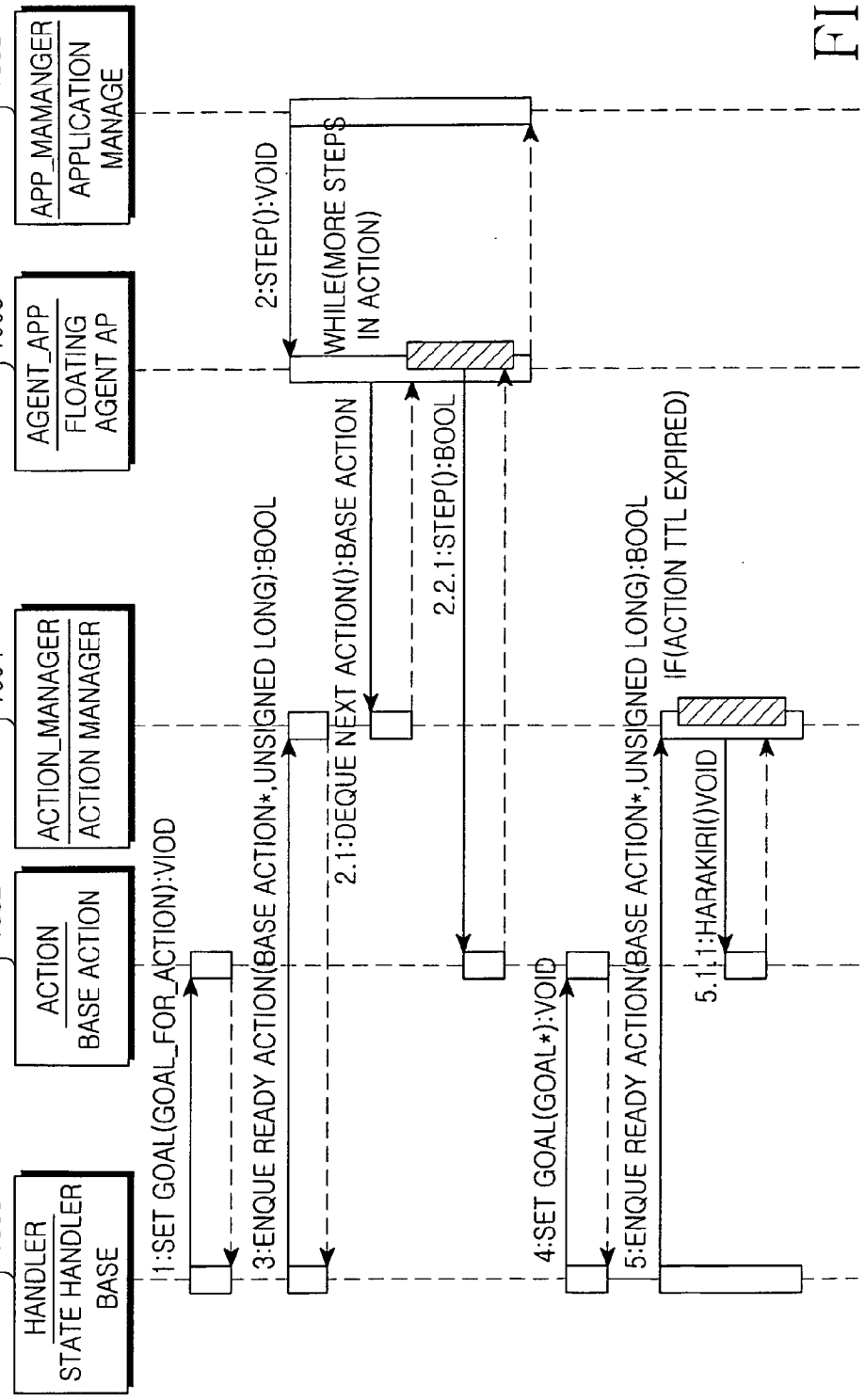

1000
HANDLER
STATE HANDLER
BASE
1002
ACTION
BASE ACTION
1004
ACTION_MANAGER
ACTION MANAGER
1006
AGENT_APP
FLOATING
AGENT AP
1008
APP_MAMANGER
APPLICATION
MANAGE
1:SET GOAL(GOAL_FOR_ACTION):VIOD
2:STEP():VOID
WHILE(MORE STEP$
IN ACTION)
3:ENQUE READY ACTION(BASE ACTION*,UNSIGNED LONG):BOOL
2.1:DEQUE NEXT ACTION():BASE ACTION
2.2.1:STEP():BOOL
4:SET GOAL(GOAL*):VOID
5:ENQUE READY ACTION(BASE ACTION*,UNSIGNED LONG):BOOL
IF(ACTION TTL EXPIRED)
5.1.1:HARAKIRI()VOID

FIG.10

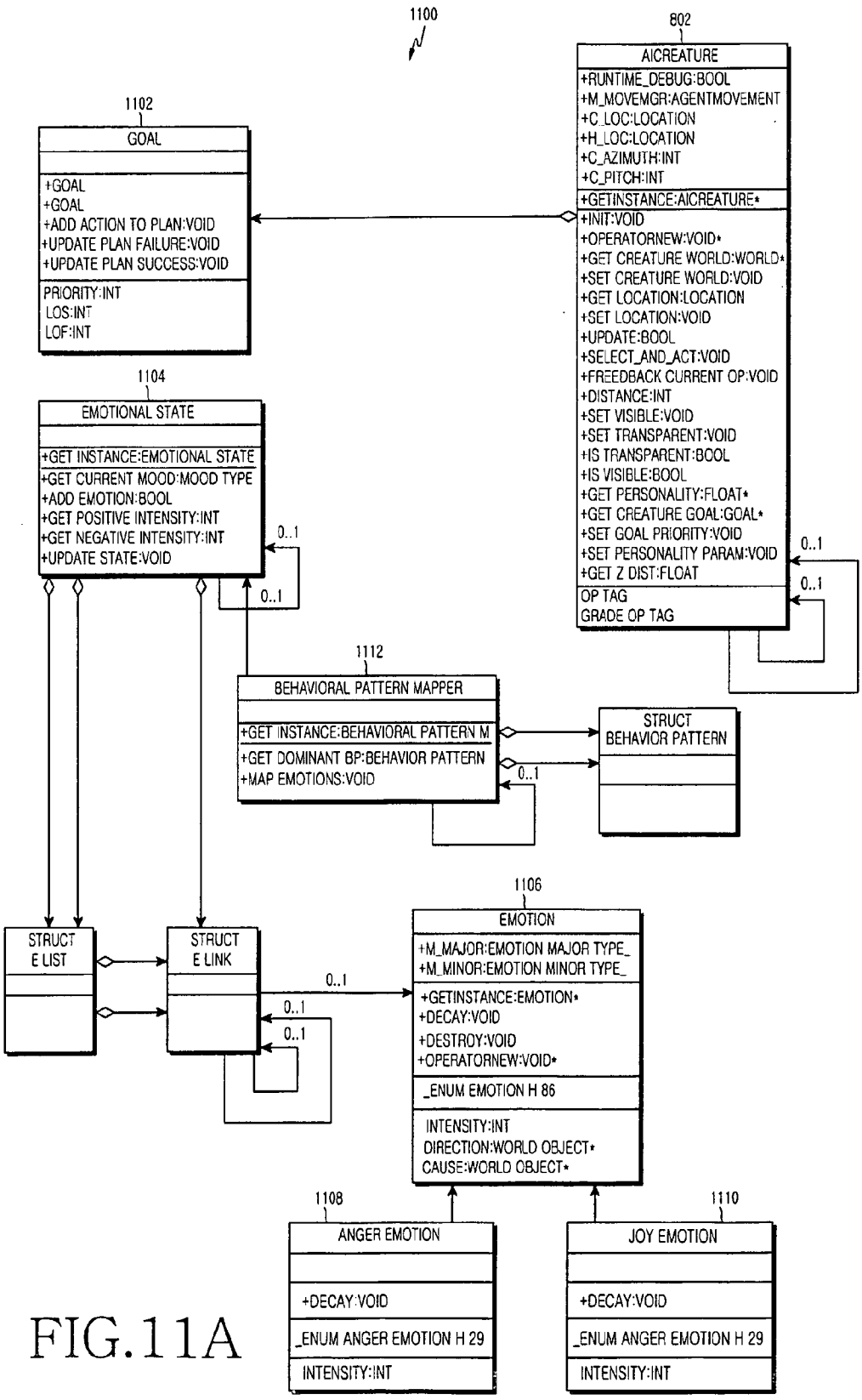

1100
802
AICREATURE
+RUNTIME_DEBUG:BOOL
+M_MOVEMGR:AGENTMOVEMENT
+C_LOC:LOCATION
+H_LOC:LOCATION
+C_AZIMUTH:INT
+C_PITCH:INT
+GETINSTANCE:AICREATURE*
+INIT:VOID
+OPERATORNEW:VOID*
+GET CREATURE WORLD:WORLD*
+SET CREATURE WORLD:VOID
+GET LOCATION:LOCATION
+SET LOCATION:VOID
+UPDATE:BOOL
+SELECT_AND_ACT:VOID
+FREEDBACK CURRENT OP:VOID
+DISTANCE:INT
+SET VISIBLE:VOID
+SET TRANSPARENT:VOID
+IS TRANSPARENT:BOOL
+IS VISIBLE:BOOL
+GET PERSONALITY:FLOAT*
+GET CREATURE GOAL:GOAL*
+SET GOAL PRIORITY:VOID
+SET PERSONALITY PARAM:VOID
+GET Z DIST:FLOAT
OP TAG
GRADE OP TAG
1102
GOAL
+GOAL
+GOAL
+ADD ACTION TO PLAN:VOID
+UPDATE PLAN FAILURE:VOID
+UPDATE PLAN SUCCESS:VOID
PRIORITY:INT
LOS:INT
LOF:INT
1104
EMOTIONAL STATE
+GET INSTANCE:EMOTIONAL STATE
+GET CURRENT MOOD:MOOD TYPE
+ADD EMOTION:BOOL
+GET POSITIVE INTENSITY:INT
+GET NEGATIVE INTENSITY:INT
+UPDATE STATE:VOID
1112
BEHAVIORAL PATTERN MAPPER
+GET INSTANCE:BEHAVIORAL PATTERN M
+GET DOMINANT BP:BEHAVIOR PATTERN
+MAP EMOTIONS:VOID
STRUCT BEHAVIOR PATTERN
1106
EMOTION
+M_MAJOR:EMOTION MAJOR TYPE_
+M_MINOR:EMOTION MINOR TYPE_
+GETINSTANCE:EMOTION*
+DECAY:VOID
+DESTROY:VOID
+OPERATORNEW:VOID*
_ENUM EMOTION H 86
INTENSITY:INT
DIRECTION:WORLD OBJECT*
CAUSE:WORLD OBJECT*
STRUCT E LIST
STRUCT E LINK
1108
ANGER EMOTION
+DECAY:VOID
_ENUM ANGER EMOTION H 29
INTENSITY:INT
1110
JOY EMOTION
+DECAY:VOID
_ENUM ANGER EMOTION H 29
INTENSITY:INT
0..1

FIG.11A

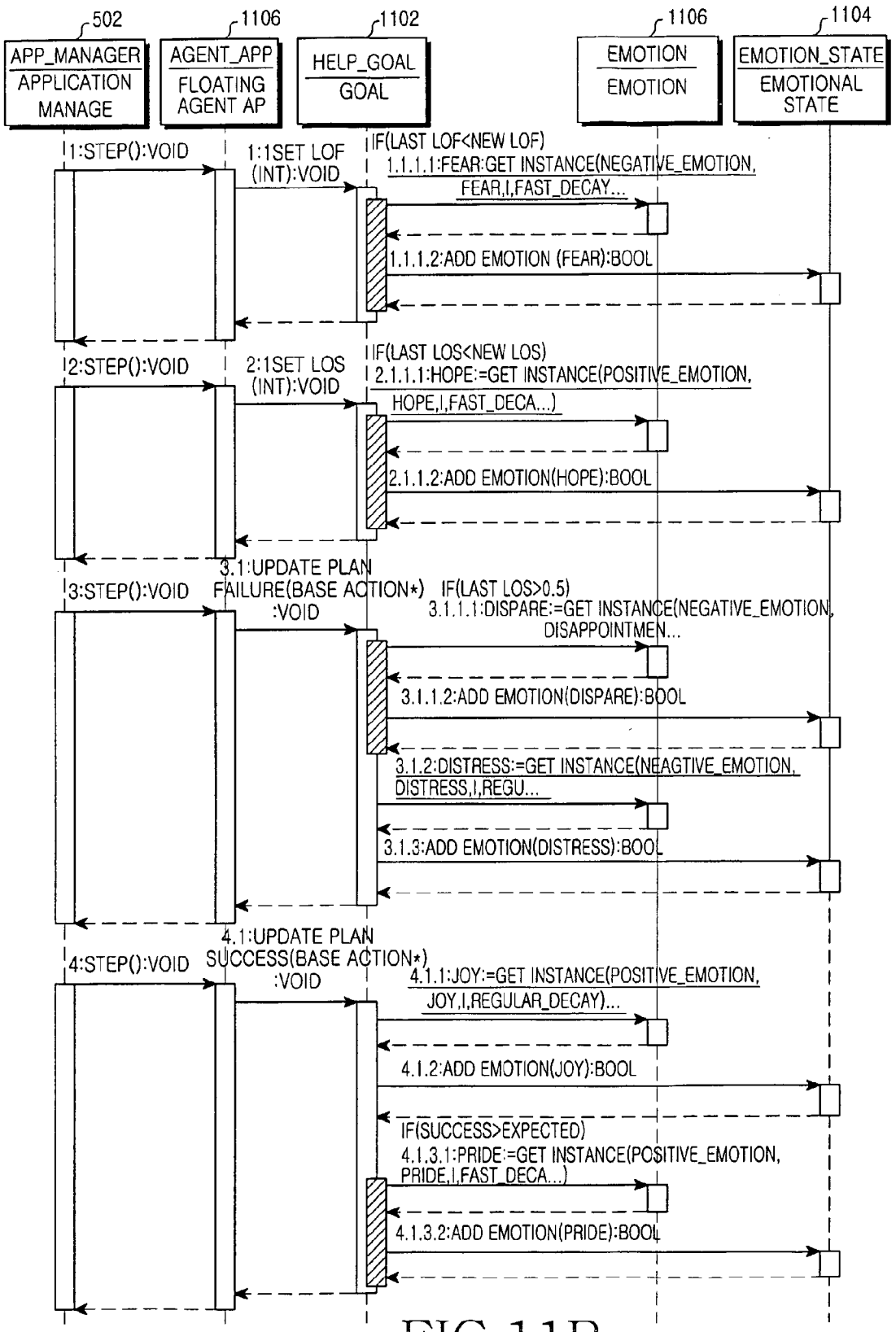

502
APP_MANAGER
APPLICATION MANAGE
1106
AGENT_APP
FLOATING AGENT AP
1102
HELP_GOAL
GOAL
1106
EMOTION
EMOTION
1104
EMOTION_STATE
EMOTIONAL STATE
1:STEP():VOID
1:1SET LOF (INT):VOID
IF(LAST LOF<NEW LOF)
1.1.1.1:FEAR:GET INSTANCE(NEGATIVE_EMOTION, FEAR,I,FAST_DECAY...
1.1.1.2:ADD EMOTION (FEAR):BOOL
2:STEP():VOID
2:1SET LOS (INT):VOID
IF(LAST LOS<NEW LOS)
2.1.1.1:HOPE:=GET INSTANCE(POSITIVE_EMOTION, HOPE,I,FAST_DECA...)
2.1.1.2:ADD EMOTION(HOPE):BOOL
3:STEP():VOID
3.1:UPDATE PLAN FAILURE(BASE ACTION*) :VOID
IF(LAST LOS>0.5)
3.1.1.1:DISPARE:=GET INSTANCE(NEGATIVE_EMOTION, DISAPPOINTMEN...
3.1.1.2:ADD EMOTION(DISPARE):BOOL
3.1.2:DISTRESS:=GET INSTANCE(NEAGTIVE_EMOTION, DISTRESS,I,REGU...
3.1.3:ADD EMOTION(DISTRESS):BOOL
4:STEP():VOID
4.1:UPDATE PLAN SUCCESS(BASE ACTION*) :VOID
4.1.1:JOY:=GET INSTANCE(POSITIVE_EMOTION, JOY,I,REGULAR_DECAY)...
4.1.2:ADD EMOTION(JOY):BOOL
IF(SUCCESS>EXPECTED)
4.1.3.1:PRIDE:=GET INSTANCE(POSITIVE_EMOTION, PRIDE,I,FAST_DECA...)
4.1.3.2:ADD EMOTION(PRIDE):BOOL

FIG.11B

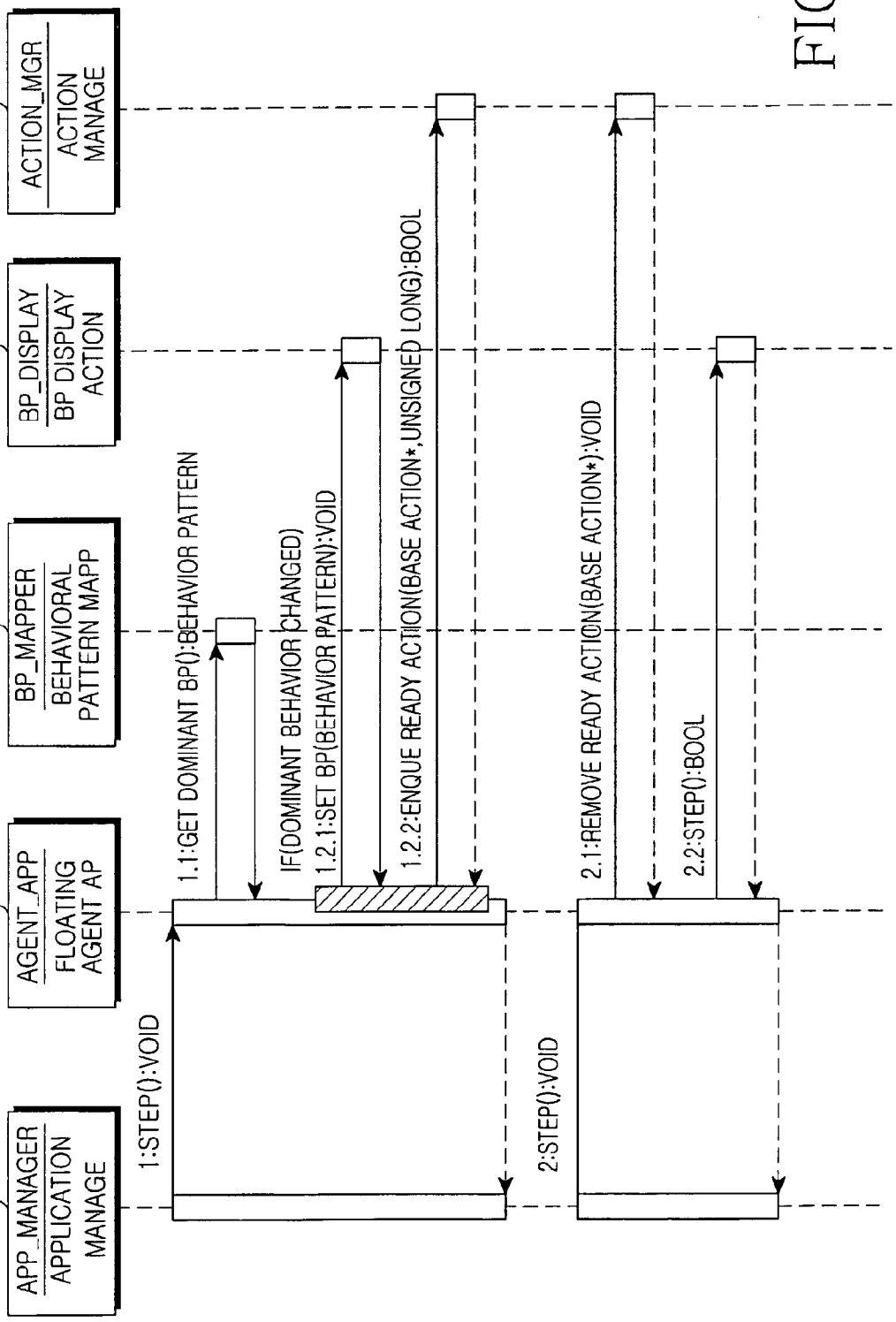

APP_MANAGER
APPLICATION MANAGE
502
AGENT_APP
FLOATING AGENT AP
1006
BP_MAPPER
BEHAVIORAL PATTERN MAPP
1108
BP_DISPLAY
BP DISPLAY ACTION
1110
ACTION_MGR
ACTION MANAGE
1004
1:STEP():VOID
1.1:GET DOMINANT BP():BEHAVIOR PATTERN
IF(DOMINANT BEHAVIOR CHANGED)
1.2.1:SET BP(BEHAVIOR PATTERN):VOID
1.2.2:ENQUE READY ACTION(BASE ACTION*,UNSIGNED LONG):BOOL
2:STEP():VOID
2.1:REMOVE READY ACTION(BASE ACTION*):VOID
2.2:STEP():BOOL

FIG.11C

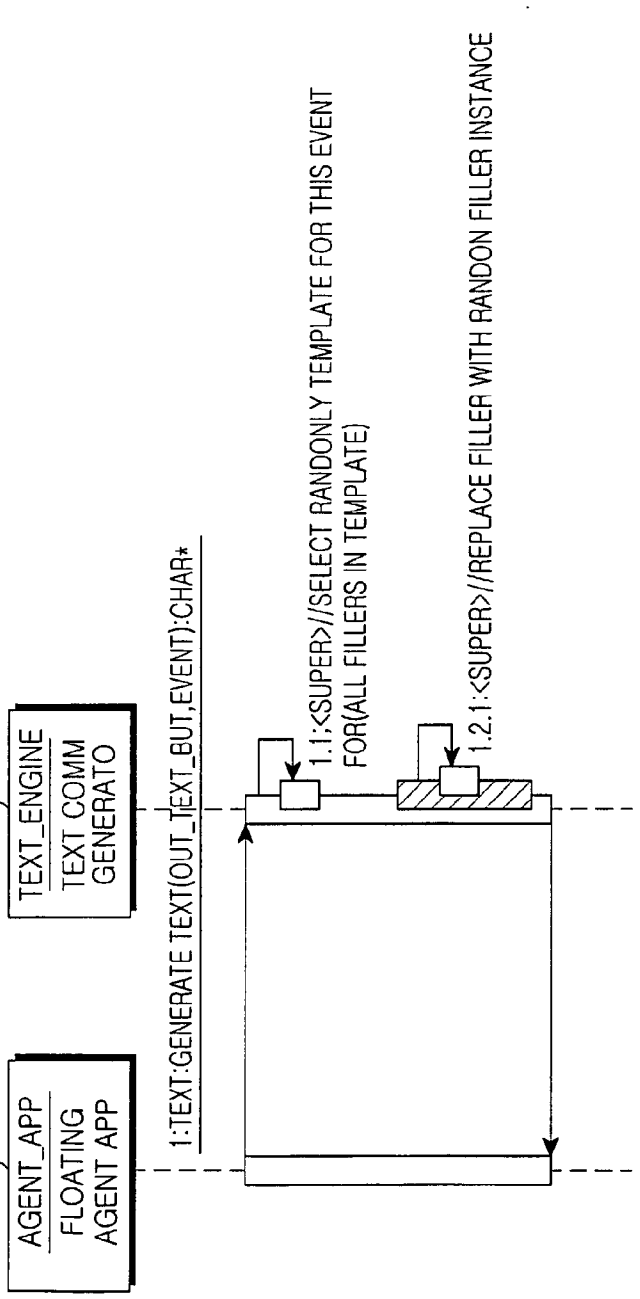

AGENT_APP
FLOATING AGENT APP
1006
TEXT_ENGINE
TEXT COMM GENERATO
1200
1:TEXT:GENERATE TEXT(OUT_TEXT_BUT,EVENT):CHAR*
1.1:<SUPER>//SELECT RANDONLY TEMPLATE FOR THIS EVENT FOR(ALL FILLERS IN TEMPLATE)
1.2.1:<SUPER>//REPLACE FILLER WITH RANDON FILLER INSTANCE

FIG.12A

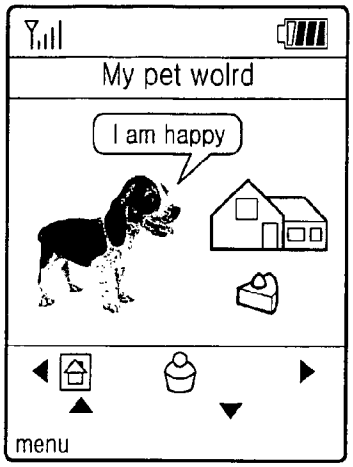


FIG.12B

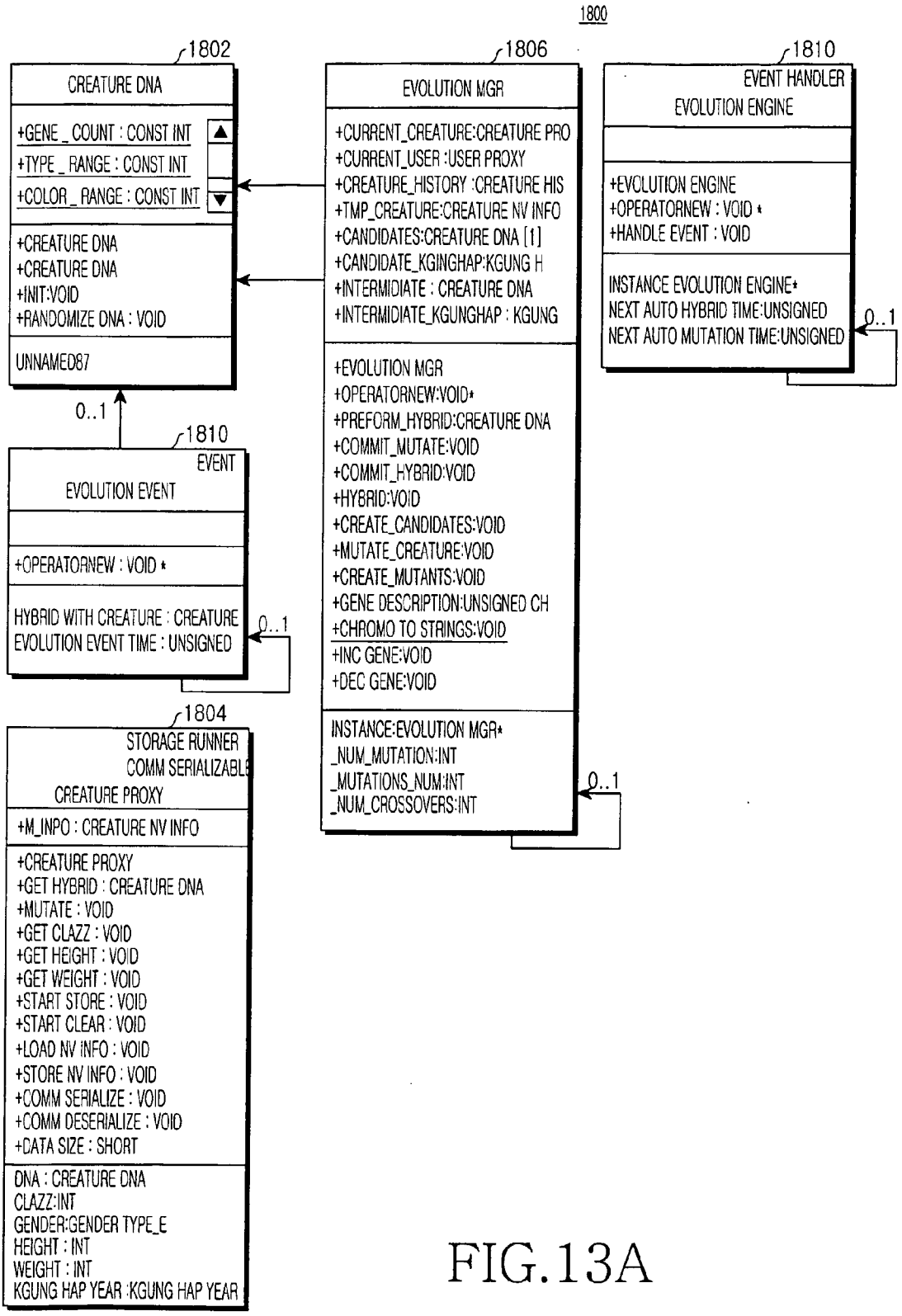

1800
1802
CREATURE DNA
+GENE _ COUNT : CONST INT
+TYPE _ RANGE : CONST INT
+COLOR _ RANGE : CONST INT
+CREATURE DNA
+CREATURE DNA
+INIT:VOID
+RANDOMIZE DNA : VOID
UNNAMED87
1806
EVOLUTION MGR
+CURRENT_CREATURE:CREATURE PRO
+CURRENT_USER :USER PROXY
+CREATURE_HISTORY :CREATURE HIS
+TMP_CREATURE:CREATURE NV INFO
+CANDIDATES:CREATURE DNA [1]
+CANDIDATE_KGINGHAP:KGUNG H
+INTERMIDIATE : CREATURE DNA
+INTERMIDIATE_KGUNGHAP : KGUNG
+EVOLUTION MGR
+OPERATORNEW:VOID*
+PREFORM_HYBRID:CREATURE DNA
+COMMIT_MUTATE:VOID
+COMMIT_HYBRID:VOID
+HYBRID:VOID
+CREATE_CANDIDATES:VOID
+MUTATE_CREATURE:VOID
+CREATE_MUTANTS:VOID
+GENE DESCRIPTION:UNSIGNED CH
+CHROMO TO STRINGS:VOID
+INC GENE:VOID
+DEC GENE:VOID
INSTANCE:EVOLUTION MGR*
_NUM_MUTATION:INT
_MUTATIONS_NUM:INT
_NUM_CROSSOVERS:INT
1810
EVENT HANDLER
EVOLUTION ENGINE
+EVOLUTION ENGINE
+OPERATORNEW : VOID *
+HANDLE EVENT : VOID
INSTANCE EVOLUTION ENGINE*
NEXT AUTO HYBRID TIME:UNSIGNED
NEXT AUTO MUTATION TIME:UNSIGNED
0..1
0..1
1810
EVENT
EVOLUTION EVENT
+OPERATORNEW : VOID *
HYBRID WITH CREATURE : CREATURE
EVOLUTION EVENT TIME : UNSIGNED
0..1
1804
STORAGE RUNNER
COMM SERIALIZABLE
CREATURE PROXY
+M_INPO : CREATURE NV INFO
+CREATURE PROXY
+GET HYBRID : CREATURE DNA
+MUTATE : VOID
+GET CLAZZ : VOID
+GET HEIGHT : VOID
+GET WEIGHT : VOID
+START STORE : VOID
+START CLEAR : VOID
+LOAD NV INFO : VOID
+STORE NV INFO : VOID
+COMM SERIALIZE : VOID
+COMM DESERIALIZE : VOID
+DATA SIZE : SHORT
DNA : CREATURE DNA
CLAZZ:INT
GENDER:GENDER TYPE_E
HEIGHT : INT
WEIGHT : INT
KGUNG HAP YEAR :KGUNG HAP YEAR

FIG.13A

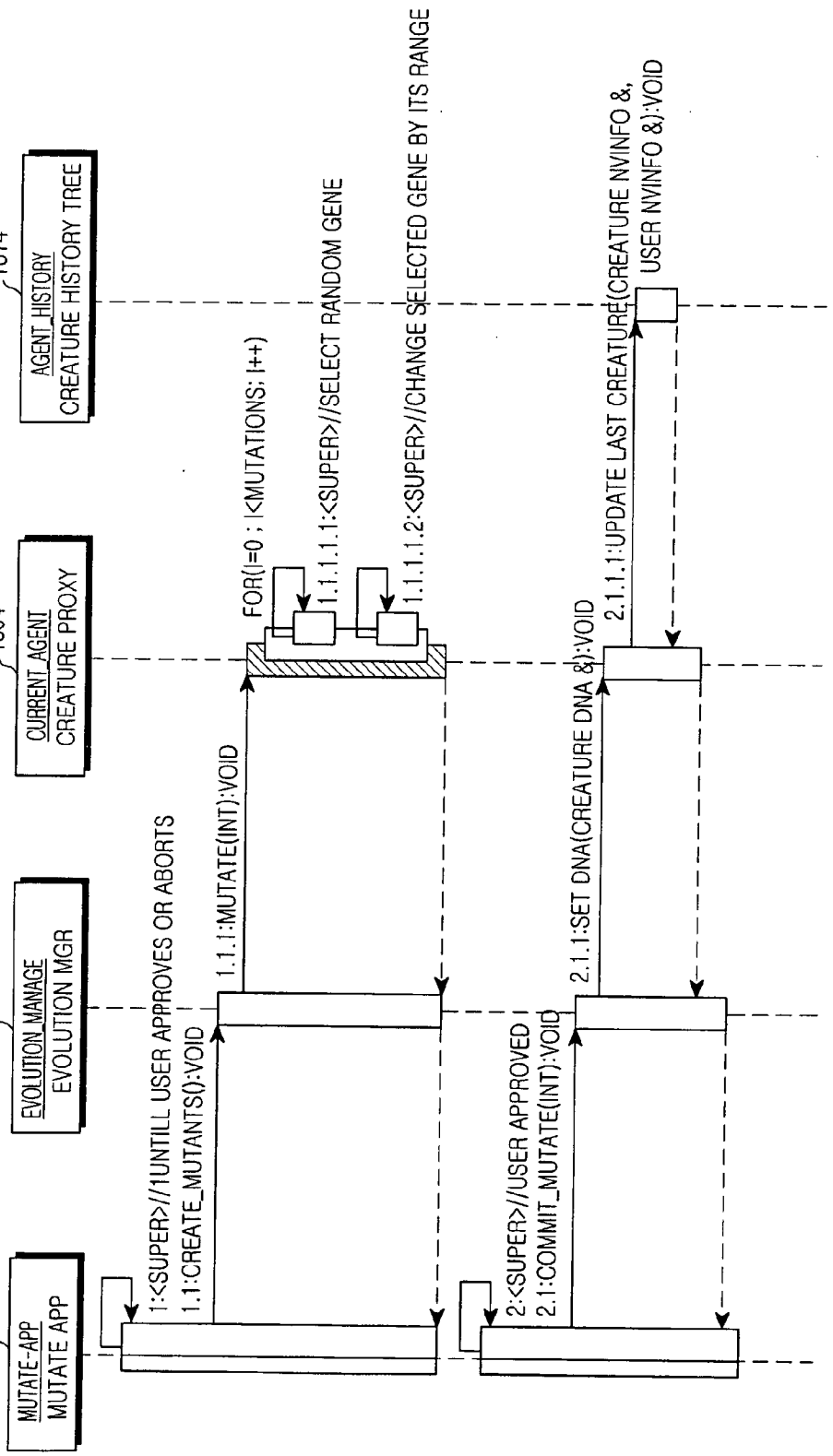

1812
MUTATE-APP
MUTATE APP
1806
EVOLUTION_MANAGE
EVOLUTION MGR
1804
CURRENT_AGENT
CREATURE PROXY
1814
AGENT_HISTORY
CREATURE HISTORY TREE
1:<SUPER>//1UNTILL USER APPROVES OR ABORTS
1.1:CREATE_MUTANTS():VOID
1.1.1:MUTATE(INT):VOID
FOR(I=0 ; I<MUTATIONS; I++)
1.1.1.1.1:<SUPER>//SELECT RANDOM GENE
1.1.1.1.2:<SUPER>//CHANGE SELECTED GENE BY ITS RANGE
2:<SUPER>//USER APPROVED
2.1:COMMIT_MUTATE(INT):VOID
2.1.1:SET DNA(CREATURE DNA &):VOID
2.1.1.1:UPDATE LAST CREATURE(CREATURE NVINFO &,
USER NVINFO &):VOID

FIG.13B

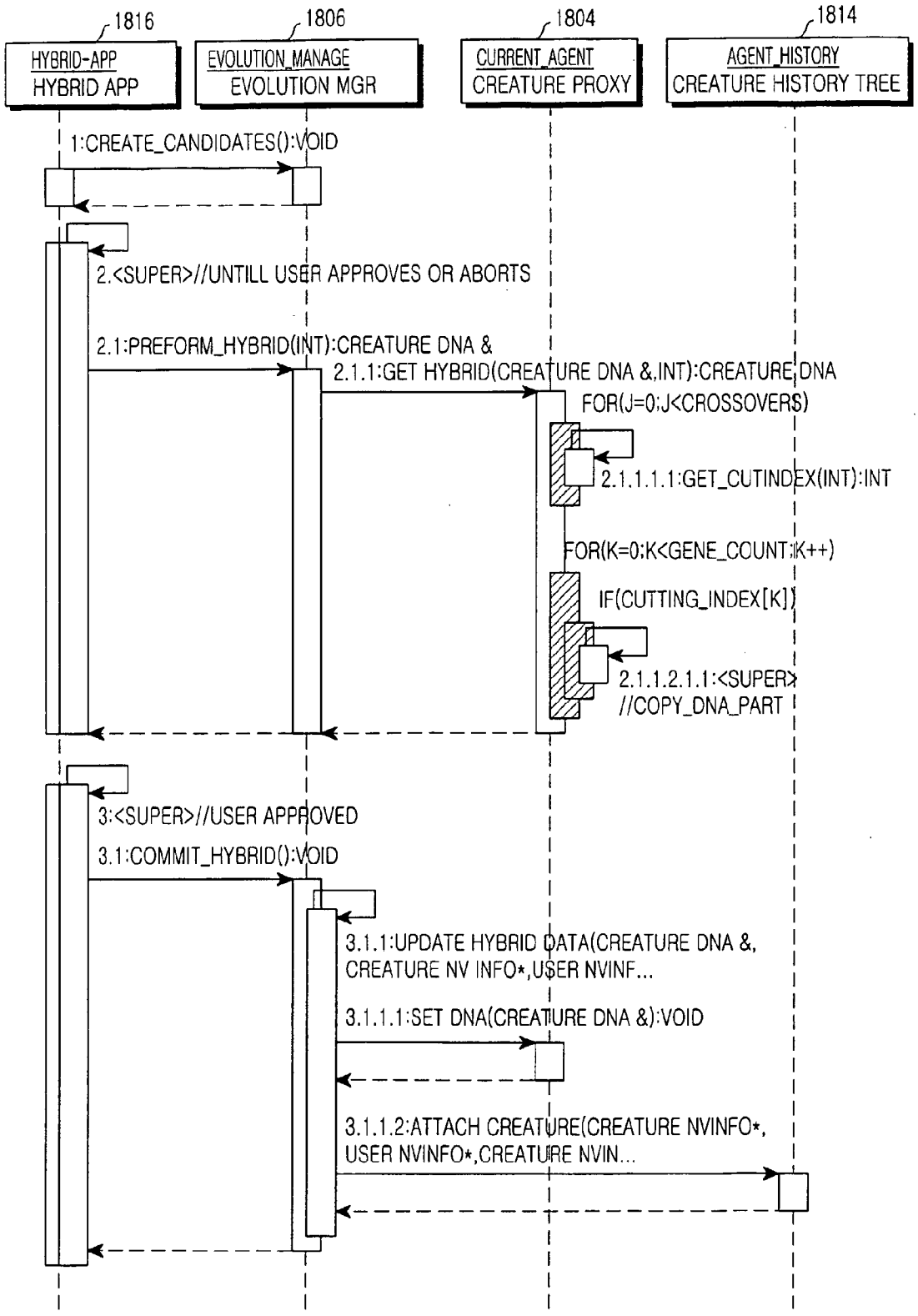

1816
HYBRID-APP
HYBRID APP
1806
EVOLUTION_MANAGE
EVOLUTION MGR
1804
CURRENT_AGENT
CREATURE PROXY
1814
AGENT_HISTORY
CREATURE HISTORY TREE
1:CREATE_CANDIDATES():VOID
2.<SUPER>//UNTILL USER APPROVES OR ABORTS
2.1:PREFORM_HYBRID(INT):CREATURE DNA &
2.1.1:GET HYBRID(CREATURE DNA &,INT):CREATURE DNA
FOR(J=0;J<CROSSOVERS)
2.1.1.1.1:GET_CUTINDEX(INT):INT
FOR(K=0;K<GENE_COUNT;K++)
IF(CUTTING_INDEX[K])
2.1.1.2.1.1:<SUPER>
//COPY_DNA_PART
3:<SUPER>//USER APPROVED
3.1:COMMIT_HYBRID():VOID
3.1.1:UPDATE HYBRID DATA(CREATURE DNA &,
CREATURE NV INFO*,USER NVINF...
3.1.1.1:SET DNA(CREATURE DNA &):VOID
3.1.1.2:ATTACH CREATURE(CREATURE NVINFO*,
USER NVINFO*,CREATURE NVIN...

FIG.13C

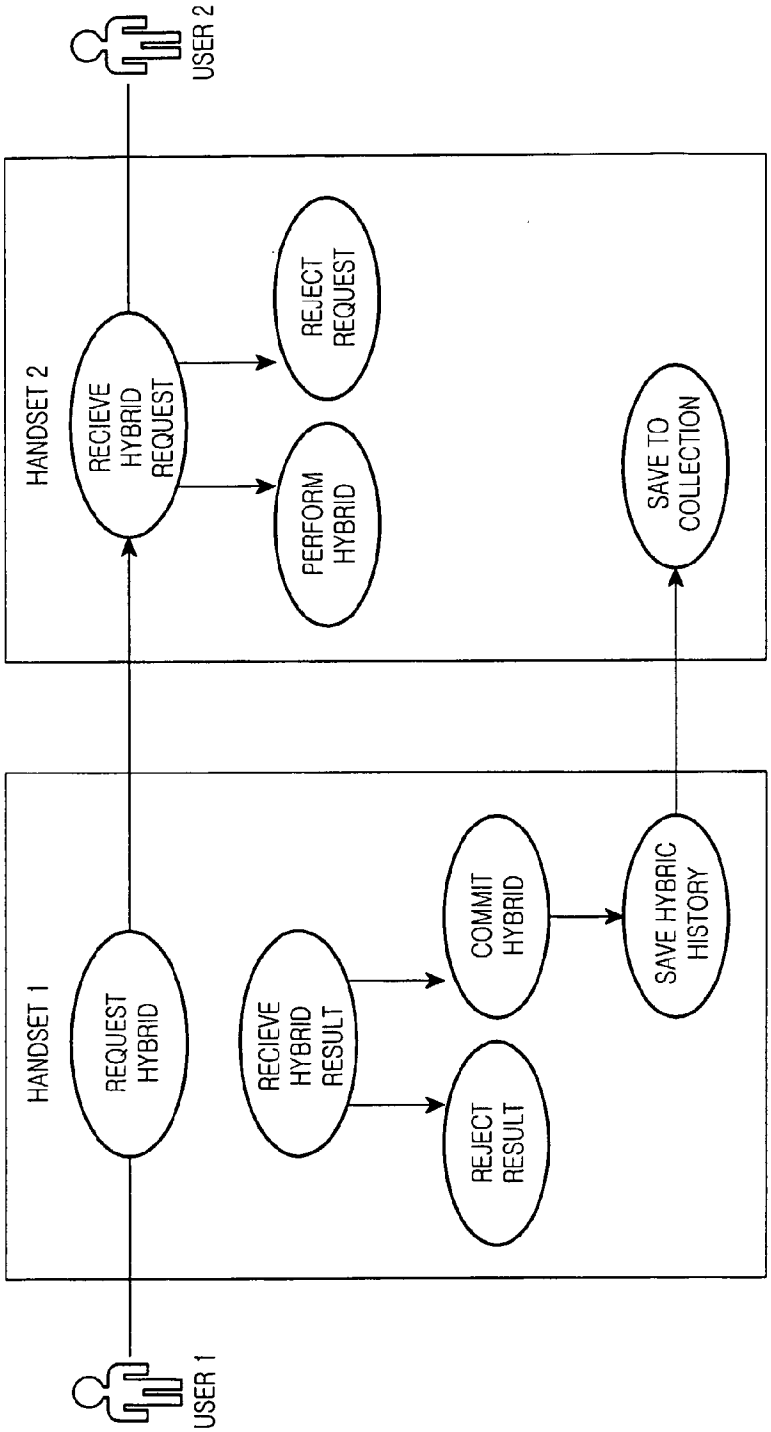

USER 2
HANDSET 2
RECIEVE HYBRID REQUEST
REJECT REQUEST
PERFORM HYBRID
SAVE TO COLLECTION
HANDSET 1
REQUEST HYBRID
RECIEVE HYBRID RESULT
COMMIT HYBRID
REJECT RESULT
SAVE HYBRIC HISTORY
USER 1

FIG.14

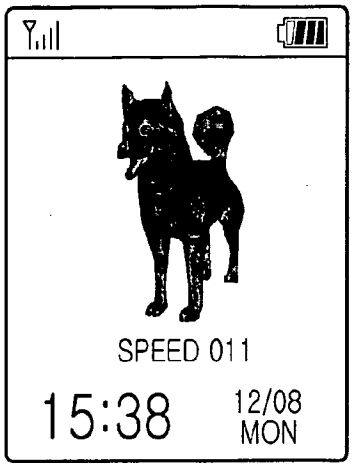


FIG.15

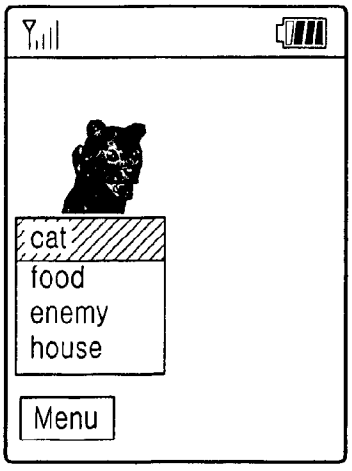


FIG.16

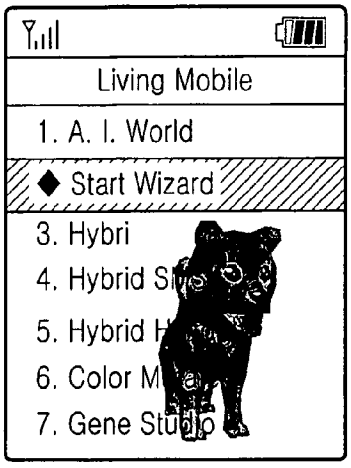


FIG.17A

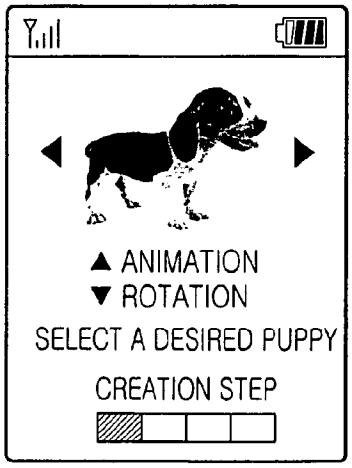


FIG.17B

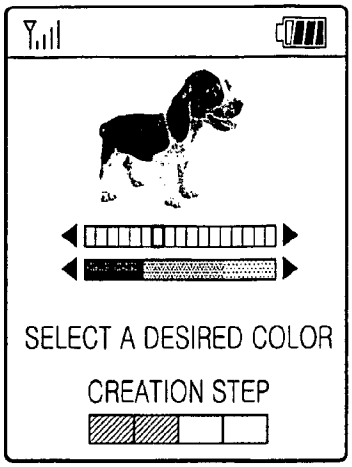


FIG.17C

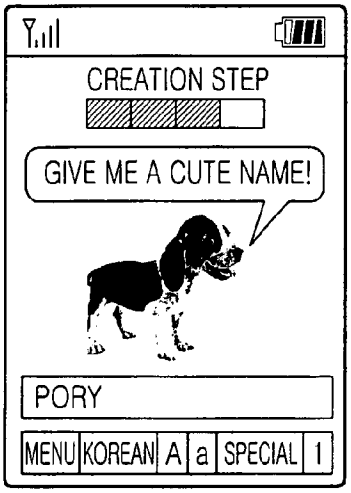


FIG.17D

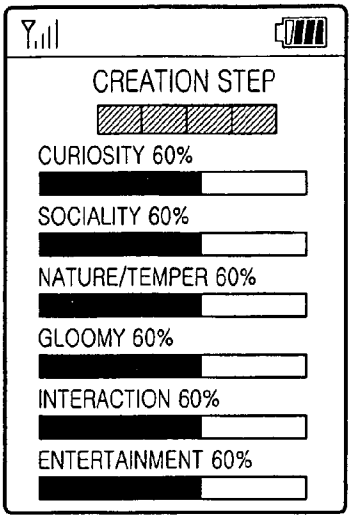


FIG.17E

FIG.17F

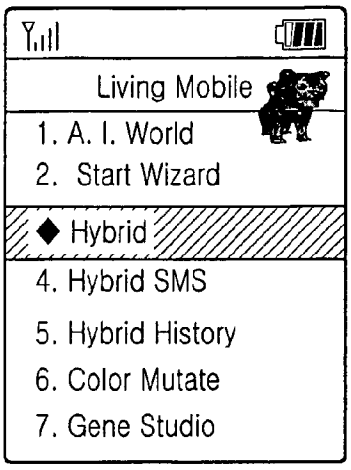


FIG.18

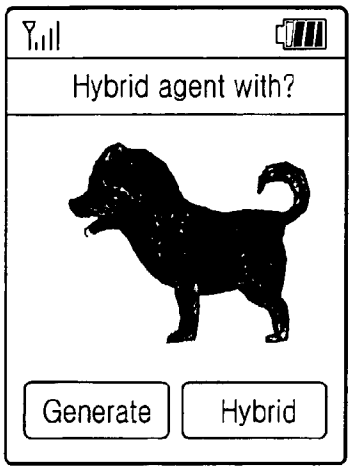


FIG.19A

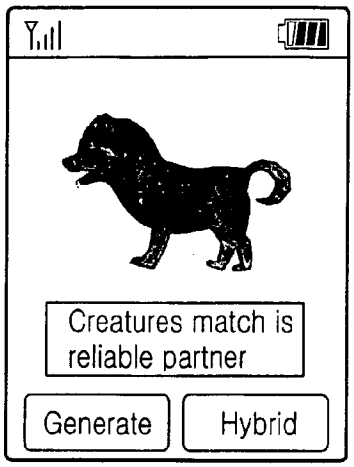


FIG.19B

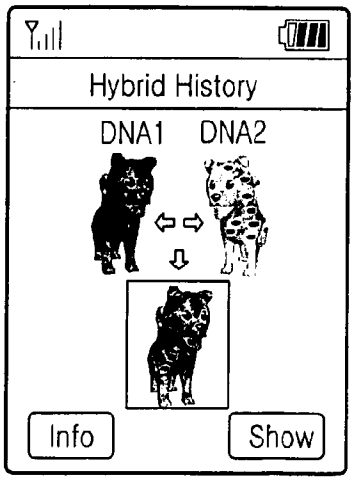


FIG.20

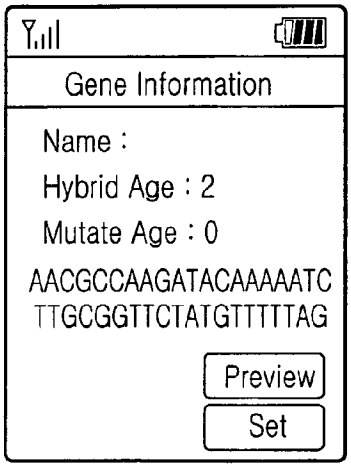

Gene Information
Name :
Hybrid Age : 2
Mutate Age : 0
AACGCCAAGATACAAAAATC
TTGCGGTTCTATGTTTTTAG
Preview
Set

FIG.21

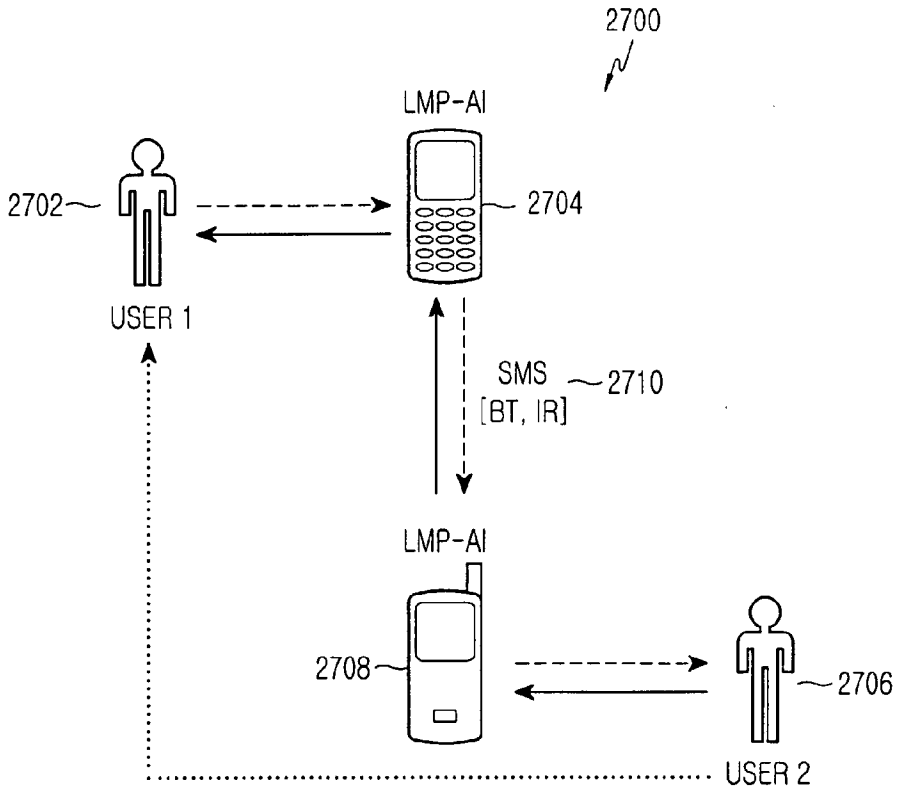

2700
LMP-AI
2702
2704
USER 1
SMS
[BT, IR]
2710
LMP-AI
2708
2706
USER 2

FIG.22

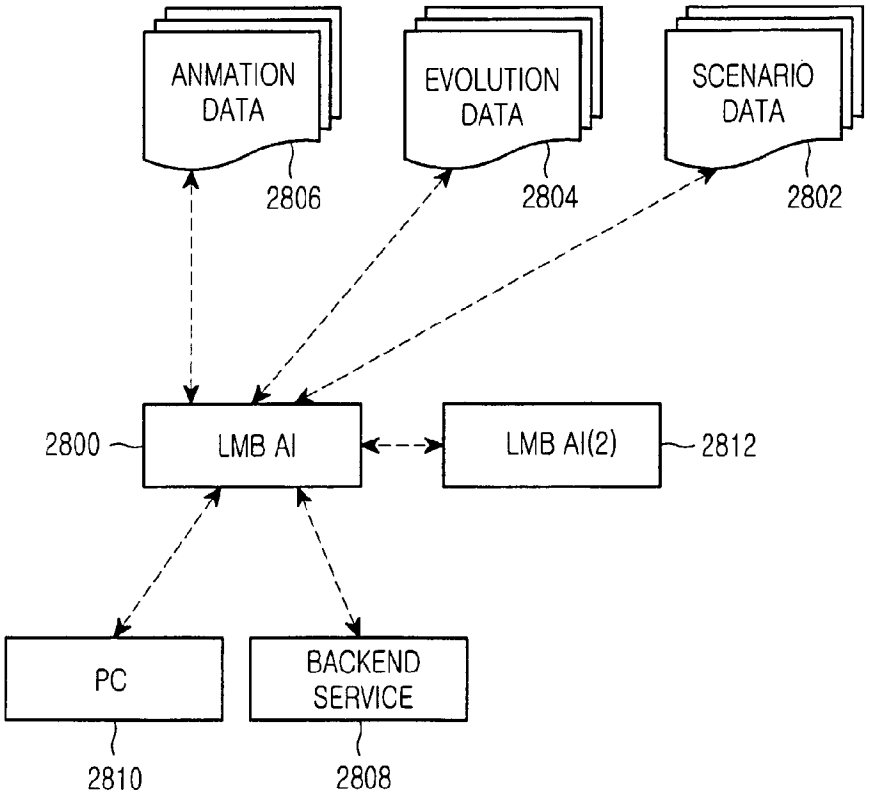

ANMATION DATA
2806
EVOLUTION DATA
2804
SCENARIO DATA
2802
2800
LMB AI
LMB AI(2)
2812
PC
2810
BACKEND SERVICE
2808

FIG.23

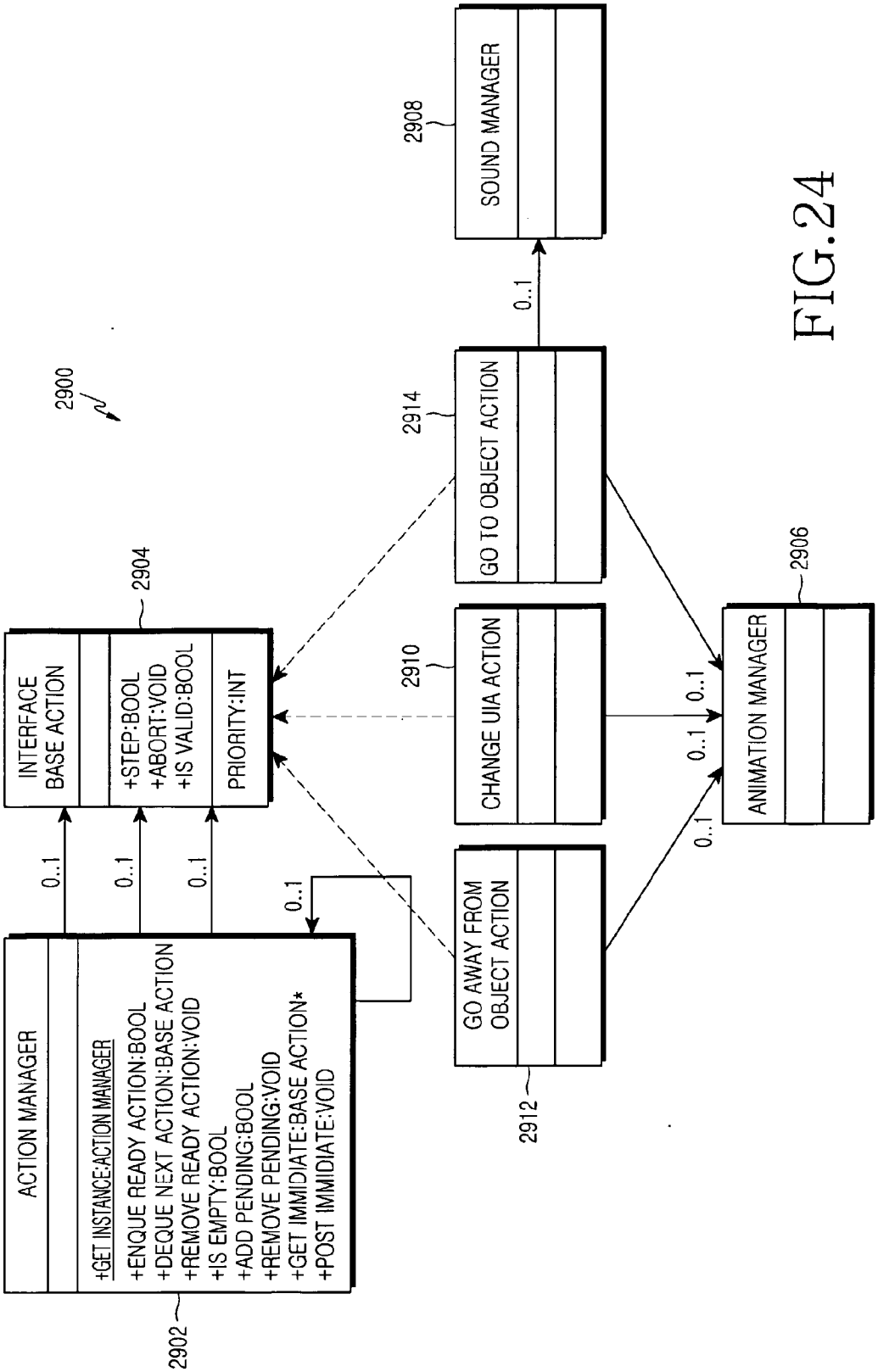

2900
2902
ACTION MANAGER
+GET INSTANCE:ACTION MANAGER
+ENQUE READY ACTION:BOOL
+DEQUE NEXT ACTION:BASE ACTION
+REMOVE READY ACTION:VOID
+IS EMPTY:BOOL
+ADD PENDING:BOOL
+REMOVE PENDING:VOID
+GET IMMIDIATE:BASE ACTION*
+POST IMMIDIATE:VOID
2904
INTERFACE BASE ACTION
+STEP:BOOL
+ABORT:VOID
+IS VALID:BOOL
PRIORITY:INT
2906
ANIMATION MANAGER
2908
SOUND MANAGER
2910
CHANGE UIA ACTION
2912
GO AWAY FROM OBJECT ACTION
2914
GO TO OBJECT ACTION
0..1

FIG.24

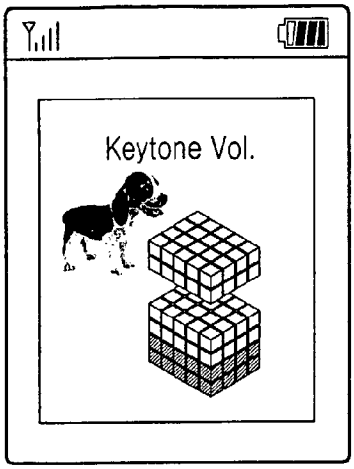


FIG.25A

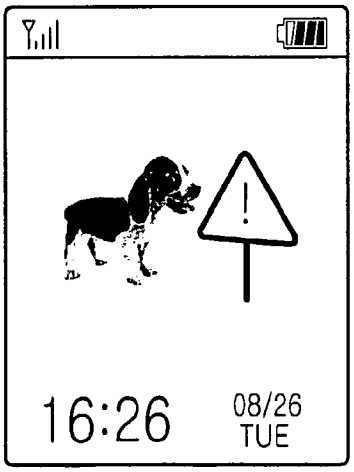


FIG.25B

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
4 September 2003 (04.09.2003)

PCT

(10) International Publication Number
WO 03/073417 A2

(51) International Patent Classification[7]: G10L 21/00

(21) International Application Number: PCT/US03/06218

(22) International Filing Date: 26 February 2003 (26.02.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/359,348	26 February 2002 (26.02.2002)	US
10/134,679	30 April 2002 (30.04.2002)	US
10/158,213	31 May 2002 (31.05.2002)	US
10/184,113	28 June 2002 (28.06.2002)	US

(71) Applicant *(for all designated States except US)*: **SAP AKTIENGESELLSCHAFT** [DE/DE]; Intellectual Property Department, Neurottstrasse 16, 69190 Walldorf (DE).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **GONG, Li** [CN/US]; 1828 15th Street, Apartment #3, San Francisco, CA 94103 (US).

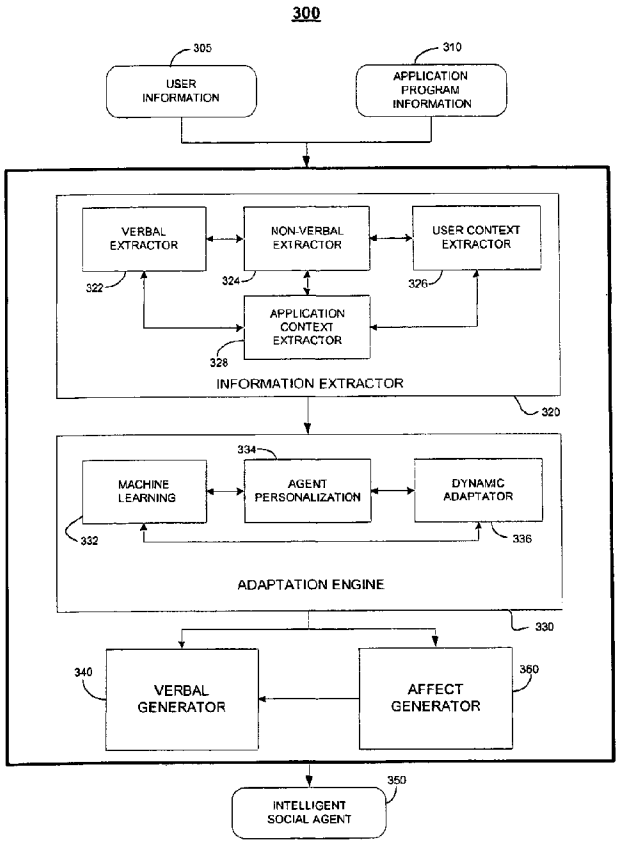
(74) Agent: **TROESCH, Hans, R.**; Fish & Richardson P.C., 500 Arguello Street #500, Redwood City, CA 94063 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),

[Continued on next page]

(54) Title: INTELLIGENT PERSONAL ASSISTANTS



(57) Abstract: An intelligent social agent is an animated computer interface agent with social intelligence that has been developed for a given application or type of applications and a particular user population. The social intelligence of the agent comes from the ability of the agent to be appealing, affective, adaptive, and an appropriate when interacting with the user. An intelligent personal assistant is an implementation of an intelligent social agent that assists a user in operating a computing device and using application programs on a computing device.

WO 03/073417 A2

European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Published:

— *without international search report and to be republished upon receipt of that report*

INTELLIGENT PERSONAL ASSISTANTS

TECHNICAL FIELD

This description relates to techniques for developing and using a computer interface agent to assist a computer system user.

BACKGROUND

A computer system may be used to accomplish many tasks. A user of a computer system may be assisted by a computer interface agent that provides information to the user or performs a service for the user.

SUMMARY

In one general aspect, implementing an intelligent personal assistant includes receiving an input associated with a user and an input associated with an application program, and accessing a user profile associated with the user. Context information is extracted from the received input, and the context information and the user profile are processed to produce an adaptive response by the intelligent personal assistant.

Implementations may include one or more of the following features. For example, the application program may be a personal information management application program, an application program to operate a computing device, an entertainment application program, or a game.

An adaptive response by the intelligent personal assistant may be associated with a personal information management application program, an application program to operate a computing device, an entertainment application program, or a game.

In another general aspect, an apparatus for implementing an intelligent social agent includes an information extractor, an adaptation engine, and an output generator. The information extractor is configured to access a user profile associated with the user, receive an input associated with a user, and extract context information from the received input. The adaptation engine is configured to receive the context information and the user profile from the information extractor and process the context information and the user

profile to produce an adaptive output. The output generator is configured to receive the adaptive output and represent the adaptive output in the intelligent social agent.

Implementations may include one or more of the features noted above and one or more of the following features. For example, the information extractor may be configured to receive physiological data or application program information associated with the user. The information extractor may be configured to extract information about an affective state of the user from physiological information associated with the user, vocal analysis information associated with the user by extracting verbal content and analyzing speech characteristics of the user, or verbal information from the user. Extracting context information also may include extracting a geographical position of the user and extracting information based on the geographical position of the user by using a global positioning system. Extracting context information may include extracting information about the application context associated with the user or about a linguistic style of the user.

An output generator may be a verbal generator, the adaptation engine may be configured to produce a verbal expression, and the verbal generator may produce the verbal expression in the intelligent social agent. An output generator may be an affect generator, the adaptation engine may be configured to produce a facial expression, and the affect generator may produce the facial expression in the intelligent social agent. The output generator may be a multi-modal generator that represents an adaptive output in the intelligent social agent using at least one of two modes. One mode may be a verbal mode and another mode may be an affect mode. The adaptive engine may be configured to produce a facial expression and a verbal expression that is represented in the intelligent social agent by the multi-modal output generator. The adaptation engine may be configured to produce an emotional expression in the intelligent social agent. The output generator may be configured to represent the emotional expression in the intelligent social agent.

In yet another general aspect, implementing an intelligent social agent includes receiving an input associated with a user, accessing a user profile associated with the user, extracting context information from the received input, and processing the context information and the user profile to produce an adaptive output to be represented by the intelligent social agent.

Implementations may include one of more of the features noted above or one or more of the following features. For example, the input associated with the user may include physiological data or application program information associated with the user. Extracting context information may include extracting information about an affective state of the user from physiological information, vocal analysis information, or verbal information associated with a user. Extracting context information also may include extracting a geographical position of the user and extracting information based on the geographical position of the user. Extracting context information may include extracting information about the application context associated with the user or about a linguistic style of the user. An adaptive output to be represented by the intelligent social agent may be a verbal expression, a facial expression, or an emotional expression.

Implementations of any of the techniques described above may include a method or process, a computer program on computer-readable media, a system or an apparatus, or a mobile device for implementing an intelligent social agent that interacts with a user or other type of system.

The details of one or more of the implementations are set forth in the accompanying drawings and description below. Other features and advantages will be apparent from the descriptions and drawings, and from the claims.

DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a programmable system for developing and using an intelligent social agent.

FIG. 2 is a block diagram of a computing device on which an intelligent social agent operates.

FIG. 3 is a block diagram illustrating an architecture of a social intelligence engine.

FIGS. 4A and 4B are flow charts of processes for extracting affective and physiological states of the user.

FIG. 5 is a flow chart of a process for adapting an intelligent social agent to the user and the context.

FIG. 6 is a flow chart of a process for casting an intelligent social agent.

FIGS. 7-10 are block diagrams showing various aspects of an architecture of an intelligent personal assistant.

Like reference symbols in the various drawings indicate like elements.

DETAILED DESCRIPTION

Referring to FIG. 1, a programmable system 100 for developing and using an intelligent social agent includes a variety of input/output (I/O) devices (e.g., a mouse 102, a keyboard 103, a display 104, a voice recognition and speech synthesis device 105, a video camera 106, a touch input device with stylus 107, a personal digital assistant or "PDA" 108, and a mobile phone 109) operable to communicate with a computer 110 having a central processor unit (CPU) 120, an I/O unit 130, a memory 140, and a data storage device 150. Data storage device 150 may store machine-executable instructions, data (such as configuration data or other types of application program data), and various programs such as an operating system 152 and one or more application programs 154 for developing and using an intelligent social agent, all of which may be processed by CPU 120. Each computer program may be implemented in a high-level procedural or object-oriented programming language, or in assembly or machine language if desired; and in any case, the language may be a compiled or interpreted language. Data storage device 150 may be any form of non-volatile memory, including by way of example semiconductor memory devices, such as Erasable Programmable Read-Only Memory (EPROM), Electrically Erasable Programmable Read-Only Memory (EEPROM), and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and Compact Disc Read-Only Memory (CD-ROM).

System 100 also may include a communications card or device 160 (e.g., a modem and/or a network adapter) for exchanging data with a network 170 using a communications link 175 (e.g., a telephone line, a wireless network link, a wired network link, or a cable network). Alternatively, a universal system bus (USB) connector may be used to connect system 100 for exchanging data with a network 170. Other examples of system 100 may include a handheld device, a workstation, a server, a device, or some combination of these capable of responding to and executing instructions in a defined manner. Any of the foregoing may be supplemented by, or incorporated in, ASICs (application-specific integrated circuits).

Although FIG. 1 illustrates a PDA and a mobile phone as being peripheral with respect to system 100, in some implementations, the functionality of the system 100 may be directly integrated into the PDA or mobile phone.

FIG. 2 shows an exemplary implementation of intelligent social agent 200 for a computing device including a PDA 210, a stylus 212, and a visual representation of a intelligent social agent 220. Although FIG. 2 shows an intelligent social agent as an animated talking head style character, an intelligent social agent is not limited to such an appearance and may be represented as, for example, a cartoon head, an animal, an image captured from a video or still image, a graphical object, or as a voice only. The user may select the parameters that define the appearance of the social agent. The PDA may be, for example, an iPAQ™ Pocket PC available from COMPAQ.

An intelligent social agent 200 is an animated computer interface agent with social intelligence that has been developed for a given application or device or a target user population. The social intelligence of the agent comes from the ability of the agent to be appealing, affective, adaptive, and appropriate when interacting with the user. Creating the visual appearance, voice, and personality of an intelligent social agent that is based on the personal and professional characteristics of the target user population may help the intelligent social agent be appealing to the target users. Programming an intelligent social agent to manifest affect through facial, vocal and linguistic expressions may help the intelligent social agent appear affective to the target users. Programming an intelligent social agent to modify its behavior for the user, application, and current context may help the intelligent social agent be adaptive and appropriate to the target users. The interaction between the intelligent social agent and the user may result in an improved experience for the user as the agent assists the user in operating a computing device or computing device application program.

FIG. 3 illustrates an architecture of a social intelligence engine 300 that may enable an intelligent social agent to be appealing, affective, adaptive, and appropriate when interacting with a user. The social intelligence engine 300 receives information from and about the user 305 that may include a user profile, and from and about the application program 310. The social intelligence engine 300 produces behaviors and verbal and nonverbal expressions for an intelligent social agent.

The user may interact with the social intelligence engine 300 by speaking, entering text, using a pointing device, or using other types of I/O devices (such as a touch

screen or vision tracking device). Text or speech may be processed by a natural language processing system and received by the social intelligence engine as a text input. Speech will be recognized by speech recognition software and may be processed by a vocal feature analyzer that provides a profile of the affective and physiological states of the user based on characteristics of the user's speech, such as pitch range and breathiness.

Information about the user may be received by the social intelligence engine 300. The social intelligence engine 300 may receive personal characteristics (such as name, age, gender, ethnicity or national origin information, and preferred language) about the user, and professional characteristics about the user (such as occupation, position of employment, and one or more affiliated organizations). The user information received may include a user profile or may be used by the central processor unit 120 to generate and store a user profile.

Non-verbal information received from a vocal feature analyzer or natural language processing system may include vocal cues from the user (such as fundamental pitch and speech rate). A video camera or a vision tracking device may provide non-verbal data about the user's eye focus, head orientation, and other body position information. A physical connection between the user and an I/O device (such as a keyboard, a mouse, a handheld device, or a touch pad) may provide physiological information (such as a measurement of the user's heart rate, blood pressure, respiration, temperature, and skin conductivity). A global positioning system may provide information about the user's geographic location. Other such contextual awareness tools may provide additional information about a user's environment, such as a video camera that provides one or more images of the physical location of the user that may be processed for contextual information, such as whether the user is alone or in a group, inside a building in an office setting, or outside in a park.

The social intelligence engine 300 also may receive information from and about an application program 310 running on the computer 110. The information from the application program 310 is received by the information extractor 320 of the social intelligence engine 300. The information extractor 320 includes a verbal extractor 322, a non-verbal extractor 324, and a user context extractor 326.

The verbal extractor 322 processes verbal data entered by the user. The verbal extractor may receive data from the I/O device used by the user or may receive data after processing (such as text generated by a natural language processing system from the

original input of the user). The verbal extractor 322 captures verbal content, such as commands or data entered by the user for a computing device or an application program (such as those associated with the computer 110). The verbal extractor 322 also parses the verbal content to determine the linguistic style of the user, such as word choice, grammar choice, and syntax style.

The verbal extractor 322 captures verbal content of an application program, including functions and data. For example, functions in an email application program may include viewing an email message, writing an email message, and deleting an email message, and data in an email message may include the words included in a subject line, identification of the sender, time that the message was sent, and words in the email message body. An electronic commerce application program may include functions such as searching for a particular product, creating an order, and checking a product price and data such as product names, product descriptions, product prices, and orders.

The nonverbal extractor 324 processes information about the physiological and affective states of the user. The nonverbal extractor 324 determines the physiological and affective states of the user from 1) physiological data, such as heart rate, blood pressure, blood pulse volume, respiration, temperature, and skin conductivity; 2) from the voice feature data such as speech rate and amplitude; and 3) from the user's verbal content that reveals affective information such as "I am so happy" or "I am tired". Physiological data provide rich cues to induce a user's emotional state. For example, an accelerated heart rate may be associated with fear or anger and a slow heart rate may indicate a relaxed state. Physiological data may be determined using a device that attaches from the computer 110 to a user's finger and is capable of detecting the heart rate, respiration rate, and blood pressure of the user. The nonverbal extraction process is described in FIG. 4.

The user context extractor 326 determines the internal context and external context of the user. The user context extractor 326 determines the mode in which the user requests or executes an action (which may be referred to as internal context) based on the user's physiological data and verbal data. For example, the command to show sales figures for a particular period of time may indicate an internal context of urgency when the words are spoken with a faster speech rate, less articulation, and faster heart rate than when the same words are spoken with a normal style for the user. The user context extractor 326 may determine an urgent internal context from the verbal content of the command, such as when the command includes the term "quickly" or "now".

The user context extractor 326 determines the characteristics for the user's environment (which may be referred to as the external context of the user). For example, a global positioning system (integrated within or connected to the computer 110) may determine the geographic location of the user from which the user's local weather conditions, geology, culture, and language may be determined. The noise level in the user's environment may be determined, for instance, through a natural language processing system or vocal feature analyzer stored on the computer 110 that processes audio data detected through a microphone integrated within or connected to the computer 110. By analyzing images from a video camera or vision tracking device, the user context extractor 326 may be able to determine other physical and social environment characteristics, such as whether the user is alone or with others, located in an office setting, or in a park or automobile.

The application context extractor 328 determines information about the application program context. This information may, for example, include the importance of an application program, the urgency associated with a particular action, the level of consequence of a particular action, the level of confidentiality of the application or the data used in the application program, frequency that the user interacts with the application program or a function in the application program, the level of complexity of the application program, whether the application program is for personal use or in an employment setting, whether the application program is used for entertainment, and the level of computing device resources required by the application program.

The information extractor 320 sends the information captured and compiled by the verbal extractor 322, the non-verbal extractor 324, the user context extractor 326, and the application context extractor 328 to the adaptation engine 330. The adaptation engine 330 includes a machine learning module 332, an agent personalization module 334, and a dynamic adaptor module 336.

The machine learning module 332 receives information from the information extractor 320 and also receives personal and professional information about the user. The machine learning module 332 determines a basic profile of the user that includes information about the verbal and non-verbal styles of the user, application program usage patterns, and the internal and external context of the user. For example, a basic profile of a user may include that the user typically starts an email application program, a portal, and a list of items to be accomplished from a personal information management system

from after the computing device is activated, the user typically speaks with correct grammar and accurate wording, the internal context of the user is typically hurried, and the external context of the user has a particular level of noise and number of people. The machine learning module 332 modifies the basic profile of the user during interactions between the user and the intelligent social agent.

The machine learning module 332 compares the received information about the user and application content and context with the basic profile of the user. The machine learning module 332 may make the comparison using decision logic stored on the computer 110. For example, when the machine learning module 332 has received information that the heart rate of the user is 90 beats per minute, the machine learning module 332 compares the received heart rate with the typical heart rate from the basic profile of the user to determine the difference between the typical and received heart rates, and if the heart rate is elevated a certain number of beats per minute or a certain percentage, the machine learning module 332 determines the heart rate of the user is significantly elevated and a corresponding emotional state is evident in the user.

The machine learning module 332 produces a dynamic digest about the user, the application, the context, and the input received from the user. The dynamic digest may list the inputs received by the machine learning module 332, any intermediate values processed (such as the difference between the typical heart rate and current heart rate of the user), and any determinations made (such as the user is angry based on an elevated heart rate and speech change or semantics indicating anger). The machine learning module 332 uses the dynamic digest to update the basic profile of the user. For example, if the dynamic digest indicates that the user has an elevated heart rate, the machine learning module 332 may so indicate in the current physiological profile section of the user's basic profile. The agent personalization module 334 and the dynamic adaptor module 336 may also use the dynamic digest.

The agent personalization module 334 receives the basic profile of the user and the dynamic digest about the user from the machine learning module 332. Alternatively, the agent personalization module 334 may access the basic profile of the user or the dynamic digest about the user from the data storage device 150. The agent personalization module 334 creates a visual appearance and voice for an intelligent social agent (which may be referred to as casting the intelligent social agent) that may be appealing and appropriate for a particular user population and adapts the intelligent social

agent to fit the user and the user's changing circumstances as the intelligent social agent interacts with the user (which may be referred to as personalizing the intelligent social agent).

The dynamic adaptor module 336 receives the adjusted basic profile of the user and the dynamic digest about the user from the machine learning module 332 and information received or compiled by the information extractor 320. The dynamic adaptor module 336 also receives casting and personalization information about the intelligent social agent from the agent personalization module 334.

The dynamic adaptor module 336 determines the actions and behavior of the intelligent social agent. The dynamic adaptor module 336 may use verbal input from the user and the application program context to determine the one or more actions that the intelligent social agent should perform. For example, when the user enters a request to "check my email messages" and the email application program is not activated, the intelligent social agent activates the email application program and initiates the email application function to check email messages. The dynamic adaptor module 336 may use nonverbal information about the user and contextual information about the user and the application program to help ensure that the behaviors and actions of the intelligent social agent are appropriate for the context of the user.

For example, when the machine learning module 332 indicates that the user's internal context is urgent, the dynamic adaptor module 336 may adjust the intelligent social agent so that the agent has a facial expression that looks serious and stops or pauses a non-critical function (such as receiving a large data file from a network) or closing unnecessary application programs (such as a drawing program) to accomplish a requested urgent action as quickly as possible.

When the machine learning module 332 indicates that the user is fatigued, the dynamic adaptor module 336 may adjust the intelligent social agent so that the agent has a relaxed facial expression, speaks more slowly, and uses words with fewer syllables, and sentences with fewer words.

When the machine learning module 332 indicates that the user is happy or energetic, the dynamic adaptor module 336 may adjust the intelligent social agent to have a happy facial expression and speak faster. The dynamic adaptor module 336 may have

the intelligent social agent to suggest additional purchases or upgrades when the user is placing an order using an electronic commerce application program.

When the machine learning module 332 indicates that the user is frustrated, the dynamic adaptor module 336 may adjust the intelligent social agent to have a concerned facial expression and make fewer or only critical suggestions. If the machine learning module 332 indicates that the user is frustrated with the intelligent social agent, the dynamic adaptor module 336 may have the intelligent social agent apologize and explain sensibly what is the problem and how it should be fixed.

The dynamic adaptor module 336 may adjust the intelligent social agent to behave based on the familiarity of the user with the current computer device, application program, or application program function and the complexity of the application program. For example, when the application program is complex and the user is not familiar with the application program (e.g., the user is using an application program for the first time or the user has not used the application program for some predetermined period of time), the dynamic adaptor module 336 may have the intelligent social agent ask the user whether the user would like help, and, if the user so indicates, the intelligent social agent starts a help function for the application program. When the application program is not complex or the user is familiar with the application program, the dynamic adaptor module 336 typically does not have the intelligent social agent offer help to the user.

The verbal generator 340 receives information from the adaptation engine 330 and produces verbal expressions for the intelligent social agent 350. The verbal generator 340 may receive the appropriate verbal expression for the intelligent social agent from the dynamic adaptor module 336. The verbal generator 340 uses information from the machine learning module 332 to produce the specific content and linguistic style for the intelligent social agent 350.

The verbal generator 340 then sends the textual verbal content to an I/O device for the computer device, typically a display device, or a text-to-speech generation program that converts the text to speech and sends the speech to a speech synthesizer.

The affect generator 360 receives information from the adaptation engine 330 and produces the affective expression for the intelligent social agent 350. The affect generator 360 produces facial expressions and vocal expressions for the intelligent social agent 350 based on an indication from the dynamic adaptor module 336 as to what emotion the

intelligent social agent 350 should express. A process for generating affect is described with respect to FIG. 5.

Referring to FIG. 4A, a process 400A controls a processor to extract nonverbal information and determine the affective state of the user. The process 400A is initiated by receiving physiological state data about the user (step 410A). Physiological state data may include autonomic data, such as heart rate, blood pressure, respiration rate, temperature, and skin conductivity. Physiological data may be determined using a device that attaches from the computer 110 to a user's finger or palm and is capable of detecting the heart rate, respiration rate, and blood pressure of the user.

The processor then tentatively determines a hypothesis for the affective state of the user based on the physiological data received through the physiological channel (step 415A). The processor may use predetermined decision logic that correlates particular physiological responses with an affective state. As described above with respect to FIG. 3, an accelerated heart rate may be associated with fear or anger and a slow heart rate may indicate a relaxed state.

The second channel of data received by the processor to determine the user's affective state is the vocal analysis data (step 420A), such as the pitch range, the volume, and the degree of breathiness in the speech of the user. For example, louder and faster speech compared to the user's basic pattern may indicate that a user is happy. Similarly, quieter and slower speech than normal may indicate that a user is sad. The processor then determines a hypothesis for the affective state of the user based on the vocal analysis data received through the vocal feature channel (step 425A).

The third channel of data received by the processor for determining the user's affective state is the user's verbal content that reveals the user's emotions (step 430A). Examples of such verbal content include phrases such as "Wow, this is great" or "What? The file disappeared?". The processor then determines a hypothesis for the affective state of the user based on the verbal content received through the verbal channel (step 435A).

The processor then integrates the affective state hypotheses based on the data from the physiological channel, the vocal feature channel, and the verbal channel, resolves any conflict, and determines a conclusive affective state of the user (step 440A). Conflict resolution may be accomplished through predetermined decision logic. A confidence coefficient is given to the affective state predicted by each of the three channels based on

the inherent predictive power of that channel for that particular emotion and the unambiguity level of the specific diagnosis of the emotional state in occurrence. Then the processor disambiguates by comparing and integrating the confidence coefficients.

Some implementations may receive either physiological data, vocal analysis data, verbal content, or a combination. When only one type of data is received, integration (step 440A) may not be performed. For example, when only physiological data is received, steps 420A-440A are not performed and the processor uses the affective state of the user based on physiological data as the affective state of the user. Similarly, when only vocal analysis data is received, the process is initiated when vocal analysis data is received and steps 410A, 415A, and 430A-445A are not performed. The processor uses the affective state of the user based on vocal analysis data as the affective state of the user.

Similarly, referring to FIG. 4B, a process 400B controls a processor to extract nonverbal information and determine the affective state of the user. The processor receives physiological data about the user (step 410B), vocal analysis data (step 420B), and verbal content that indicates the emotion of the user (step 430B) and determines a hypothesis for the affective state of the user based on each type of data (steps 415B, 425B, and 435B) in parallel. The processor then integrates the affective state hypotheses based on the data from the physiological channel, the vocal feature channel, and the verbal channel, resolves any conflict, and determines a conclusive affective state of the user (step 440B) as described with respect to FIG. 4A.

Referring to FIG. 5, a process 500 controls a processor to adapt an intelligent social agent to the user and the context. The process 500 may help an intelligent social agent to act appropriately based on the user and the application context.

The process 500 is initiated when content and contextual information is received (step 510) by the processor from an input/output device (such as a voice recognition and speech synthesis device, a video camera, or physiological detection device connected to a finger of the user) to the computer 110. The content and contextual information received may be verbal information, nonverbal information, or contextual information received from the user or application program or may be information compiled by an information extractor (as described previously with respect to FIG. 3).

The processor then accesses data storage device 150 to determine the basic user profile for the user with whom the intelligent social agent is interacting (step 515). The

basic user profile includes personal characteristics (such as name, age, gender, ethnicity or national origin information, and preferred language) about the user, professional characteristics about the user (such as occupation, position of employment, and one or more affiliated organizations), and non-verbal information about the user (such as linguistic style and physiological profile information). The basic user profile information may be received during a registration process for a product that hosts an intelligent social agent or by a casting process to create an intelligent social agent for a user and stored on the computing device.

The processor may adjust the context and content information received based on the basic user profile information (step 520). For example, a verbal instruction to "read email messages now" may be received. Typically, a verbal instruction modified with the term "now" may result in a user context mode of "urgent." However, when the basic user profile information indicates that the user typically uses the term "now" as part of an instruction, the user context mode may be changed to "normal".

The processor may adjust the content and context information received by determining the affective state of the user. The affective state of the user may be determined from content and context information (such as physiological data or vocal analysis data).

The processor modifies the intelligent social agent based on the adjusted content and context information (step 525). For example, the processor may modify the linguistic style and speech style of the intelligent social agent to be more similar to the linguistic style and speech style of the user.

The processor then performs essential actions in the application program (step 530). For example, when the user enters a request to "check my email messages" and the email application program is not activated, the intelligent social agent activates the email application program and initiates the email application function to check email messages (as described previously with respect to FIG. 3).

The processor determines the appropriate verbal expression (step 535) and an appropriate emotional expression for the intelligent social agent (step 540) that may include a facial expression.

The processor generates an appropriate verbal expression for the intelligent social agent (step 545). The appropriate verbal expression includes the appropriate verbal

content and appropriate emotional semantics based on the content and contextual information received, the basic user profile information, or a combination of the basic user profile information and the content and contextual information received.

For example, words that have affective connotation may be used to match the appropriate emotion that the agent should express. This may be accomplished by using an electronic lexicon that associates a word with an affective state, such as associating the word "fantastic" with happiness, the word "delay" with frustration, and so on. The processor selects the word from the lexicon that is appropriate for the user and the context. Similarly, the processor may increase the number of words used in a verbal expression when the affective state of the user is happy or may decrease the number of words used or use words with fewer syllables if the affective state of the user is sad.

The processor may send the verbal expression text to an I/O device for the computer device, typically a display device. The processor may convert the verbal expression text to speech and output the speech. This may be accomplished using a text-to-speech conversion program and a speech synthesizer.

In the meantime, the processor generates an appropriate affect for the facial expression of the intelligent social agent (step 550). Otherwise, a default facial expression may be selected. A default facial expression may be determined by the application, the role of the agent, and the target user population. In general, an intelligent social agent by default may be slightly friendly, smiling, and pleasant.

Facial emotional expressions may be accomplished by modifying portions of the face of the intelligent social agent to show affect. For example, surprise may be indicated by showing the eyebrows raised (e.g., curved and high), skin below brow stretched horizontally, wrinkles across forehead, eyelids opened, and the white of the eye is visible, jaw open without tension or stretching of the mouth.

Fear may be indicated by showing the eyebrows raised and drawn together, forehead wrinkles drawn to the center of the forehead, upper eyelid is raised and lower eyelid is drawn up, mouth open, and lips slightly tense or stretched and drawn back. Disgust may be indicated by showing upper lip is raised, lower lip is raised and pushed up to upper lip or lower lip is lowered, nose is wrinkled, cheeks are raised, lines appear below the lower lid, lid is pushed up but not tense, and brows are lowered. Anger may be indicated by eyebrows lowered and drawn together, vertical lines between eyebrows,

lower lid is tensed, upper lid is tense, eyes have a hard stare, and eyes have a bulging appearance, lips are either pressed firmly together or tensed in a square shape, nostrils may be dilated. Happiness may be indicated by the corners of the lips being drawn back and up, a wrinkle is shown from the nose to the outer edge beyond the lip corners, cheeks are raised, lower eyelid shows wrinkles below it, lower eyelid may be raised but not tense, and crow's-feet wrinkles go outward from the outer corners of the eyes. Sadness may be indicated by drawing the inner corners of eyebrows up, triangulating the skin below the eyebrow, the inner corner of the upper lid and upper corner is raised, and corners of the lips are drawn or lip is trembling.

The processor then generates the appropriate affect for the verbal expression of the intelligent social agent (step 555). This may be accomplished by modifying the speech style from the baseline style of speech for the intelligent social agent. Speech style may include speech rate, pitch average, pitch range, intensity, voice quality, pitch changes, and level of articulation. For example, a vocal expression may indicate fear when the speech rate is much faster, the pitch average is very much higher, the pitch range is much wider, the intensity of speech normal, the voice quality irregular, the pitch change is normal, and the articulation precise. Speech style modifications that may connote a particular affective state are set forth in the table below and are further described in Murray, I. R., & Arnott, J. L. (1993), Toward the simulation of emotion in synthetic speech: A review of the literature on human vocal emotion, Journal of Acoustical Society of America, 93, 1097-1108.

	Fear	Anger	Sadness	Happiness	Disgust
Speech Rate	Much Faster	Slightly Faster	Slightly Slower	Faster Or Slower	Very Much Slower
Pitch Average	Very Much Higher	Very Much Higher	Slightly Lower	Much Higher	Very Much Lower
Pitch Range	Much Wider	Much Wider	Slightly Narrower	Much Wider	Slightly Wider
Intensity	Normal	Higher	Lower	Higher	Lower
Voice Quality	Irregular Voicing	Breathy Chest Tone	Resonant	Breathy Blaring	Grumbled Chest Tone
Pitch Changes	Normal	Abrupt On Stressed Syllables	Downward Inflections	Smooth Upward Inflections	Wide Downward Terminal Inflections
Articulation	Precise	Tense	Slurring	Normal	Normal

Referring to FIG. 6, a process 600 controls a processor to create an intelligent social agent for a target user population. This process (which may be referred to as casting an intelligent social agent) may produce an intelligent social agent whose appearance and voice are appealing and appropriate for the target users.

The process 600 begins with the processor accessing user information stored in the basic user profile (step 605). The user information stored within the basic user profile may include personal characteristics (such as name, age, gender, ethnicity or national origin information, and preferred language) about the user and professional characteristics about the user (such as occupation, position of employment, and one or more affiliated organizations).

The processor receives information about the role of the intelligent social agent for one or more particular application programs (step 610). For example, the intelligent social agent may be used as a help agent to provide functional help information about an

application program or may be used as an entertainment player in a game application program.

The processor then applies an appeal rule to further analyze the basic user profile and to select a visual appearance for the intelligent social agent that may be appealing to the target user population (step 620). The processor may apply decision logic that associates a particular visual appearance for an intelligent social agent with particular age groups, occupations, gender, or ethnic or cultural groups. For example, decision logic may be based on similarity-attraction (that is, matching the ages, personalities, and ethnical identities of the intelligent social agent and the user). A professional-looking talking-head may be more appropriate for an executive user (such as a chief executive officer or a chief financial officer), and a talking-head with an ultra-modern hair style may be more appealing to an artist.

The processor applies an appropriateness rule to further analyze the basic user profile and to modify the casting of the intelligent social agent (step 630). For example, a male intelligent social agent may be more suitable for technical subject matter, and a female intelligent social agent may be more appropriate for fashion and cosmetics subject matter.

The processor then presents the visual appearance for the intelligent social agent to the user (step 640). Some implementations may allow the user to modify attributes (such as the hair color, eye color, and skin color) of the intelligent social agent or select from among several intelligent social agents with different visual appearances. Some implementations also may allow a user to import a graphical drawing or image to use as the visual appearance for the intelligent social agent.

The processor applies the appeal rule to the stored basic user profile (step 650) and the appropriateness rule to the stored basic user profile to select a voice for the intelligent social agent (step 660). The voice should be appealing to the user and be appropriate for the gender represented by the visual intelligent social agent (e.g., an intelligent social agent with a male visual appearance has a male voice and an intelligent social agent with a female visual appearance has a female voice). The processor may match the user's speech style characteristics (such as speech rate, pitch average, pitch range, and articulation) as appropriate for the voice of the intelligent social agent.

The processor presents the voice choice for the intelligent social agent (step 670). Some implementations may allow the user to modify the speech characteristics for the intelligent social agent.

The processor then associates the intelligent social agent with the particular user (step 680). For example, the processor may associate an intelligent social agent identifier with the intelligent social agent, store the intelligent social agent identifier and characteristics of the intelligent social agent in the data storage device 150 of the computer 110 and store the intelligent social agent identifier with the basic user profile. Some implementations may cast one or more intelligent social agents to be appropriate for a group of users that have similar personal or professional characteristics.

Referring to FIG. 7, an implementation of an intelligent social agent is an intelligent personal assistant. The intelligent personal assistant interacts with a user of the computing device such as computing device 210 to assist the user in operating the computing device 210 and using application programs. The intelligent personal assistant assists the user of the computing device to manage personal information, operate the computing device 210 or one or more application programs running on the computing device, and use the computing device for entertainment.

The intelligent personal assistant may operate on a mobile computing device, such as a PDA, laptop, or mobile phone, or a hybrid device including the functions associated with a PDA, laptop, or mobile phone. When an intelligent personal assistant operates on a mobile computing device, the intelligent personal assistant may be referred to as an intelligent mobile personal assistant. The intelligent personal assistant also may operate on a stationary computing device, such as a desktop personal computer or workstation, and may operate on a system of networked computing devices, as described with respect to FIG. 1.

FIG. 7 illustrates one implementation of an architecture 700 for an intelligent personal assistant 730. Application program 710, including a personal information management application program 715, one or more entertainment application programs 720, and/or one or more application programs to operate the computing device 725, may run on a computing device, as described with respect to FIG. 1.

The intelligent personal assistant 730 uses the social intelligence engine 735 to interact with a user 740 and the application programs 710. Social intelligence engine 735

is substantially similar to social intelligence engine 300 of FIG. 3. The information extractor 745 of the intelligent personal assistant 730 receives information from and about the application programs 710 and information from and about the user 740, in a similar manner as described with respect to FIG. 3.

The intelligent personal assistant 730 processes the extracted information using an adaptation engine 750 and then generates one or more responses (including verbal content and facial expressions) to interact with the user 740 using by the verbal generator 755 and the affect generator 760, in a similar manner as described with respect to FIG. 3. The intelligent personal assistant 730 also may produce one or more responses to operate one or more of the application programs 710 running on the computing device 210, as described with respect to FIGS. 2-3 and FIGS. 8-10. The responses produced may enable the intelligent personal assistant 730 to appear appealing, affective, adaptive, and appropriate when interacting with the user 740. The user 740 also interacts with one or more of the applications programs 710.

FIG. 8 illustrates an architecture 800 for implementing an intelligent personal assistant that helps a user to manage personal information. The intelligent personal assistant 810 may assist the user 815 as an assistant that works across all personal information management application program functions. For a business user using a mobile computing device, the intelligent personal assistant 810 may be able to function as an administrative assistant in helping the user manage appointments, email messages, and contact lists. As similarly described with respect to FIGS. 3 and 7, the intelligent personal assistant 810 interacts with the user 815 and the personal information management application program 820 using the social intelligence engine 825, that also includes an information extractor 830, an adaptation engine 835, a verbal generator 840, and an affect generator 845.

The personal information management application program 820 (which also may be referred to as a PIM) includes email functions 850, calendar functions 855, contact management functions 860, and task list functions 865 (which also may be referred to as a "to do" list). The personal information management application program may be, for example, a version of Microsoft® Outlook®, such as Pocket Outlook®, by Microsoft Corporation, that operates on a PDA.

The intelligent personal assistant 810 may interact with the user 815 concerning email functions 850. For example, the intelligent personal assistant 810 may report the

status of the user's email account, such as the number of unread messages or the number of unread messages having an urgent status, at the beginning of a work day or when the user requests such an action. The intelligent personal assistant 810 may communicate with the user 815 with a more intense affect about unread messages having an urgent status, or when the number of unread messages is higher than typical for the user 815 (based on intelligent and/or statistical monitoring of typical e-mail patterns). The intelligent personal assistant 810 may notify the user 815 of recently received messages and may communicate with a more intense affect when a recently received message has an urgent status. The intelligent personal assistant 810 may help the user manage messages, such as suggesting messages be deleted or archived based on the user's typical message deletion or archival patterns or when the storage space for messages is reaching or exceeding its limit, or suggesting messages be forwarded to particular users or groups of users based on the user's typical message forwarding patterns.

The intelligent personal assistant 810 may help the user 815 manage the user's calendar 850. For example, the intelligent personal assistant 810 can report to the user his/her upcoming appointments for the day in the morning or at any time the user desires. The intelligent personal assistant 810 may remind the user 815 of upcoming appointments at a time desired by the user and also decide how far the location of the appointment is from the user's current location. If the user is late or seems late for an appointment, the intelligent personal assistant 810 will accordingly remind him/her in an urgent manner such as speaking a little louder and appearing a little concerned. For example, when a user does not need to travel to an upcoming appointment, such as a business meeting at the office in which the user is located, and the appointment is a regular one in terms of significance and urgency, the intelligent personal assistant 810 may remind the user 815 of the appointment in a neutral affect with regular voice tone and facial expression. As the time approaches for an upcoming appointment that requires the user to leave the premises to travel to the appointment, the intelligent personal assistant 810 may remind the user 815 of the appointment in a voice with a higher volume and with more urgent affect.

The intelligent personal assistant 810 may help the user 815 enter an appointment in the calendar. For example, the user 815 may verbally describe the appointment using general or relative terms. The intelligent personal assistant 810 transforms the general description of the appointment into information that can be entered into the calendar

application program 860 and sends a command to enter the information into the calendar. For example, the user may say "I have an appointment with Dr. Brown next Thursday at 1." Using the social intelligence engine 825, the intelligent personal assistant 810 may generate the appropriate commands to the calendar application program 860 to enter an appointment in the user's calendar. For example, the intelligent personal assistant 810 may understand that Dr. Brown is the user's physician (possibly by performing a search within the contacts database 860) and that the user will have to travel to the physician's office. The intelligent personal assistant 810 also may look up the address using contact information in the contact management application program 860, and may use a mapping application program to estimate the time required to travel from the user's office address to the doctor's office, and determine the date that corresponds to "next Thursday". The intelligent personal assistant 810 then sends commands to the calendar application program to enter the appointment at 1:00 pm on the appropriate date and to generate a reminder message for a sufficient time before the appointment that allows the user time to travel to the doctor's office.

The intelligent personal assistant 810 also may help the user 815 manage the user's contacts 860. For example, the intelligent personal assistant 810 may enter information for a new contact that the user 815 has spoken to the intelligent personal assistant 810. For example, the user 815 may say "My new doctor is Dr. Brown in Oakdale." The intelligent personal assistant 810 looks up the full name, address, and telephone number of Dr. Brown by using a web site of the user's insurance company that lists the doctors that accept payment from the user's insurance carrier. The intelligent personal assistant 810 then sends commands to the contact application program 860 to enter the contact information. The intelligent personal assistant 810 may help organize the contact list by entering new contacts that cross-reference contacts entered by the user 815, such as entering the contact information for Dr. Brown also under "Physician".

The intelligent personal assistant 810 may help the user 815 manage the user's task list application 865. For example, the intelligent personal assistant 810 may enter information for a new task, read the task list to the user when the user may not be able to view the text display of the computing device, such as when the user is driving an automobile, and remind the user of tasks that are due in the near future. The intelligent personal assistant 810 may remind the user 815 of a task with a higher importance rating that is due in the near future using a voice with a higher volume and more urgent affect.

Some personal information management application programs may include voice mail and phone call functions (not shown). The intelligent personal assistant 810 may help manage the voice mail messages received by the user 815, such as by playing messages, saving messages, or reporting the status of messages (e.g., how many new messages have been received). The intelligent personal assistant 810 may remind the user 815 that a new message has not been played using a voice with higher volume and more urgent affect when more time has passed than typical for the user to check his voice mail messages.

The intelligent personal assistant 810 may help the user manage the user's phone calls. The intelligent personal assistant 810 may act as if the intelligent personal assistant 810 is a virtual secretary for the user 815 by receiving and selectively processing received phone calls. For example, when the user is busy and does not want to receive phone calls, the intelligent personal assistant 810 may not notify the user about an incoming call. The intelligent personal assistant 810 may selectively notify the user about incoming phone calls based on a priority scheme in which the user specifies a list of people from whom the user will speak with if a phone call is received, or will speak with if a phone call is received under particular conditions specified by the user, for example, even when the user is busy.

The intelligent personal assistant 810 also may be able to organize and present news to the user 815. The intelligent personal assistant 810 may use news sources and categories of news based on the user's typical patterns. Additionally or alternatively, the user 815 may select news sources and categories for the intelligent personal assistant 810 to use.

The user 815 may select the modality through which the intelligent personal assistant 810 produces output, such as whether the intelligent personal assistant produces only speech output, only text output on a display, or both speech and text output. The user 815 may indicate by using speech input or clicking a mute button that the intelligent personal assistant 810 is only to use text output.

FIG. 9 illustrates an architecture 900 of an intelligent personal assistant helping a user to operate applications in a computing device. The intelligent personal assistant 910 may assist the user 915 across various application programs or functions. As described with respect to FIGS. 3 and 7, intelligent personal assistant 910 interacts with the user 915 and the application programs 920 in a computing device, including basic functions

relating to the device itself and applications running on the device such as enterprise applications. The intelligent personal assistant 910 similarly uses the social intelligence engine 945 including an information extractor 950, an adaptation engine 955, a verbal generator 960, and an affect generator 965.

Some example of basic functions relating to a computing device itself are checking battery status 925, opening or closing an application program 930, 935, and synchronizing data 940, among many other functions. The intelligent personal assistant 910 may interact with the user 915 concerning the status of the battery 925 in the computing device. For example, the intelligent personal assistant 910 may report that the battery is running low when the battery is running lower than ten percent (or other user defined threshold) of the battery's capacity. The intelligent personal assistant 910 may make suggestions, such as dimming the screen or closing some applications, and send the commands to accomplish those functions when the user 915 accepts the suggestions.

The intelligent personal assistant 910 may interact with the user 915 to switch applications by using an open application program 930 function and a close application program 935 function. For example, the intelligent personal assistant 910 may close a particular spreadsheet file and open a particular word processing document when the user indicates that a particular word processing document should be opened because the user typically closes the particular spreadsheet file when opening the particular word processing document.

The intelligent personal assistant 910 may interact with the user to synchronize data 940 between two computing devices. For example, the intelligent personal assistant 910 may send commands to copy personal management information from a portable computing device, such as a PDA, to a desktop computing device. The user 915 may request that the devices be synchronized without specifying what information is to be synchronized. The intelligent personal assistant 910 may synchronize appropriate personal management information based on the user's typical pattern of keeping contact and task list information synchronized on the desktop but not copying appointment information that resides only in the PDA.

Beyond the basic functions for operating a computing device itself, the intelligent personal assistant 910 can help a user operate a wide range of applications running on the computing device. Examples of enterprise applications for an intelligent personal

assistant 901 are business reports, budget management, project management, manufacturing monitoring, inventory control, purchase, sales, learning and training.

On mobile enterprise portals, an intelligent personal assistant 910 can provide tremendous assistance to the user 915 by prioritizing and pushing out important and urgent information. The context-defining method for applications in the intelligent social agent architecture guides the intelligent personal assistant 910 in this matter. For example, the intelligent personal assistant 910 can push out the alerts of sales drop in top priority either by displaying it on the screen or saying it to the user. The intelligent personal assistant 910 adapts its verbal style to make it straightforward and concise, speaks a little faster, and appears concerned such as with slight frowning in the case of sales-drop alert. The intelligent personal assistant 910 can present the business reports such as sales reports, acquisition reports and project status such as a production timeline to the user through speech or graphical display. The intelligent personal assistant 910 would push out or mark any emergent or serious problems in these matters. The intelligent personal assistant 910 may present approval requests to the managers in a simple and straightforward method so that the user can immediately grasp the most critical information instead of taking numerous steps to dig out the information by him/herself.

FIG. 10 illustrates an architecture 1000 of an intelligent personal assistant helping a user to use a computing device for entertainment. Using the intelligent personal assistant for entertainment may increase the user's willingness to interact with the intelligent personal assistant for non-entertainment applications. The intelligent personal assistant 1010 may assist the user 1015 across various entertainment application programs. As described with respect to FIGS. 3 and 7, intelligent personal assistant 1010 interacts with the user 1015 and the computing device entertainment programs 1020, such as by participating in games, providing narrative entertainment, and performing as an entertainer. The intelligent personal assistant 1010 similarly uses the social intelligence engine 1030, including an information extractor 1035, an adaptation engine 1040, a verbal generator 1045, and an affect generator 1050.

The intelligent personal assistant 1010 may interact with the user 1015 by participating in computing device-based games. For example, the intelligent personal assistant 1010 may act as a participant when playing a game with the user, for example, a card game or other computing device-based game, such as an animated car racing game

or chess game. The intelligent personal assistant 1010 may interact with the user in a more exaggerated manner when helping the user 1015 use the computing device for entertainment than when helping the user with non-entertainment application programs. For example, the intelligent personal assistant 1010 may speak louder, use colloquial expressions, laugh, move its eyebrows up and down often, and open its eyes widely when playing a game with the user. When the user wins a competitive game against the intelligent personal assistant 1010, the intelligent personal assistant may praise the user 1015, or when the user loses to the intelligent personal assistant, the intelligent personal assistant may console the user, compliment the user, or discuss how to improve.

The intelligent personal assistant 1010 may act as an entertainment companion by providing narrative entertainment, such as by reading stories or re-narrating sporting events to the user while the user is driving an automobile or telling jokes to the user when the user is bored or tired. The intelligent personal assistant 1010 may perform as an entertainer, such as by appearing to sing music lyrics (which may be referred to as "lip-synching") or, when an intelligent personal assistant 1010 is represented as a full-bodied agent, dancing to music to entertain.

Implementations may include a method or process, an apparatus or system, or computer software on a computer medium. It will be understood that various modifications may be made without departing from the spirit and scope of the following claims. For example, advantageous results still could be achieved if steps of the disclosed techniques were performed in a different order and/or if components in the disclosed systems were combined in a different manner and/or replaced or supplemented by other components.

WHAT IS CLAIMED IS:

1. A computer-implemented method for implementing an intelligent personal assistant comprising:

receiving an input associated with a user and an input associated with an application program;

accessing a user profile associated with the user;

extracting context information from the received input; and

processing the context information and the user profile to produce an adaptive response by the intelligent personal assistant.

2. The method of claim 1 wherein:

the application program is a personal information management application program, and

the adaptive response by the intelligent personal assistant is associated with the personal information management application program.

3. The method of claim 1 wherein:

the application program is an application program to operate a computing device, and

the adaptive response by the intelligent personal assistant is associated with operating the computing device.

4. The method of claim 1 wherein:

the application program is an entertainment application program, and

the adaptive response by the intelligent personal assistant is associated with the entertainment application program.

5. The method of claim 4 wherein:

the entertainment application program is a game, and

the adaptive response by the intelligent personal assistant is associated with the game.

6. A computer-readable medium or propagated signal having embodied thereon a computer program configured to implement an intelligent personal assistant, the medium comprising a code segment configured to:

receive an input associated with a user and an input associated with an application program;

access a user profile associated with the user;

extract context information from the received input; and

process the context information and the user profile to produce an adaptive response by the intelligent personal assistant.

7. The medium of claim 6 wherein:

the application program is a personal information management application program, and

the adaptive response by the intelligent personal assistant is associated with the personal information management application program.

8. The medium of claim 6 wherein:

the application program is an application program to operate a computing device, and

the adaptive response by the intelligent personal assistant is associated with operating the computing device.

9. The medium of claim 6 wherein:

the application program is an entertainment application program, and

the adaptive response by the intelligent personal assistant is associated with the entertainment application program.

10. The medium of claim 9 wherein:

the entertainment application program is a game, and

the adaptive response by the intelligent personal assistant is associated with the game.

11. A system for implementing a intelligent personal assistant, the system comprising a processor connected to a storage device and one or more input/output devices, wherein the processor is configured to:

receive an input associated with a user and an input associated with an application program;

access a user profile associated with the user;

extract context information from the received input; and

process the context information and the user profile to produce an adaptive response by the intelligent personal assistant.

12. The system of claim 11 wherein:

the application program is a personal information management application program, and

the adaptive response by the intelligent personal assistant is associated with the personal information management application program.

13. The system of claim 11 wherein:

the application program is an application program to operate a computing device, and

the adaptive response by the intelligent personal assistant is associated with operating the computing device.

14. The system of claim 11 wherein:

the application program is an entertainment application program, and

the adaptive response by the intelligent personal assistant is associated with the entertainment application program.

15. The system of claim 14 wherein:

the entertainment application program is a game, and

the adaptive response by the intelligent personal assistant is associated with the game.

16. An apparatus for implementing an intelligent social agent, the apparatus comprising:

an information extractor configured to:

access a user profile associated with the user,

receive an input associated with a user, and

extract context information from the received input;

an adaptation engine configured to:

receive the context information and the user profile from the information extractor, and process the context information and the user profile to produce an adaptive output; and

an output generator configured to:

receive the adaptive output from the adaptation engine, and represent the adaptive output in the intelligent social agent.

17. The apparatus of claim 16 wherein the input is physiological data associated with the user and the information extractor is configured to receive the physiological data.

18. The apparatus of claim 16 wherein the input is application program information associated with the user and the information extractor is configured to receive application program information associated with the user.

19. The apparatus of claim 16 wherein the information extractor is further configured to extract information about an affective state of the user from the received input.

20. The apparatus of claim 19 wherein the information extractor is configured to extract information about an affective state of the user based on physiological information associated with the user.

21. The apparatus of claim 19 wherein the information extractor configured to extract information about an affective state of the user is configured to extract information about an affective state of the user based on vocal analysis information associated with the user by extracting verbal content and analyzing speech characteristics of the user.

22. The apparatus of claim 19 wherein the information extractor configured to extract information about an affective state of the user is configured to extract information about an affective state of the user based on verbal information from the received input.

23. The apparatus of claim 16 wherein the information extractor configured to extract context information is configured to extract a geographical position of the user by using a global positioning system.

24. The apparatus of claim 23 wherein the information extractor configured to extract context information is configured to extract information based on the geographical position of the user.

25. The apparatus of claim 16 wherein the information extractor configured to extract context information is configured to extract information about the application content associated with the user.

26. The apparatus of claim 16 wherein the information extractor configured to extract context information is configured to extract information about a linguistic style of the user from the received input.

27. The apparatus of claim 16 wherein:

the output generator is a verbal generator;

the adaptation engine configured to produce an adaptive output is configured to produce a verbal expression; and

the verbal generator produces the verbal expression in the intelligent social agent.

28. The apparatus of claim 16 wherein:

the generator is an affect generator;

the adaptation engine configured to produce an adaptive output is configured to produce a facial expression; and

the affect generator represents the facial expression in the intelligent social agent.

29. The apparatus of claim 16 wherein the output generator is a multi-modal output generator that represents the adaptive output in the intelligent social agent using at least one of a first mode and a second mode.

30. The apparatus of claim 29 wherein:

the first mode is a verbal mode;

the second mode is an affect mode;

the adaptive engine configured to produce an adaptive output is configured to:

produce a facial expression, and

produce an verbal expression; and

the multi-modal output generator represents the facial expression and the verbal expression in the intelligent social agent.

31. The apparatus of claim 16 wherein:

the adaptation engine is further configured to produce an emotional expression to be represented by the intelligent social agent; and

the output generator is configured to represent the emotional expression in the intelligent social agent.

32. A mobile device for implementing an intelligent social agent that interacts with a user, the mobile device comprising:

a processor connected to a memory and one or more input/output devices;

a social intelligence engine configured to interact with the processor, the social intelligence engine including:

an information extractor configured to:

access a user profile associated with the user,

receive an input associated with a user, and

extract context information from the received input;

an adaptation engine configured to:

receive the context information and the user profile from the information extractor, and process the context information and the user profile to produce an adaptive output; and

an output generator configured to:

receive the adaptive output from the adaptation engine, and represent the adaptive output in the intelligent social agent.

33. The mobile device of claim 32 wherein the input is physiological data associated with the user and the information extractor is configured to receive the physiological data.

34. The mobile device of claim 32 wherein the input is application program information associated with the user and the information extractor is configured to receive the application program information.

35. The mobile device of claim 32 wherein the information extractor is further configured to extract information about an affective state of the user from the received input.

36. The mobile device of claim 35 wherein the information extractor is configured to extract information about an affective state of the user based on physiological information associated with the user.

37. The mobile device of claim 35 wherein the information extractor configured to extract information about an affective state of the user is configured to extract information about an affective state of the user based on vocal analysis information associated with the user by extracting verbal content and analyzing speech characteristics of the user from the received input.

38. The mobile device of claim 35 wherein the information extractor configured to extract information about an affective state of the user is configured to extract information about an affective state of the user based on verbal information from the received input.

39. The mobile device of claim 32 wherein the information extractor configured to extract context information is configured to extract a geographical position of the user by using a global positioning system.

40. The mobile device of claim 35 wherein the information extractor configured to extract context information is configured to extract information based on the geographical position of the user.

41. The mobile device of claim 32 wherein information extractor configured to extract context information is configured to extract information about the application content associated with the user.

42. The mobile device of claim 32 wherein information extractor configured to extract context information is configured to extract information about a linguistic style of the user from the received input.

43. The mobile device of claim 32 wherein:

the output generator is a verbal generator;

the adaptation engine configured to produce an adaptive output is configured to produce a verbal expression; and

the verbal generator produces the verbal expression in the intelligent social agent.

44. The mobile device of claim 32 wherein:

the generator is an affect generator;

the adaptation engine configured to produce an adaptive output is configured to produce a facial expression; and

the affect generator represents the facial expression in the intelligent social agent.

45. The mobile device of claim 32 wherein the output generator is a multi-modal output generator that represents the adaptive output in the intelligent social agent using at least one of a first mode and a second mode.

46. The mobile device of claim 45 wherein:

the first mode is a verbal mode;

the second mode is an affect mode;

the adaptive engine configured to produce an adaptive output is configured to:

produce a facial expression, and

produce an verbal expression; and

the multi-modal output generator represents the facial expression and the verbal expression in the intelligent social agent.

47. The mobile device of claim 32 wherein:

the adaptation engine is further configured to produce an emotional expression to be represented by the intelligent social agent; and

the output generator is configured to represent the emotional expression in the intelligent social agent.

48. A method for implementing an intelligent social agent, the method comprising:

receiving an input associated with a user;

accessing a user profile associated with the user;

extracting context information from the received input; and

processing the context information and the user profile to produce an adaptive output to be represented by the intelligent social agent.

49. The method of claim 48 wherein the input associated with the user comprises physiological data associated with the user.

50. The method of claim 48 wherein the input associated with the user comprises application program information associated with the user.

51. The method of claim 48 wherein extracting context information comprises extracting information about an affective state of the user.

52. The method of claim 51 wherein extracting information about an affective state of the user is based on physiological information associated with the user.

53. The method of claim 51 wherein extracting information about an affective state of the user is based on vocal analysis information associated with the user.

54. The method of claim 51 wherein extracting information about an affective state of the user is based on verbal information from the user.

55. The method of claim 48 wherein extracting context information comprises extracting a geographical position of the user.

56. The method of claim 55 wherein extracting context information comprises extracting information based on the geographical position of the user.

57. The method of claim 48 wherein extracting context information comprises extracting information about the application content associated with the user.

58. The method of claim 48 wherein extracting context information comprises extracting information about a linguistic style of the user.

59. The method of claim 48 wherein the adaptive output comprises a verbal expression to be represented by the intelligent social agent.

60. The method of claim 48 wherein the adaptive output comprises a facial expression to be represented by the intelligent social agent.

61. The method of claim 48 wherein an adaptive output comprises an emotional expression to be represented by the intelligent social agent.

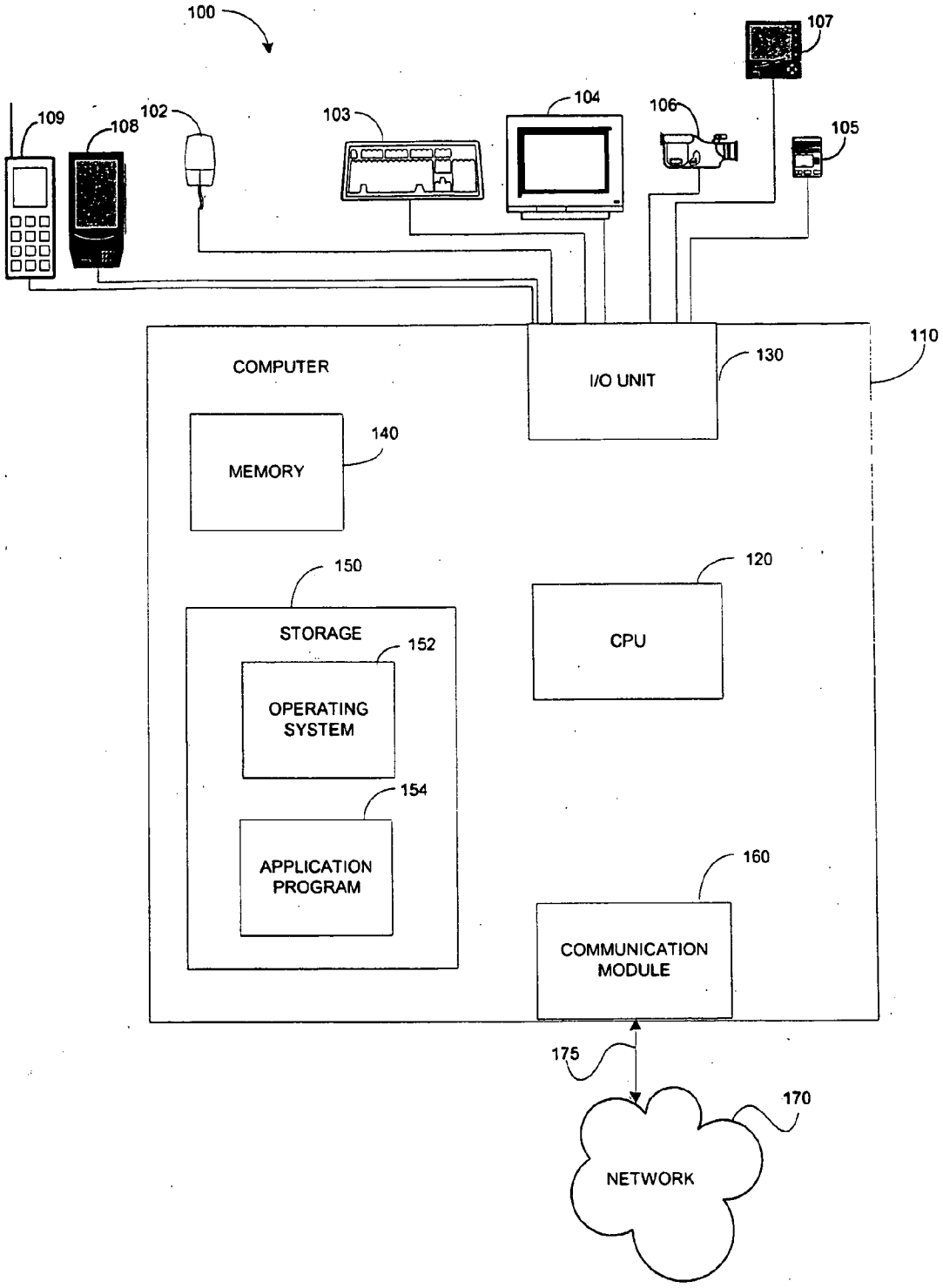

100
107
109
108
102
103
104
106
105
110
COMPUTER
I/O UNIT
130
MEMORY
140
150
STORAGE
152
OPERATING SYSTEM
154
APPLICATION PROGRAM
120
CPU
160
COMMUNICATION MODULE
175
170
NETWORK

FIG. 1

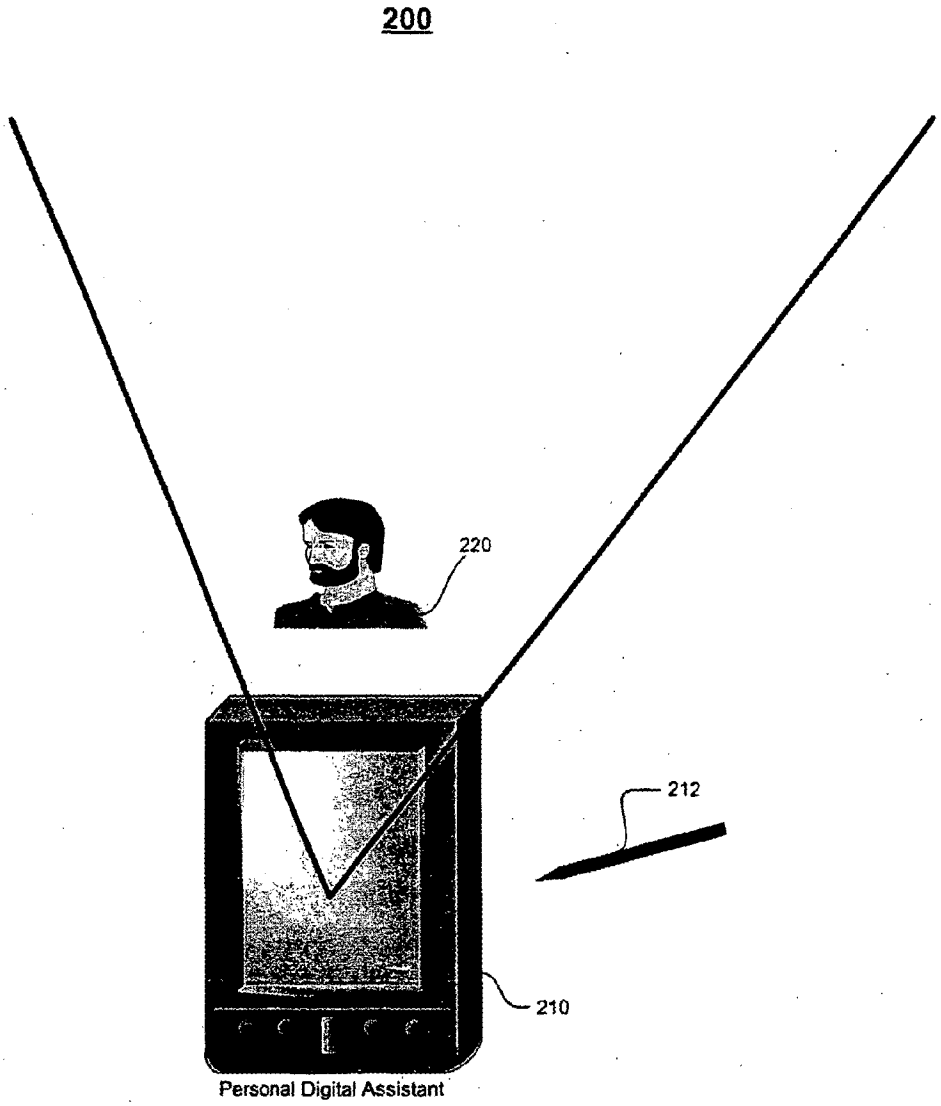


FIG. 2

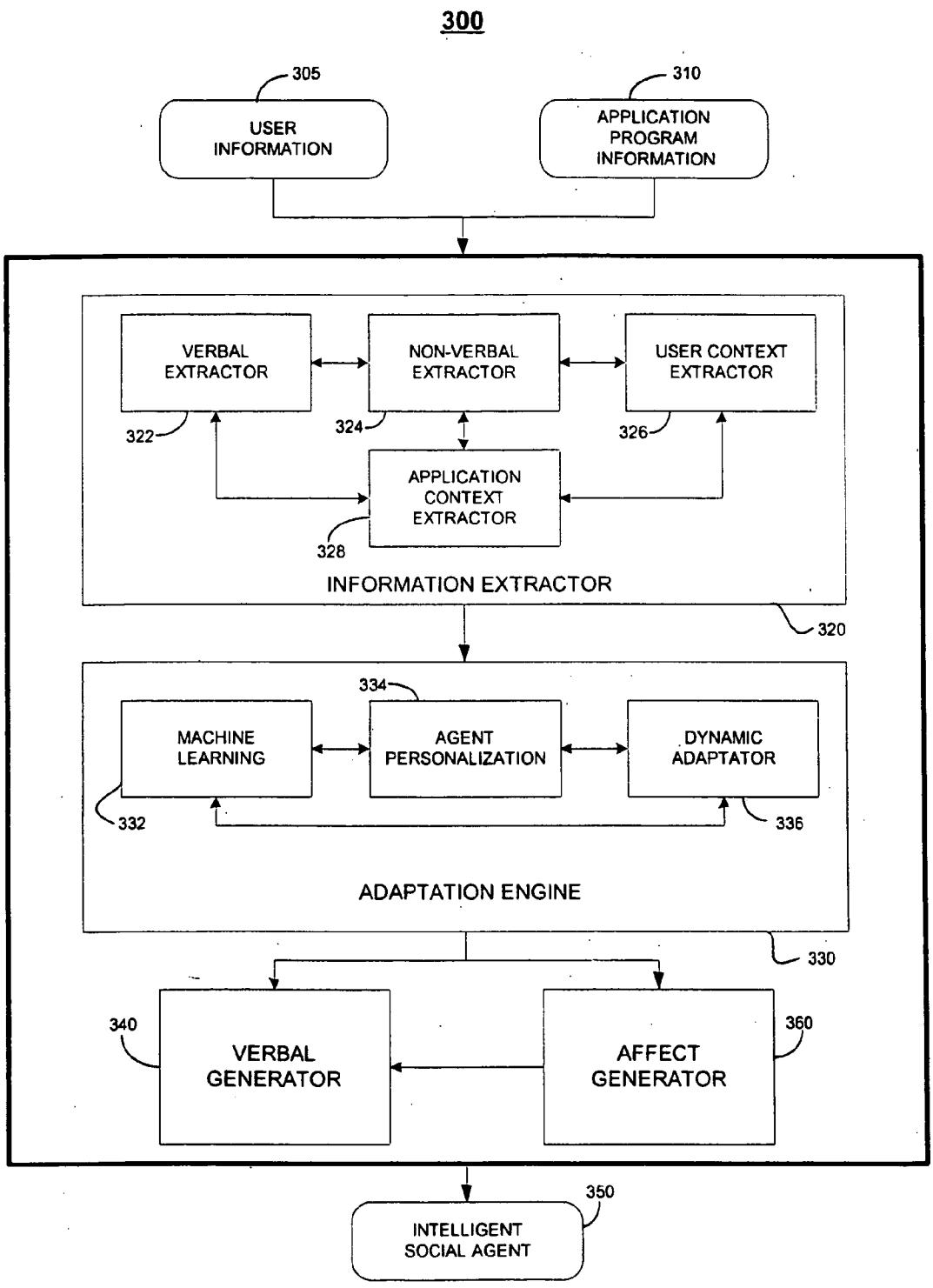

300
305
USER INFORMATION
310
APPLICATION PROGRAM INFORMATION
VERBAL EXTRACTOR
NON-VERBAL EXTRACTOR
USER CONTEXT EXTRACTOR
322
324
326
APPLICATION CONTEXT EXTRACTOR
328
INFORMATION EXTRACTOR
320
334
MACHINE LEARNING
AGENT PERSONALIZATION
DYNAMIC ADAPTATOR
332
336
ADAPTATION ENGINE
330
340
VERBAL GENERATOR
AFFECT GENERATOR
360
350
INTELLIGENT SOCIAL AGENT

FIG. 3

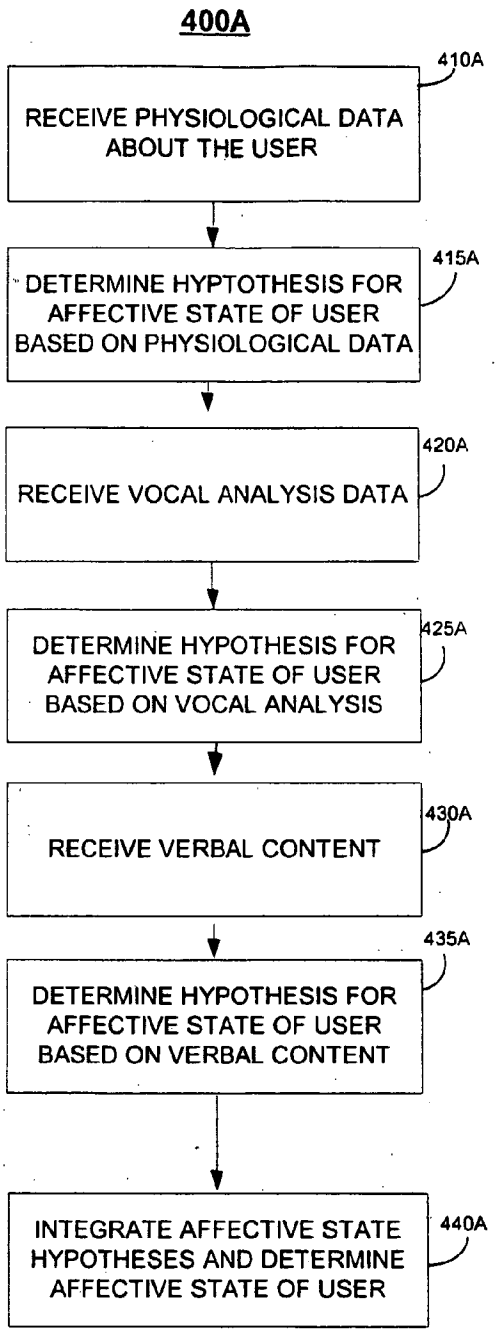

400A
410A
RECEIVE PHYSIOLOGICAL DATA ABOUT THE USER
415A
DETERMINE HYPTOTHESIS FOR AFFECTIVE STATE OF USER BASED ON PHYSIOLOGICAL DATA
420A
RECEIVE VOCAL ANALYSIS DATA
425A
DETERMINE HYPOTHESIS FOR AFFECTIVE STATE OF USER BASED ON VOCAL ANALYSIS
430A
RECEIVE VERBAL CONTENT
435A
DETERMINE HYPOTHESIS FOR AFFECTIVE STATE OF USER BASED ON VERBAL CONTENT
440A
INTEGRATE AFFECTIVE STATE HYPOTHESES AND DETERMINE AFFECTIVE STATE OF USER

FIG. 4A

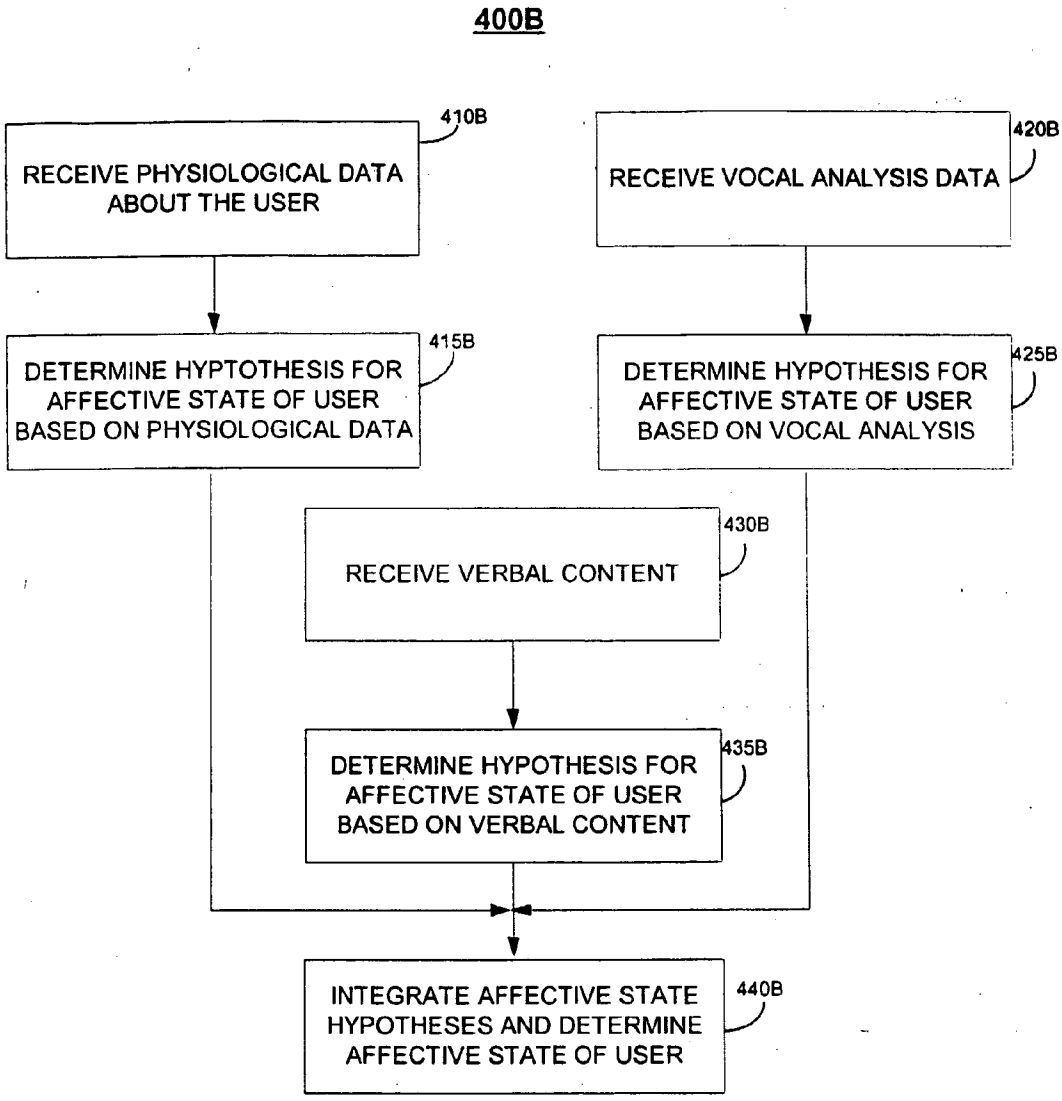

400B
410B
RECEIVE PHYSIOLOGICAL DATA ABOUT THE USER
420B
RECEIVE VOCAL ANALYSIS DATA
415B
DETERMINE HYPTOTHESIS FOR AFFECTIVE STATE OF USER BASED ON PHYSIOLOGICAL DATA
425B
DETERMINE HYPOTHESIS FOR AFFECTIVE STATE OF USER BASED ON VOCAL ANALYSIS
430B
RECEIVE VERBAL CONTENT
435B
DETERMINE HYPOTHESIS FOR AFFECTIVE STATE OF USER BASED ON VERBAL CONTENT
440B
INTEGRATE AFFECTIVE STATE HYPOTHESES AND DETERMINE AFFECTIVE STATE OF USER

FIG. 4B

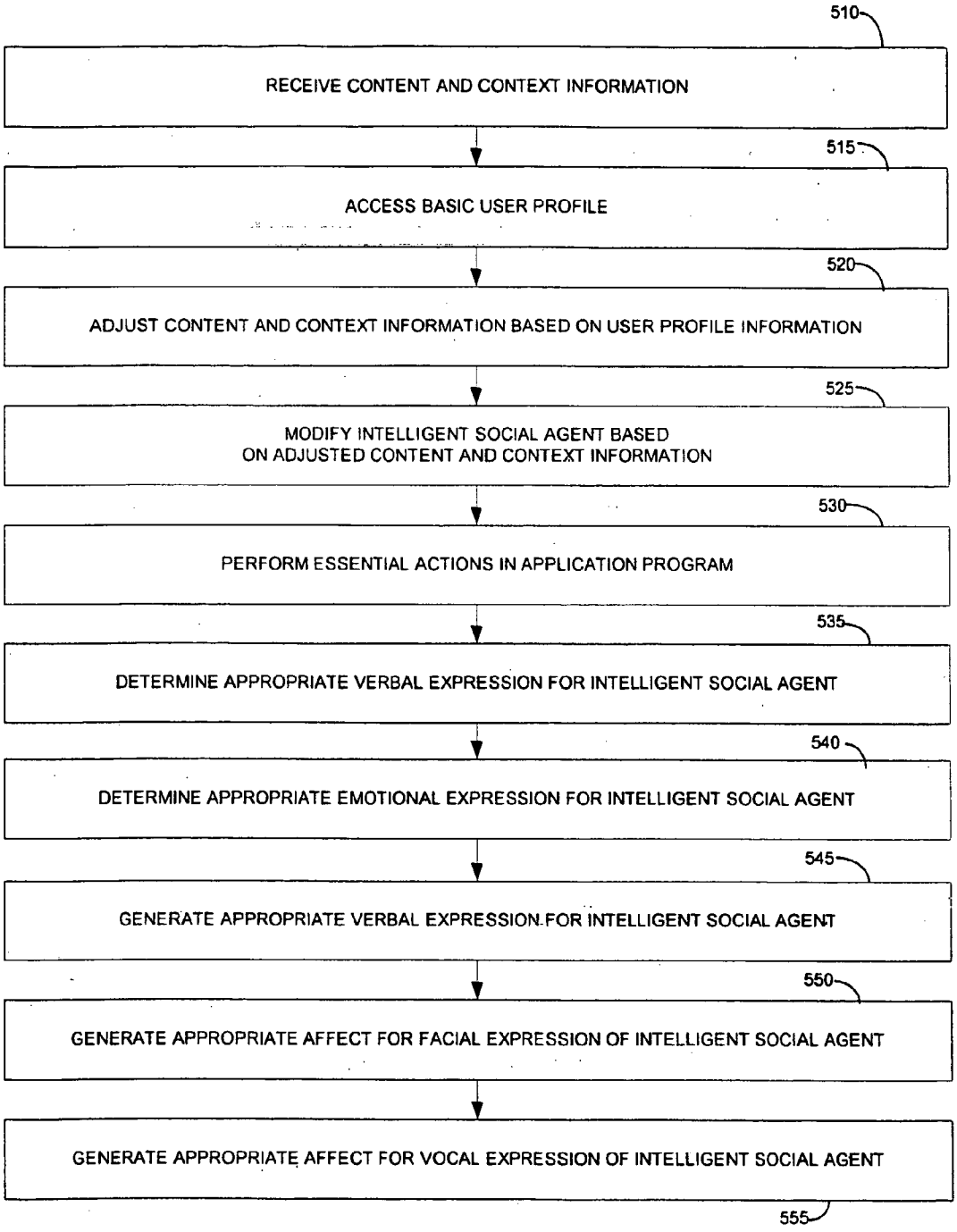

500
510
RECEIVE CONTENT AND CONTEXT INFORMATION
515
ACCESS BASIC USER PROFILE
520
ADJUST CONTENT AND CONTEXT INFORMATION BASED ON USER PROFILE INFORMATION
525
MODIFY INTELLIGENT SOCIAL AGENT BASED ON ADJUSTED CONTENT AND CONTEXT INFORMATION
530
PERFORM ESSENTIAL ACTIONS IN APPLICATION PROGRAM
535
DETERMINE APPROPRIATE VERBAL EXPRESSION FOR INTELLIGENT SOCIAL AGENT
540
DETERMINE APPROPRIATE EMOTIONAL EXPRESSION FOR INTELLIGENT SOCIAL AGENT
545
GENERATE APPROPRIATE VERBAL EXPRESSION FOR INTELLIGENT SOCIAL AGENT
550
GENERATE APPROPRIATE AFFECT FOR FACIAL EXPRESSION OF INTELLIGENT SOCIAL AGENT
GENERATE APPROPRIATE AFFECT FOR VOCAL EXPRESSION OF INTELLIGENT SOCIAL AGENT
555

FIG. 5

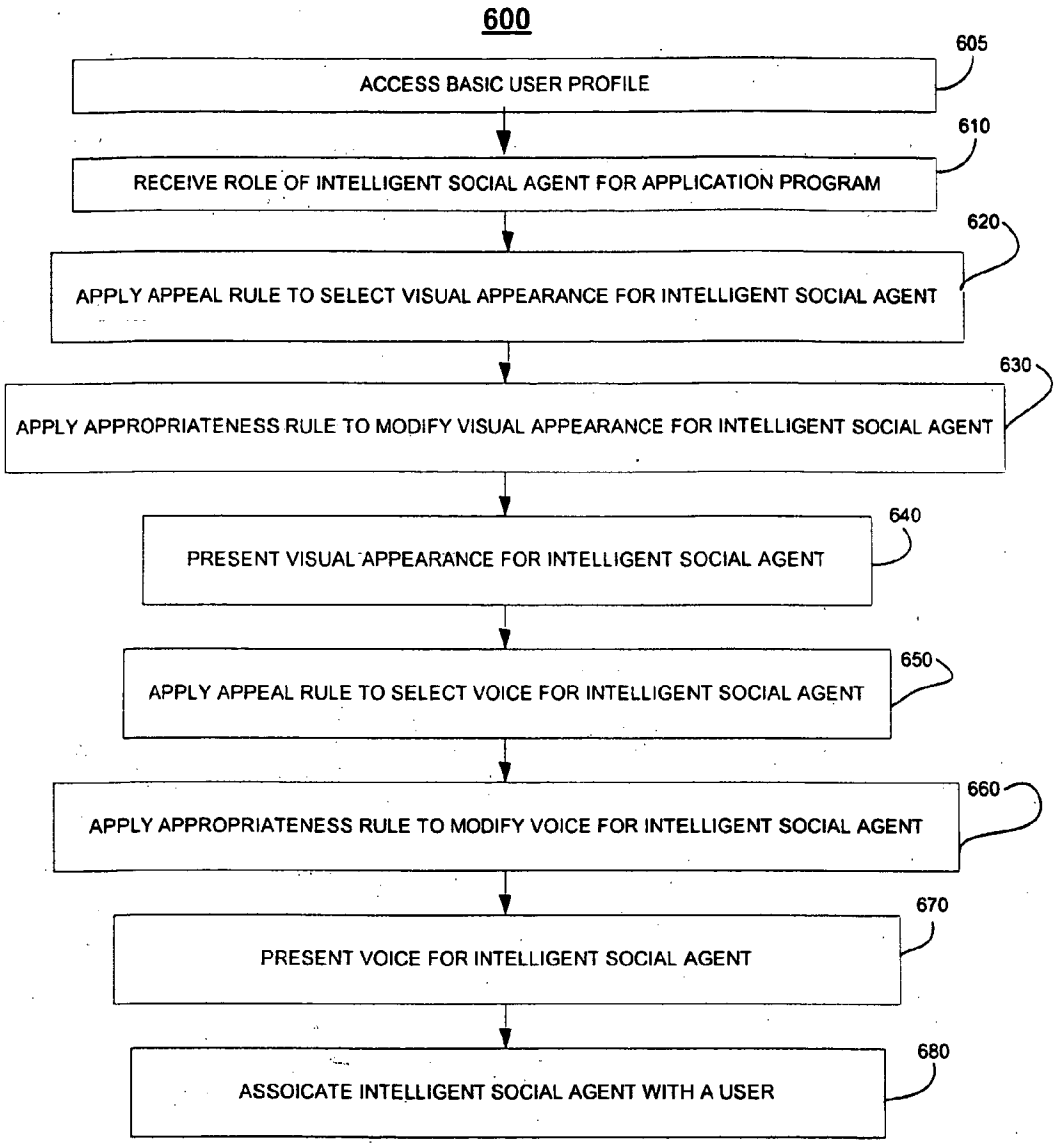

600
605
ACCESS BASIC USER PROFILE
610
RECEIVE ROLE OF INTELLIGENT SOCIAL AGENT FOR APPLICATION PROGRAM
620
APPLY APPEAL RULE TO SELECT VISUAL APPEARANCE FOR INTELLIGENT SOCIAL AGENT
630
APPLY APPROPRIATENESS RULE TO MODIFY VISUAL APPEARANCE FOR INTELLIGENT SOCIAL AGENT
640
PRESENT VISUAL APPEARANCE FOR INTELLIGENT SOCIAL AGENT
650
APPLY APPEAL RULE TO SELECT VOICE FOR INTELLIGENT SOCIAL AGENT
660
APPLY APPROPRIATENESS RULE TO MODIFY VOICE FOR INTELLIGENT SOCIAL AGENT
670
PRESENT VOICE FOR INTELLIGENT SOCIAL AGENT
680
ASSOICATE INTELLIGENT SOCIAL AGENT WITH A USER

FIG. 6

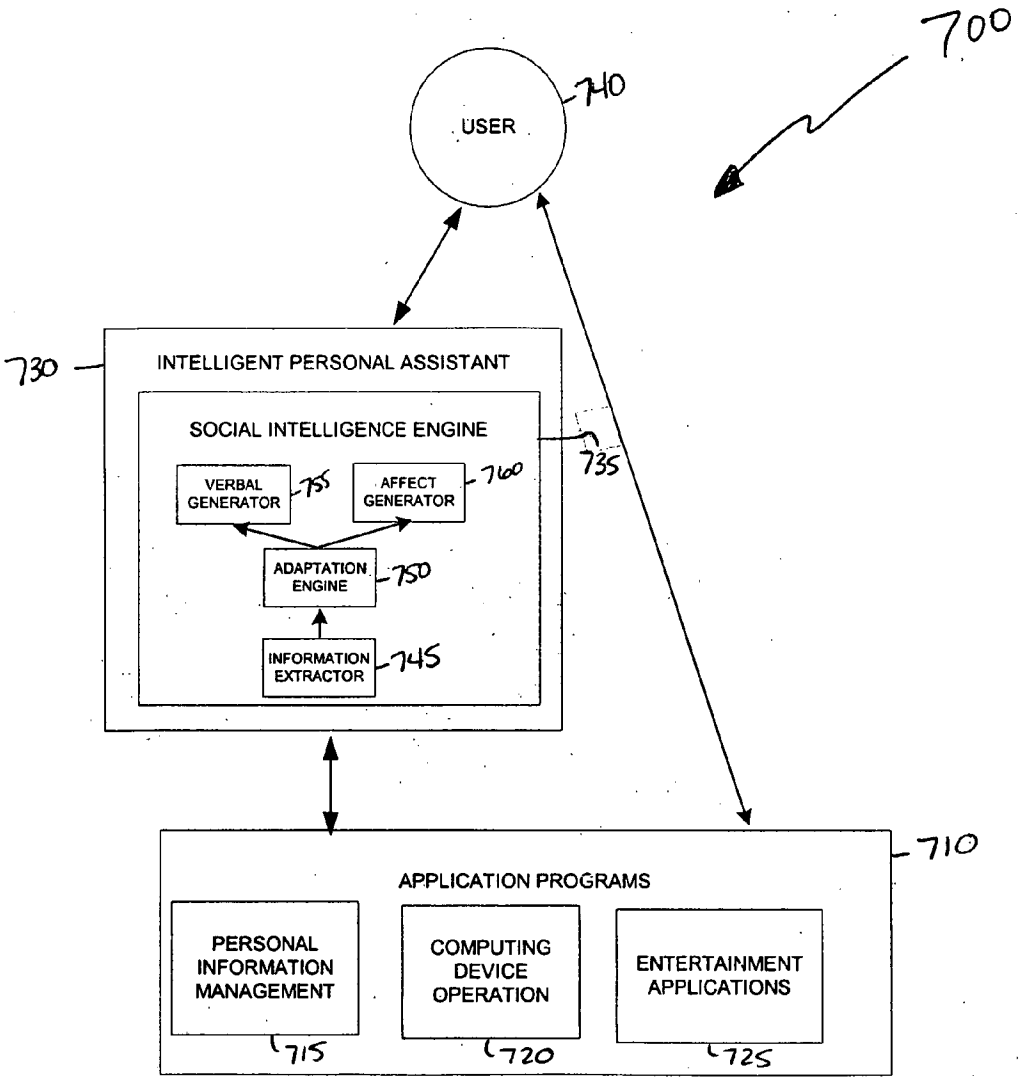

700
USER
740
730
INTELLIGENT PERSONAL ASSISTANT
SOCIAL INTELLIGENCE ENGINE
735
VERBAL GENERATOR
755
AFFECT GENERATOR
760
ADAPTATION ENGINE
750
INFORMATION EXTRACTOR
745
710
APPLICATION PROGRAMS
PERSONAL INFORMATION MANAGEMENT
715
COMPUTING DEVICE OPERATION
720
ENTERTAINMENT APPLICATIONS
725

FIG. 7

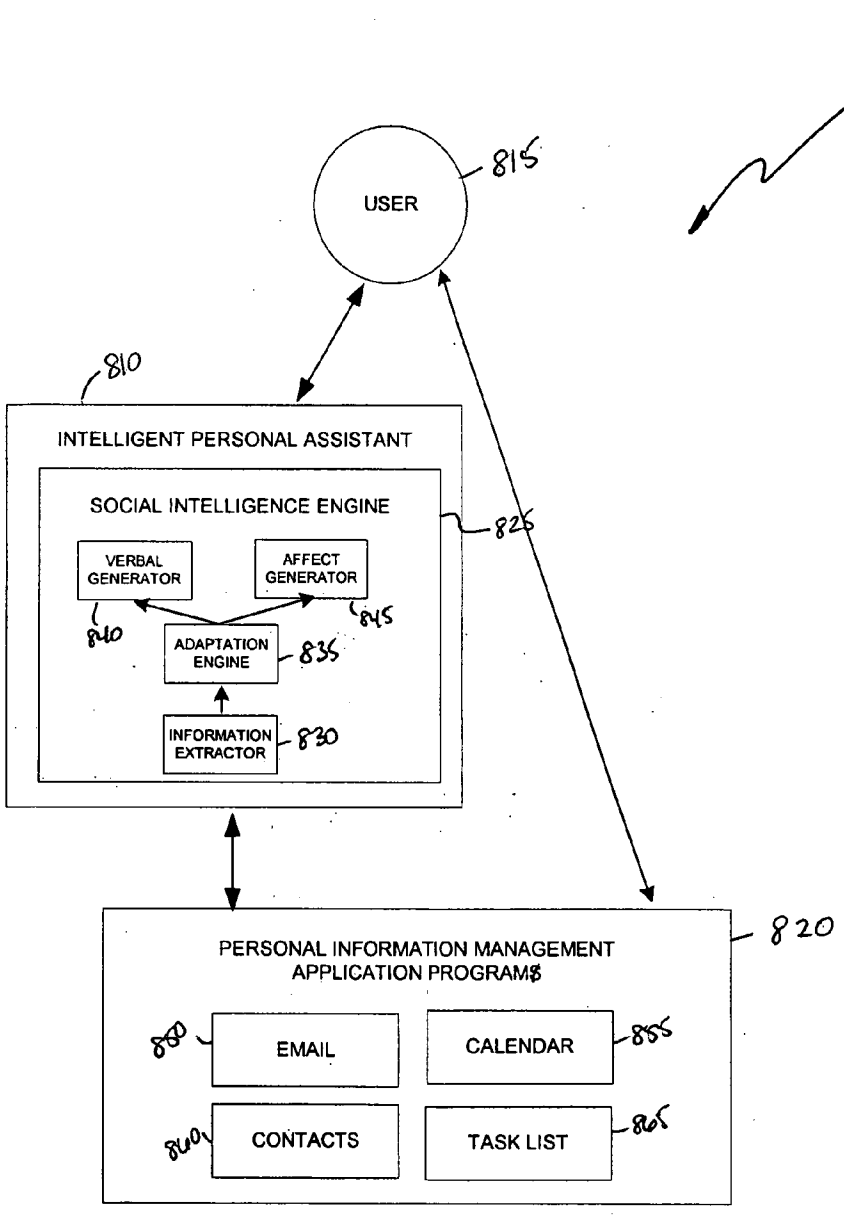

800
USER
815
810
INTELLIGENT PERSONAL ASSISTANT
SOCIAL INTELLIGENCE ENGINE
825
VERBAL GENERATOR
AFFECT GENERATOR
840
845
ADAPTATION ENGINE
835
INFORMATION EXTRACTOR
830
820
PERSONAL INFORMATION MANAGEMENT APPLICATION PROGRAMS
EMAIL
CALENDAR
CONTACTS
TASK LIST

FIG. 8

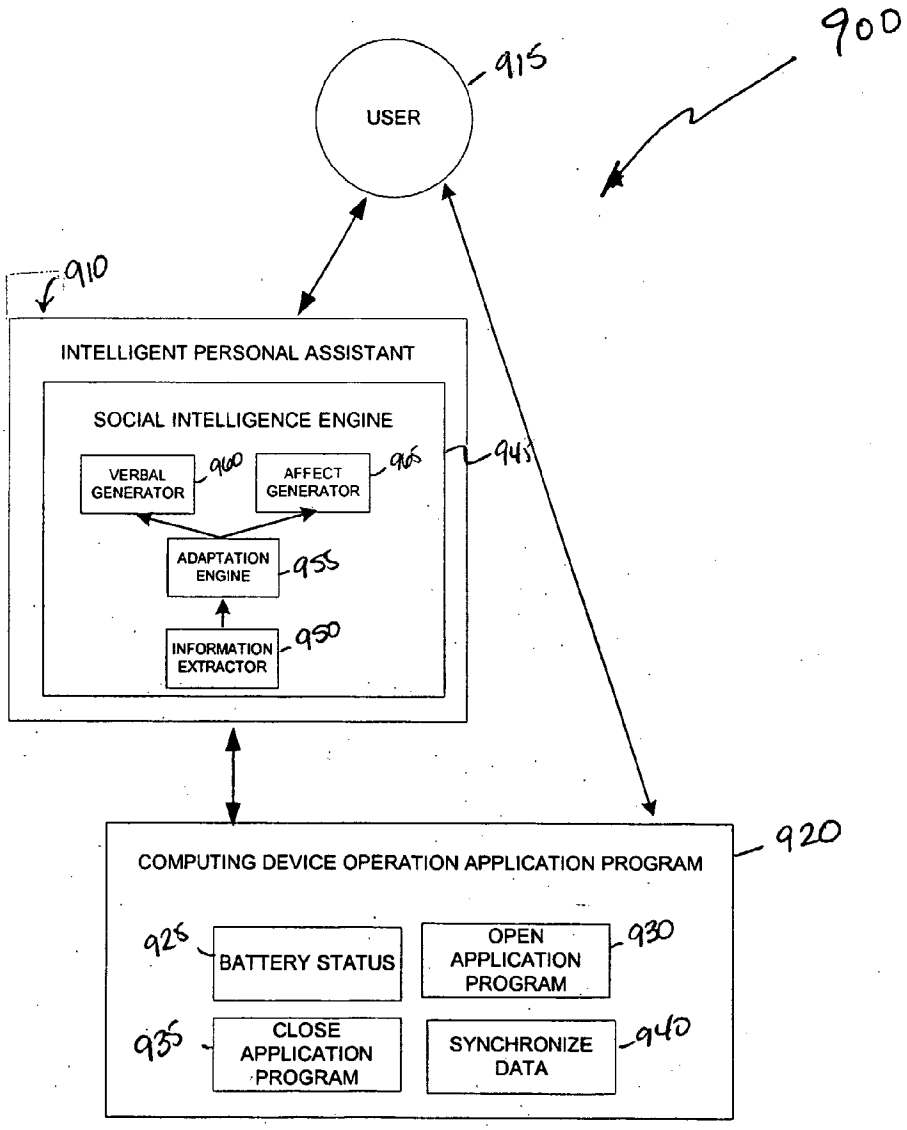

900
915
USER
910
INTELLIGENT PERSONAL ASSISTANT
SOCIAL INTELLIGENCE ENGINE
VERBAL GENERATOR
960
AFFECT GENERATOR
965
945
ADAPTATION ENGINE
955
INFORMATION EXTRACTOR
950
920
COMPUTING DEVICE OPERATION APPLICATION PROGRAM
925
BATTERY STATUS
OPEN APPLICATION PROGRAM
930
935
CLOSE APPLICATION PROGRAM
SYNCHRONIZE DATA
940

FIG. 9

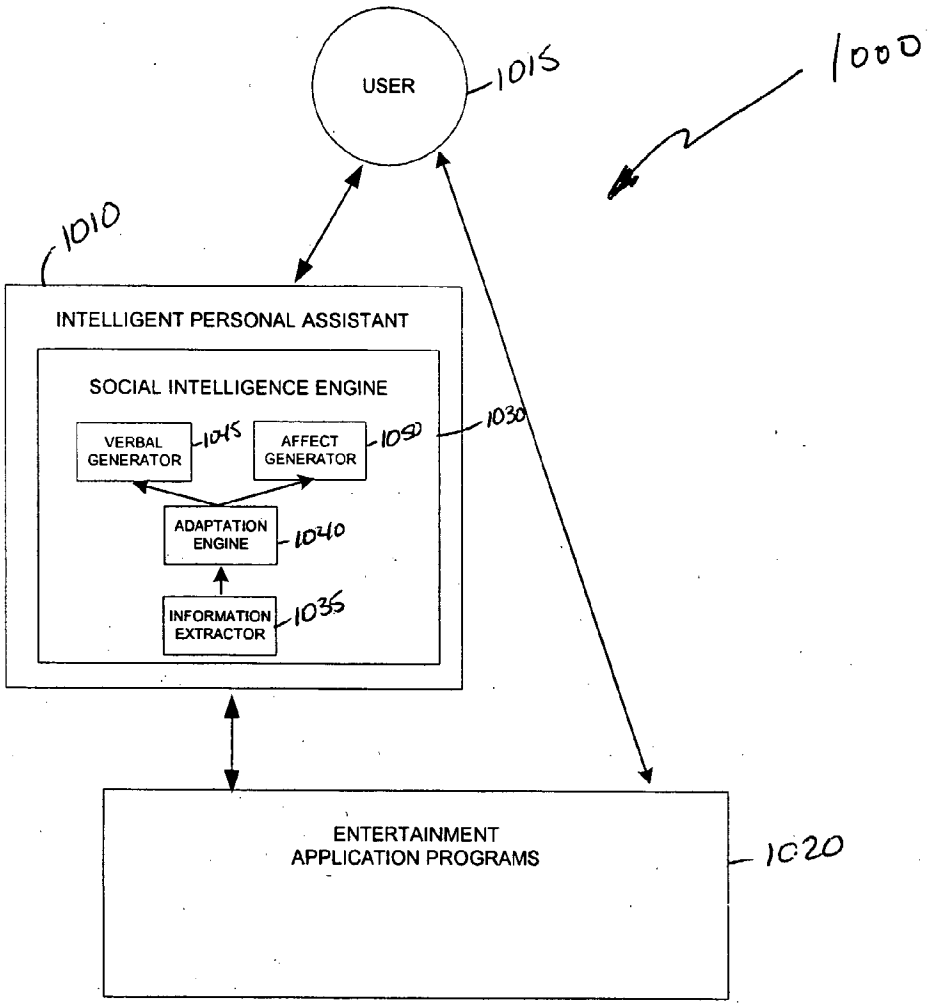

USER
1015
1000
1010
INTELLIGENT PERSONAL ASSISTANT
SOCIAL INTELLIGENCE ENGINE
VERBAL GENERATOR
1045
AFFECT GENERATOR
1050
1030
ADAPTATION ENGINE
1040
INFORMATION EXTRACTOR
1035
ENTERTAINMENT APPLICATION PROGRAMS
1020

FIG. 10

Electronic Patent Application Fee Transmittal

Application Number:	10743476
Filing Date:	23-Dec-2003
Title of Invention:	Proactive user interface
First Named Inventor/Applicant Name:	Jong-Goo Lee
Filer:	Paul J. Farrell/Deanna Marman
Attorney Docket Number:	678-1264

Filed as Large Entity

Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
Submission- Information Disclosure Stmt	1806	1	180	180
		Total in USD ($)		**180**

Electronic Acknowledgement Receipt

EFS ID:	8761834
Application Number:	10743476
International Application Number:	
Confirmation Number:	9615
Title of Invention:	Proactive user interface
First Named Inventor/Applicant Name:	Jong-Goo Lee
Customer Number:	66547
Filer:	Paul J. Farrell/Deanna Marman
Filer Authorized By:	Paul J. Farrell
Attorney Docket Number:	678-1264
Receipt Date:	03-NOV-2010
Filing Date:	23-DEC-2003
Time Stamp:	15:42:48
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Credit Card
Payment was successfully received in RAM	$180
RAM confirmation Number	2026
Deposit Account	504053
Authorized User	FARRELL,PAUL J.

The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:

Charge any Additional Fees required under 37 C.F.R. Section 1.16 (National application filing, search, and examination fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.19 (Document supply fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.20 (Post Issuance fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Transmittal Letter	Supplemental_IDS.pdf	100288 22fc023fc825799bce6af16b9b793deb5eccaeca9	no	3
Warnings:					
Information:					
2	Foreign Reference	CA_2536233.pdf	4955785 e697666643eb844128645464bb7890784b40cfe8	no	116
Warnings:					
Information:					
3	NPL Documents	CA_OA.pdf	201948 b40667c75486301e41fc152503128c5f02831d00	no	5
Warnings:					
Information:					
4	Foreign Reference	WO_03073417.pdf	2044235 b53d1126b2bc0eb64c4975f1221b98845c8937a3	no	49
Warnings:					
Information:					
5	Fee Worksheet (PTO-875)	fee-info.pdf	29550 3a94490b46bdc501732fee3dbbf7a51f70b4d56b	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	7331806		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

Electronic Acknowledgement Receipt

EFS ID:	8761834
Application Number:	10743476
International Application Number:	
Confirmation Number:	9615
Title of Invention:	Proactive user interface
First Named Inventor/Applicant Name:	Jong-Goo Lee
Customer Number:	66547
Filer:	Paul J. Farrell/Deanna Marman
Filer Authorized By:	Paul J. Farrell
Attorney Docket Number:	678-1264
Receipt Date:	03-NOV-2010
Filing Date:	23-DEC-2003
Time Stamp:	15:42:48
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Credit Card
Payment was successfully received in RAM	$180
RAM confirmation Number	2026
Deposit Account	504053
Authorized User	FARRELL,PAUL J.

The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:

Charge any Additional Fees required under 37 C.F.R. Section 1.16 (National application filing, search, and examination fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.19 (Document supply fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.20 (Post Issuance fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Transmittal Letter	Supplemental_IDS.pdf	100288 22fc023fc825799bce6af16b9b793deb5eccaeca9	no	3
Warnings:					
Information:					
2	Foreign Reference	CA_2536233.pdf	4955785 e697666643eb844128645464bb7890784b40cfe8	no	116
Warnings:					
Information:					
3	NPL Documents	CA_OA.pdf	201948 b40667c75486301e41fc152503128c5f02831d00	no	5
Warnings:					
Information:					
4	Foreign Reference	WO_03073417.pdf	2044235 b53d1126b2bc0eb64c4975f1221b98845c8937a3	no	49
Warnings:					
Information:					
5	Fee Worksheet (PTO-875)	fee-info.pdf	29550 3a94490b46bdc501732fee3dbbf7a51f70b4d56b	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	7331806		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/743,476	12/23/2003	Jong-Goo Lee	678-1264	9615

66547 7590 07/13/2009

THE FARRELL LAW FIRM, LLP
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
THERIAULT, STEVEN B	

ART UNIT	PAPER NUMBER
2179	

MAIL DATE	DELIVERY MODE
07/13/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

United States Patent and Trademark Office

Under Secretary of Commerce for Intellectual Property and
Director of the United States Patent and Trademark Office
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

THE FARRELL LAW FIRM, P.C.
290 BROADHOLLOW ROAD
SUITE 210E
MELVILLE, NY 11747

Appeal No: 2009-012781
Application: 10/743,476
Appellant: Jong-Goo Lee et al.

Board of Patent Appeals and Interferences
Docketing Notice

Application 10/743,476 was received from the Technology Center at the Board on July 07, 2009 and has been assigned Appeal No: 2009-012781.

A review of the file indicates that the following documents have been filed by appellant:

Appeal Brief filed on:	July 07, 2008
Reply Brief filed on:	December 08, 2008
Request for Hearing filed on:	NONE

In all future communications regarding this appeal, please include both the application number and the appeal number.

The mailing address for the Board is:

BOARD OF PATENT APPEALS AND INTERFERENCES
UNITED STATES PATENT AND TRADEMARK OFFICE
P.O. BOX 1450
ALEXANDRIA, VIRGINIA 22313-1450

The facsimile number of the Board is 571-273-0052. Because of the heightened security in the Washington D.C. area, facsimile communications are recommended. Telephone inquiries can be made by calling 571-272-9797 and should be directed to a Program and Resource Administrator.

By order of the Board of Patent Appeals and Interferences.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/743,476	12/23/2003	Jong-Goo Lee	678-1264	9615

66547 7590 06/25/2009

THE FARRELL LAW FIRM, LLP
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
THERIAULT, STEVEN B	
ART UNIT	PAPER NUMBER
2179	

MAIL DATE	DELIVERY MODE
06/25/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES DEPARTMENT OF COMMERCE
U.S. Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450

APPLICATION NO./ CONTROL NO.	FILING DATE	FIRST NAMED INVENTOR / PATENT IN REEXAMINATION	ATTORNEY DOCKET NO.
10743476	12/23/2003	LEE ET AL.	678-1264

THE FARRELL LAW FIRM, LLP
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
STEVEN B. THERIAULT	
ART UNIT	**PAPER**
2179	20090619

DATE MAILED:

Please find below and/or attached an Office communication concerning this application or proceeding.

Commissioner for Patents

This communication is subsequent to the office communication mailed 06/09/2009 to specifically address in the record the IDS entered on 12/30/2008. As the examiner indicated on 06/09/2009, the IDS was considered and entered, however the record does not show the IDS with the examiners signature and initials on it and to prevent possible delays at the appeals center, this communication specifically identifies that the examiner considered the IDS and shows via signature that the IDS was considered.

/Steven B Theriault/
Primary Examiner
Art Unit: 2179

Form PTO-1449 U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE	ATTY. DOCKET NO. 678-1264	SERIAL NO. 10/743,476
INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Use several sheets if necessary)	APPLICANT Jong-Goo LEE et al.	
	FILING DATE December 23, 2003	GROUP ART UNIT

U.S. PATENT DOCUMENTS

EXAMINER INITIAL		DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE

FOREIGN PATENT DOCUMENTS

		DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	TRANSLATION YES	TRANSLATION NO
/S.T./		KR 1020020024824	April 3, 2002	Korea			X	
/S.T./		CN 1274747	November 29, 2000	China			X	

OTHER PRIOR ART (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER /Steven Theriault/	DATE CONSIDERED 06/19/2009

* EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw citation if not in conformance and not considered. Include copy of this form with next communication to ap

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/743,476	12/23/2003	Jong-Goo Lee	678-1264	9615

66547 7590 06/09/2009

THE FARRELL LAW FIRM, LLP
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
THERIAULT, STEVEN B	
ART UNIT	PAPER NUMBER
2179	

MAIL DATE	DELIVERY MODE
06/09/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES DEPARTMENT OF COMMERCE
U.S. Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450

APPLICATION NO./ CONTROL NO.	FILING DATE	FIRST NAMED INVENTOR / PATENT IN REEXAMINATION	ATTORNEY DOCKET NO.
10743476	12/23/2003	LEE ET AL.	678-1264

THE FARRELL LAW FIRM, LLP
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
STEVEN B. THERIAULT	
ART UNIT	**PAPER**
2179	20090607

DATE MAILED:

Please find below and/or attached an Office communication concerning this application or proceeding.

Commissioner for Patents

In response to the order returing the appeal to the Examiner, the Examiner hereby vacates the advisory action mailed 04/20/2009 as an improper response and this communication represents a proper response in accordance with MPEP 1208, part II. The reply brief filed 12/08/2008 has been entered and considered. The application has been forwarded to the Board of Patent Appeals and Interferences for decision on the appeal. The IDS entered on 04/20/2009 and considered with that advisory action was made a part of the record and therefore will not be remailed with this communication.

/Steven B Theriault/
Primary Examiner
Art Unit: 2179

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/743,476	12/23/2003	Jong-Goo Lee	678-1264	9615

66547 7590 06/01/2009

THE FARRELL LAW FIRM, LLP
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
THERIAULT, STEVEN B	

ART UNIT	PAPER NUMBER
2179	

MAIL DATE	DELIVERY MODE
06/01/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte: JONG-GOO LEE, EYAL TOLEDANO, NATAN LINDER, YARIV EISENBERG, AND RAN BEN-YAIR

Application No. 10/743,476
Technology Center 2100

Mailed: May 29, 2009

Before ERIC W. HAWTHORNE, *Supervisory Paralegal Specialist*

HAWTHORNE, *Supervisory Paralegal Specialist.*

ORDER RETURNING UNDOCKETED APPEAL TO EXAMINER

This application was electronically received by the Board of Patent Appeals and Interferences on April 21, 2009. A review of the application revealed that it is not ready for docketing as an appeal. Accordingly, the application is herewith being returned to the Examiner to address the following matter(s) requiring attention prior to docketing.

PRIOR ORDER FOR RETURN

A prior "Order Returning Undocketed Appeal to Examiner" was mailed on April 8, 2009 wherein the Examiner was instructed that corrections were required. A review of the file finds that the required corrections have not been made or have not been made in entirety. The matters still requiring attention prior to docketing are identified below.

EXAMINER'S CONSIDERATION OF REPLY BRIEF

A Reply Brief was filed on December 8, 2008, in response to the Examiner's Answer mailed October 7, 2008.

Title 37, Code of Federal Regulations, §41.43 states:

> (a)(1)... the primary examiner must acknowledge receipt and entry of the reply brief. In addition, the primary examiner may withdraw the final rejection and reopen prosecution or may furnish a supplemental examiner's answer responding to any new issue raised in the reply brief.

The Communication mailed April 20, 2009, was an improper acknowledgment of the Reply Brief, as it constitutes a Supplemental Examiner's Answer as per MPEP§ 1208, part II. A Supplemental Examiner's Answers requires a Director or designees approval.

CONCLUSION

Accordingly, it is

ORDERED that the application is returned to the Examiner:

1) vacate the Communication mailed April 20, 2009;

2) generate and mail either:

a) a revised Communication properly acknowledging to the Reply Brief dated December 8, 2009 in accordance with MPEP§ 1208, part II;

OR

b) issue a Supplemental Examiner's Answer with the required signature (Technology Center Director or designee), if appropriate, and;

3) for such further action as may be appropriate.

If there are any questions pertaining to this Order, please contact the Board of Patent Appeals and Interferences at 571-272-9797.

EWH/nhl

THE FARRELL LAW FIRM, LLP
290 Broadhollow Road
Suite 210E
Melville NY 11747

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/743,476	12/23/2003	Jong-Goo Lee	678-1264	9615

66547 7590 04/20/2009

THE FARRELL LAW FIRM, P.C.
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
THERIAULT, STEVEN B	

ART UNIT	PAPER NUMBER
2179	

MAIL DATE	DELIVERY MODE
04/20/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Advisory Action After the Filing of an Appeal Brief	Application No. 10/743,476	Applicant(s) LEE ET AL.	
	Examiner STEVEN B. THERIAULT	Art Unit 2179	

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

The reply filed 08 December 2008 is acknowledged.

1. ☐ The reply filed on or after the date of filing of an appeal brief, but prior to a final decision by the Board of Patent Appeals and Interferences, will not be entered because:

   a. ☐ The amendment is not limited to canceling claims (where the cancellation does not affect the scope of any other pending claims) or rewriting dependent claims into independent form (no limitation of a dependent claim can be excluded in rewriting that claim). See 37 CFR 41.33(b) and (c).

   b. ☐ The affidavit or other evidence is not timely filed before the filing of an appeal brief. See 37 CFR 41.33(d)(2).

2. ☐ The reply is not entered because it was not filed within the two month time period set forth in 37 CFR 41.39(b), 41.50(a)(2), or 41.50(b) (whichever is appropriate). Extensions of time under 37 CFR 1.136(a) are not available.

   Note: This paragraph is for a reply filed in response to one of the following: (a) an examiner's answer that includes a new ground of rejection (37 CFR 41.39(a)(2)); (b) a supplemental examiner's answer written in response to a remand by the Board of Patent Appeals and Interferences for further consideration of rejection (37 CFR 41.50(a)(2)); or (c) a Board of Patent Appeals and Interferences decision that includes a new ground of rejection (37 CFR 41.50(b)).

3. ☒ The reply is entered. An explanation of the status of the claims after entry is below or attached.

4. ☒ Other: *See Continuation Sheet*

/Steven B Theriault/
Primary Examiner
Art Unit: 2179

Continuation of 4 Other: a. The information disclosure statement filed 12/30/2008 fails to comply with 37 CFR 1.98(a)(3) because it does not include a concise explanation of the relevance, as it is presently understood by the individual designated in 37 CFR 1.56(c) most knowledgeable about the content of the information, of each patent listed that is not in the English language. It has been placed in the application file, but the information referred to therein has not been considered.

b. The reply brief has been entered but contains substantially the same argument as presented in the appeal. It is clear applicant does not believe the rejection is a proper 102 and relies on two arguments. 1) applicant argues the rejection is based on multiple embodiments, which is improper according to the supplied netmoney decision. 2) applicant argues the rejection is based on multiple references and is improper. The answer filed 10/07/08 is clear as to the position of the office and does not agree with applicant. In response and in summary, a reference to specific paragraphs, columns, pages, or figures in a cited prior art reference is not limited to preferred embodiments or any specific examples. It is well settled that a prior art reference, in its entirety, must be considered for all that it expressly teaches and fairly suggests to one having ordinary skill in the art. Stated differently, a prior art disclosure reading on a limitation of Applicant's claim cannot be ignored on the ground that other embodiments disclosed were instead cited. Therefore, the Examiner's citation to a specific portion of a single prior art reference is not intended to exclusively dictate, but rather, to demonstrate an exemplary disclosure commensurate with the specific limitations being addressed. In re Heck, 699 F.2d 1331, 1332-33,216 USPQ 1038, 1039 (Fed. Cir. 1983) (quoting In re Lemelson, 397 F.2d 1006,1009, 158 USPQ 275, 277 (CCPA 1968)). In re: Upsher-Smith Labs. v. Pamlab, LLC, 412 F.3d 1319, 1323, 75 USPQ2d 1213, 1215 (Fed. Cir. 2005); In re Fritch, 972 F.2d 1260, 1264, 23 USPQ2d 1780, 1782 (Fed. Cir. 1992); Merck & Co. v. Biocraft Labs., Inc., 874 F.2d 804, 807, 10 USPQ2d 1843, 1846 (Fed. Cir. 1989); In re Fracalossi, 681 F.2d 792,794 n.1,215 USPQ 569, 570 n.1 (CCPA 1982); In re Lamberti, 545 F.2d 747, 750, 192 USPQ 278, 280 (CCPA 1976); In re Bozek, 416 F.2d 1385, 1390, 163 USPQ 545, 549 (CCPA 1969). In this example, it appears applicant has not looked at the entire reference in terms of the user interface referred referred to and used in the cited section and figure 15. Column 85, lines 5-67 and Figure 15 were used in the initial rejection but other sections of the reference further teach the same embodiment and to ignore those sections is improper. For example, Figure 15, is described in Column 83 as a flow chart for a predictive user interface. Figure 15, contains a flow diagram for leading the user through a correct sequence of steps on the user interface displayed on the screen (See column 83, bottom, Column 84, top). Figure 15, shows two steps of "analyzing a program sequence to predict a next action" and then "displaying the predicted action", which are predictive interface steps. Column 83, bottom describes the preferred embodiment as a VCR interface. The cited column 85 used in the rejection refers to the same figure 15 and user interface for a VCR but also refers to the steps shown to the user in the interface. Column 97-98 also refer to "the interface" and the mechanism for interaction with the interface. Therefore, in spite of applicants assertion the rejection is based on the entire reference. There are multiple sections of the reference that refer to the same embodiment and should be considered by applicant. The user interface described column 97-98 is the same interface displayed to the user to program a user interface as shown in example 1 and in figure 15. While example 5 refers to an additional element of using an infrared input device that allows for wireless interaction, the user interface is the same user interface discussed in example 1 column 83 and 85. Clearly, column 97-98 are directed to "the interface". Therefore, the rejection not only points out the specific limitations in the same reference but also teaches the limitations as arranged in the claim. In the cited section, there is a user interface, there is a proactive alteration of the interface based on a system detected pattern, and there is an alteration of at least on function of the system based on the pattern. Turning to the second argument, the examiner answer and the rejection serve as a record for what is used by the examiner. The cited incorporated by reference patents are a part of the Hoffberg reference as if they were written in the document and should be considered by applicant as relevant prior art. The examiner referred to the cited art to show the state of the art at the time of filing. The rejection also cited column 10, lines 15-31 that expressly states the incorporated patents are relevant to pattern recognition and are relevant to the interface of the present invention. Therefore, the examiner did not use a multiple rejection 102 and 2131.01 does not apply. All of the incorporated references are a part of the Hoffberg reference and the examiner did not specifically cite any additional reference to teach the limitations of the claim. Moreover, 2163.07 applies as the cited patents are a part of the text of the application as filed and should be considered as such.

OK TO ENTER: /S.T./

Form PTO-1449 U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE	ATTY. DOCKET NO. 678-1264	SERIAL NO. 10/743,476
INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Use several sheets if necessary)	APPLICANT Jong-Goo LEE et al.	
	FILING DATE December 23, 2003	GROUP ART UNIT

U.S. PATENT DOCUMENTS

EXAMINER INITIAL		DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE

FOREIGN PATENT DOCUMENTS

		DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	TRANSLATION YES	TRANSLATION NO
~~Not considered per MPEP 37 CFR 1.98 (b)~~								
		KR 1020020024824	April 3, 2002	Korea			X	
		CN 1274747	November 29, 2000	China			X	

OTHER PRIOR ART (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER	DATE CONSIDERED
/Steven Theriault/	04/15/2009

* EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw citation if not in conformance and not considered. Include copy of this form with next communication to ap

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/743,476	12/23/2003	Jong-Goo Lee	678-1264	9615

66547 7590 04/08/2009

THE FARRELL LAW FIRM, P.C.
290 Broadhollow Road
Suite 210E
Melville, NY 11747

EXAMINER	
THERIAULT, STEVEN B	
ART UNIT	PAPER NUMBER
2179	

MAIL DATE	DELIVERY MODE
04/08/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte: JONG-GOO LEE, EYAL TOLEDANO, NATAN LINDER, YARIV EISENBERG, and RAN BEN-YAIR

Application No. 10/743,476
Technology Center 2100

Mailed: April 8, 2009

Before Deborah L. Perry, Supervisory Paralegal Specialist, Review Team.

Perry, Supervisory Paralegal Specialist, Review Team.

ORDER RETURNING UNDOCKETED APPEAL TO EXAMINER

This application was electronically received by the Board of Patent Appeals and Interferences on December 31, 2008. A review of the application revealed that it is not ready for docketing as an appeal. Accordingly, the application is herewith being returned to the Examiner to address the following matter(s) requiring attention prior to docketing.

EXAMINER'S CONSIDERATION OF REPLY BRIEF

A Reply Brief was filed in this application on December 8, 2008. There is no evidence on the record indicating that the Examiner has considered the Reply Brief in accordance with 37 CFR § 41.43(a)(1) and MPEP § 1208, part II.

INFORMATION DISCLOSURE STATEMENT

Appellant filed an Information Disclosure Statement (IDS) dated December 30, 2008. There is no indication on the record that the Examiner has considered the above Information Disclosure Statement. MPEP § 609 requires the Examiner to consider any Information Disclosure Statement filed by Applicant if timely submitted. A written communication notifying Appellant of the Examiner's consideration of the above Information Disclosure Statement is required.

CONCLUSION

Accordingly, it is

ORDERED that the application is returned to the Examiner to:

1) consider the Reply Brief filed December 8, 2008, as indicated above;

2) consider the Information Disclosure Statement filed December 30, 2008; and

3) for such further action as may be appropriate.

If there are any questions pertaining to this Order, please contact the Board of Patent Appeals and Interferences at 571-272-9797.

DLP/bar

THE FARRELL LAW FIRM, P.C.
290 BROADHOLLOW ROAD
SUITE 210E
MELVILLE, NY 11747

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Jong-Goo LEE et al.

Serial No.: 10/743,476 Docket: 678-1264

Filed: December 23, 2003 Dated: December 30, 2008

For: PROACTIVE USER INTERFACE

Mail Stop PETITION
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

PETITION FOR CONSIDERATION OF SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT

Sir:

Applicant hereby petitions that the references cited in the attached Supplemental Information Disclosure Statement be considered by the U.S. Patent and Trademark Office. A final Office Action was mailed February 7, 2008 in connection with the above-identified patent application.

The items listed in the attached Supplemental Information Disclosure Statement were cited by the Russian Patent Office in a counterpart application, namely Appln. No. 2006110932. A copy of the Russian Office Action dated September 30, 2008 is attached hereto.

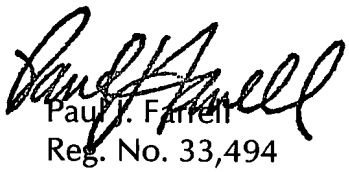
CERTIFICATION UNDER 37 C.F.R. §1.97(e)(2)

Applicant submits that each item of information contained in the attached Supplemental Information Disclosure Statement was cited in a communication from a foreign patent office in a counterpart foreign application no more than three months prior to the filing of the Statement.

The $130.00 surcharge will be paid via credit card.

Please charge any additional fees which may become due in connection with this filing to Deposit Account No. 50-4053.

Respectfully submitted,



Paul J. Farrell
Reg. No. 33,494
Attorney for Applicant

THE FARRELL LAW FIRM
333 Earle Ovington Blvd., Suite 701
Uniondale, NY 11553
(516) 228-3565

(19) KOREAN INTELLECTUAL PROPERTY OFFICE

KOREAN PATENT ABSTRACTS

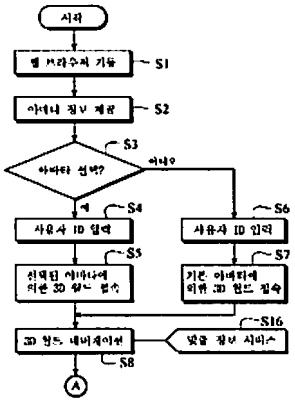
(11)Publication number: 1020020024824 A
(43)Date of publication of application: 03.04.2002

(21)Application number: 1020000056584
(22)Date of filing: 27.09.2000
(30)Priority: ..
(51)Int. Cl G06F 19/00

(71)Applicant: GOMID, INC.
(72)Inventor: CHO, GAP SON

(54) SYSTEM AND METHOD FOR ANALYZING PATTERN OF USER BEHAVIOR BASED ON NETWORK

(57) Abstract:



PURPOSE: A system and method for analyzing patterns of user behavior are provided to analyze a users behavior pattern using a virtual image having only the users basic information, without violating privacy, and to service customized information to the user according to an analyzed result. CONSTITUTION: If a user drives a 3-D environmental web browser embedded in a computer(S1), a web server extracts avatar information, or virtual image information, stored in a database and displays it on the web browsing screen of the user computer through a network(S2). A control process judges whether the user selects a desired avatar among the provided avatar information(S3). In case that an avatar is selected, the control process requests the user to input his ID and password. If the user inputs his ID and password(S4), the control process provides a 3-D web browsing screen, on which the selected avatar appears, on the user computer so that 3-D world navigation can be achieved through the selected avatar(S5,S6).

Legal Status
Date of request for an examination (20000927)
Notification date of refusal decision ( )
Final disposal of an application (registration)
Date of final disposal of an application (20031226)
Patent registration number (1004262800000)
Date of registration (20040326)
Number of opposition against the grant of a patent ( )
Date of opposition against the grant of a patent ( )
Number of trial against decision to refuse ( )
Date of requesting trial against decision to refuse ( )

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。Int. Cl. [7]
G06F 19/00E0

(11) 공개번호 특2002-0024824
(43) 공개일자 2002년04월03일

(21) 출원번호 10-2000-0056584
(22) 출원일자 2000년09월27일

(71) 출원인 (주) 고미드
김종민
서울 강남구 대치3동 1005

(72) 발명자 조갑손
서울특별시성동구금호1가128-10성호연립사동201호

(74) 대리인 장수길
주성민

심사청구 : 있음

(54) 네트웍 기반 사용자 행동 패턴 분석 시스템 및 그 방법

요약

본 발명은 웹 브라우저 환경에서 사용자의 기본 정보만을 갖는 아바타를 이용하여, 사용자의 행동 패턴을 분석하고 그 분석 결과에 따라 사용자에게 적합한 맞춤 정보를 서비스하는 시스템 및 방법을 제공한다. 웹 브라우저의 기동시, 네트웍(20)을 통해 사용자 컴퓨터(1)의 웹 브라우징 화면 상에 복수의 아바타를 제공하여, 사용자가 원하는 아바타를 선택하도록 하고, 선택된 아바타(10)에 대응하는 사용자 ID 번호를 입력하도록 요구하고, 사용자 ID 번호가 입력되면, 아바타(10)가 등장하는 웹 브라우징 화면(11)을 사용자 컴퓨터(1) 상에 제공함으로써, 사용자 아바타(10)는 웹 브라우징 화면(11) 상에서 네비게이션하게 된다. 아바타(10)에 의한 네비게이션시, 아바타(10)의 위치, 이동 경로, 방문한 지역(zone), 머문 시간 등을 실시간으로 검출하고, 검출된 정보를 분석하여 사용자 행동 패턴 정보로서 웹 서버(32)의 데이터베이스(31)에 축적한다. 웹 브라우저의 기동시, 데이터 베이스(31) 내의 사용자 행동 패턴 정보에 기초하여 사용자에 적합한 맞춤 정보를 서비스한다.

대표도
도 5

색인어
3차원, 웹 브라우저, 아바타, 행동 패턴, 맞춤 정보, 쿠키 파일, 위치

명세서

도면의 간단한 설명

도 1은 본 발명에 따라 인터넷 등의 통신 네트웍을 통해 클라이언트와 웹 서버 간의 통신 체계를 도시한 도면.

도 2는 3차원 웹 브라우저 환경에서 제공하는 화면의 일례를 도시한 도면.

도 3은 사용자 컴퓨터로부터 웹 서버로 실시간으로 전송되는 데이터 스트림의 일례를 도시한 도면.

도 4는 데이터베이스 내에 저장된 사용자 행동 패턴 정보를 일례로서 도시한 도면.

도 5 및 도 6은 본 발명의 바람직한 실시예에 따라 웹 브라우징시 사용자의 행동 패턴을 분석하는 과정을 설명하기 위한 흐름도.

< 도면의 주요 부분에 대한 부호의 설명 >

1, 2,…,N : 사용자 컴퓨터

10 : 아바타

11 : 3차원 웹 브라우징 화면

20 : 네트웍

30 : 웹 사이트

31 : 데이터 베이스

32 : 웹 서버

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은, 네트웍을 이용하는 사용자의 행동 패턴을 분석하는 시스템에 관한 것으로서, 특히, 웹 브라우저 환경에서 사용자의 행동 패턴을 분석하여 사용자에게 적합한 맞춤 정보를 서비스하는 시스템 및 방법에 관한 것이다.

현재, 인터넷과 같은 통신 네트웍에서 사용자의 행동 패턴을 분석하는 데에 쿠키(cookie file) 파일 또는 로그 파일(log file)이 주로 이용되고 있다.

주지하는 바와 같이, 쿠키는 인터넷 웹 사이트와 사용자의 컴퓨터 사이에서 통신을 매개해 주는 정보로서, 사용자가 웹 사이트에 접속했을 때 사용자 컴퓨터로 전송되는 4kB의 작은 파일이다. 쿠키에는 사용자의 ID, 비밀 번호, 방문한 사이트 등의 정보가 담겨져 사용자 컴퓨터의 하드디스크에 저장된다. 이러한 쿠키 정보를 통해, 사용자가 다음에 해당 사이트를 방문할 경우, 웹 서버에서는 그가 누구인지, 어떤 정보를 주로 접근하였는지 등의 정보를 바로 파악할 수 있다. 즉, 사용자가 어떤 제품을 구입하였는지, 어떤 분야에 관심이 많은지를 파악할 수 있어, 웹 사이트 운영자 입장에서는 쿠키를 이용한 타깃 마케팅이 가능하다.

또한, 사용자 컴퓨터의 하드디스크에는 해당 사용자가 방문한 웹 사이트에 따라 여러개의 쿠키 파일이 존재하게 되며, 이러한 쿠키 파일이 해당 사이트로만 전송되는 것이 아니라 웹 브라우저의 버그로 인해 다른 웹 사이트의 운영자도 손쉽게 쿠키 파일을 입수할 수 있어, 개인 정보가 유출되어 프라이버시를 침해할 수 있는 문제가 있다. 따라서, 쿠키는 고의로 사용자의 정보를 빼낼 수 있는 통로 역할을 할 수도 있다. 쿠키를 통해 사용자의 암호를 유추해 낼 수도 있으며, 심한 경우에는 사용자 컴퓨터의 하드디스크에 저장된 파일을 파괴하거나 훔쳐낼 수도 있다.

더우기, 사용자가 웹 사이트에 접속할 때에만, 4kB의 쿠키 파일이 사용자 컴퓨터에 다운로드되어 사용자의 네비게이션 정보를 입수하게 되므로, 이러한 쿠키 파일을 인터넷 광고나 마케팅 전략 계획에 이용하고자 하는 인터넷 업계 입장에서는 사용자의 행동 패턴을 지속적이고 체계적으로 분석할 수 없어, 사용자가 가장 관심을 갖는 분야에 대한 정보(즉, 맞춤 정보)를 서비스할 수 없다고 하는 단점이 있다. 특히, 향후 폭 넓게 이용될 3차원의 웹 브라우저 환경에서는, 이러한 소용량의 쿠키 파일을 이용할 경우, 사용자가 3차원 웹 브라우저 환경을 항해하면서 발생한 방문 지역(access zone), 방문 횟수, 접속 시간 등의 정보를 실시간으로 파악하기가 곤란하여, 사용자의 행동 패턴을 충실히 분석할 수 없는 단점이 있다.

또한, 로그 파일은 웹 서버에 축적되는 것으로 컴퓨터 시스템의 모든 사용 내역을 기록하고 있는 파일이다. 이러한 로그 파일을 이용하여 인터넷 웹 사이트에서는 어떤 고객이 접속했는가에 따라 고객별로 상이한 웹 페이지를 보여 줄 수 있다. 로그 파일에 저장되는 클라이언트 인터넷 프로토콜(IP)은 클라이언트(즉, 사용자)가 인터넷을 시작한 위치 정보에 해당하는 것으로, 이러한 IP 정보를 통해 웹 사이트에 접속하는 사용자들의 정보를 파악할 수 있다. 그러나, 사용자가 동적 IP를 사용하거나 통상의 프록시 서버(proxy server)를 사용할 경우, 사용자의 IP를 정확하게 파악할 수 없게 된다. 또한, 사용자가 캐시(cache)를 사용하는 경우에는, 사용자의 페이지 뷰 데이터 및 뒤로 가기(back) 데이터가 로그 파일에 저장되지 않아 사용자의 네비게이션을 정확하게 분석할 수 없는 단점이 있다.

발명이 이루고자 하는 기술적 과제

따라서, 본 발명의 주된 목적은, 웹 브라우저 환경에서 사용자의 기본 정보만을 갖는 가상 이미지를 이용하여, 사용자의 개인 프라이버시를 침해하지 않으면서 사용자의 행동 패턴을 분석하고 그 분석 결과에 따라 사용자에게 적합한 맞춤 정보를 서비스하는 시스템 및 방법을 제공하는 데에 있다.

발명의 구성 및 작용

상기 목적을 달성하기 위해서, 본 발명의 바람직한 실시예에 따르면, 네트웍을 통한 웹 브라우징시에 발생하는 사용자의 네비게이션 정보를 분석하고, 분석 결과에 따라 해당 사용자에게 적합한 맞춤 정보를 서비스하는 시스템에 있어서,

복수의 지역(zone)으로 분할된 화면을 갖는 웹 브라우저가 내장된 사용자 컴퓨터와, 복수의 가상 이미지를 저장하는 저장 수단을 포함하며, 상기 웹 브라우저의 기동시, 네트웍을 통해 상기 사용자 컴퓨터 상에 상기 복수의 가상 이미지를 제공하는 서버를 포함하며,

상기 서버는, 상기 복수의 가상 이미지 중에서 상기 사용자에 의해 선택된 가상 이미지가 존재하는지를 판단하는 수단, 선택된 가상 이미지가 존재하면, 상기 선택된 가상 이미지에 대응하는 사용자 식별 데이터를 입력하도록 요구하고; 그렇치 않으면, 기설정된 가상 이미지를 제공하여 상기 기설정된 가상 이미지에 대응하는 사용자 식별 데이터를 입력하도록 요구하는 수단, 상기 선택된 가상 이미지가 등장하는 웹 브라우징 화면을 상기 사용자 컴퓨터 상에 제공하여, 상기

가상 이미지에 의한 네비게이션이 가능하도록 하는 수단, 상기 가상 이미지에 의한 네비게이션 정보를 실시간으로 분석하고 분석된 정보를 상기 저장 수단에 축적하는 수단 및 상기 웹 브라우저의 기동시, 상기 저장 수단에 축적된 상기 분석 정보에 기초하여 상기 사용자에 적합한 맞춤 정보를 서비스하는 수단을 포함하는 사용자 행동 패턴 분석 시스템이 제공된다.

이하, 본 발명의 바람직한 실시예를 첨부한 도면을 참조하여 상세히 설명한다.

도 1은, 본 발명에 따라 인터넷 등의 통신 네트웍을 통해 클라이언트와 웹 서버 간의 통신 체계를 도시한 도면이다.

주지하는 바와 같이, 다수의 클라이언트(즉, 사용자)는 통신 단말기(도시하지 않음) 또는 컴퓨터(1, 2,…,N)에 내장된 웹 브라우저를 기동시켜 인터넷과 같은 통신 네트웍(20)을 통해 웹 사이트(30)에 접속하여 정보를 검색하거나 교환한다.

본 발명의 바람직한 실시예에 따르면, 쿠키 파일 및 로그 파일을 이용하지 않고, 사용자의 고유 ID 번호, 이름 등의 기본 정보와 관련된 가상 이미지(예를 들면, 아바타)를 이용하여, 사용자가 웹 브라우저 상에서 액세스하는 사이트, 페이지, 페이지 뷰 회수, 이용 시간(즉, 머문 시간) 등의 정보를 입수하여, 입수된 정보를 체계적으로 분석하여 데이터 베이스화함으로써 사용자가 가장 흥미를 갖는 분야의 정보를 적응적으로 서비스할 수 있다.

상술하면, 사용자가 예를 들면, 컴퓨터(1)에 내장된 2차원 또는 3차원의 웹 브라우저를 기동시켜 네트웍(20)를 통해 웹 사이트(30)에 접속하면, 본 발명의 웹 서버(32)는 사용자 컴퓨터(1)에 가상 이미지 정보를 제공한다. 사용자는 웹 서버(32)로부터 제공된 가상 이미지 정보 중에서 원하는 이미지를 선택하고 자신의 고유 ID 번호를 입력하여 웹 서버(32)에 등록한다. 등록 후, 웹 서버(32)는 사용자가 선택한 가상 이미지가 등장한 2차원 또는 3차원의 웹 브라우징 화면을 네트웍(20)을 통해 제공한다. 이에 따라, 사용자는 통상의 방법으로 네트웍(20)을 통해 정보를 검색하거나 교환할 수 있다.

특히, 도 2에 도시한 바와 같이, 3차원의 웹 브라우저 환경에서의 가상 현실 게임 또는 채팅 등에서, 사용자 자신을 나타내기 위한 가상 이미지로서 아바타(avatar)라고 하는 그래픽 아이콘을 사용할 수 있다.

도 2는 3차원 웹 브라우저 환경에서 제공하는 화면의 일례를 도시한 도면이다. 전술한 바와 같이, 사용자가 웹서버(32)로부터 제공된 가상 이미지 정보 중에서 원하는 이미지를 선택하면, 웹 서버(32)는 도 2에 도시한 바와 같은 화면(11)을 제공한다. 웹 브라우징 화면(11)에서, (10)은 사용자가 선택한 아바타이며, (11a)~(11d)는 3차원 환경의 웹 브라우저에서 제공하는 지역(zone) 정보의 일례로서, 비지니스 존, 게임 존, 커뮤니티 존 및 쇼핑 존을 포함한다.

사용자는 마우스 등의 위치 결정 장치를 사용하여 아바타(10)의 이동을 제어하면서 웹 브라우징 화면 상에서 네비게이션한다. 예를 들면, 사용자가 마우스를 사용하여 아바타(10)를 쇼핑 존(11d)으로 이동시킬 경우, 본 발명의 시스템은 도 3에 도시한 바와 같이, 아바타(10)에 대응하는 사용자 ID, 아바타(10)의 위치 데이터(x, y, z), 아바타(10)가 이동한(또는, 방문한) 지역 데이터를 포함하는 일련의 정보를 사용자 행동 패턴 정보로서 네트웍(20)을 통해 웹 서버(32)에 실시간으로 전송한다. 통상, x, y, z 좌표의 위치 데이터는 사용자가 3차원 환경의 웹 브라우저 상에서 이동하는 경로를 추적함으로써 구해진다.

아바타(10)가 쇼핑 존(11d)에 입장하는 이벤트가 발생한 경우, 본 발명의 시스템은 아바타(10)의 쇼핑 존(11d)으로의 입장 이벤트 시각을 검출하고, 입장 이벤트 시각 데이터를 사용자 행동 정보에 추가하여 전송한다. 계속하여, 아바타(10)가 쇼핑 존(11d)에서의 특정 상품 코너에 접근하면, 특정 상품 코너에 대응하는 식별 데이터를 사용자 행동 정보에 추가하여 전송한다. 그 다음, 아바타(10)가 입장한 특정 상품 코너로부터 퇴장하면, 시스템은 퇴장 시각을 검출하여

사용자 행동 정보에 추가하여 전송한다.

이와 같이 하여, 본 발명의 시스템은 사용자가 아바타(10)를 이용하여 웹 브라우저 상에서 행동한 모든 정보(사용자 행동 정보)를 검출하고 이를 정리하여 사용자 행동 패턴 정보로서 웹 사이트(30)의 데이터베이스(31) 내에 축적한다. 축적된 정보는 도 4에 도시한 바와 같이, 사용자의 ID 별로 지역 데이터, 각 지역에 접근한 접근 회수 데이터 및 각 지역에 접근하여 해당 지역의 정보를 이용한 시간 데이터로 구분하여 데이터베이스화한다.

도 4는 데이터베이스(31) 내에 축적된 아바타(10)의 행동 패턴 정보를 일례로서 도시한 도면이다. 도 4로부터, 아바타(10)에 대응하는 사용자는 쇼핑 존(11d)을 100회 접속하여 50000000초동안 이용하여 쇼핑몰에 가장 큰 관심을 갖는다는 것을 알 수 있다. 그 다음으로, 사용자는 비지니스, 커뮤니티, 게임 순으로 관심이 있음을 알 수 있다.

도 5 및 도 6은 본 발명의 바람직한 실시예에 따라 웹 브라우징시 사용자의 행동 패턴을 분석하는 과정을 설명하기 위한 흐름도이다.

사용자가 예를 들면, 컴퓨터(1)에 내장된 3차원 환경의 웹 브라우저를 기동시키면(단계 S1), 웹 서버(32)는 데이터베이스(31)에 저장된 가상 이미지 정보 즉, 아바타 정보를 추출하여 네트웍(20)을 통해 사용자 컴퓨터(1)의 웹 브라우징 화면 상에 제공하여 사용자가 선택하도록 한다(단계 S2).

단계 S3에서, 제어 프로세스는 사용자가 제공된 아바타 정보 중에서 원하는 아바타를 선택하였는지를 판단한다. 단계 S3에서 아바타가 선택된 경우, 제어 프로세스는 사용자에게 자신의 고유 ID 번호를 입력하도록 요청하고 사용자 ID 번호가 입력되었으면(단계 S4), 선택된 아바타가 등장하는 3차원 웹 브라우징 화면을 컴퓨터(1) 상에 제공하여 네비게이션이 가능하도록 한다. 한편, 단계 S3에서 아바타가 선택되지 않은 경우, 제어 프로세스는 전술한 바와 같이 사용자에게 자신의 ID 번호를 입력하도록 요청하고 사용자 ID가 입력되었으면(단계 S6), 사용자에 대응하는 기본 아바타가 등장하는 3차원 웹 브라우징 화면을 컴퓨터(1) 상에 제공하여 네비게이션이 가능하도록 한다(단계 S8).

본 발명에 있어서, 3차원 환경의 웹 브라우징 화면 상에서 아바타가 네비게이션할 경우, 제어 프로세스는 도 3에 도시한 바와 같이 아바타의 위치 데이터(즉, x, y, z 좌표값)를 실시간으로 구하여 사용자의 ID 데이터와 함께 웹 서버(32)에 전송하며, 웹 서버(32)는 전송된 좌표값에 기초하여 아바타의 경로를 파악한다. 이렇게 파악된 경로는 데이터베이스(31)에 축적되며, 축적된 이동 경로 정보에 기초하여 사용자가 어느 경로를 통해 주로 이동하였는지를 분석하여 마케팅 정보로서 활용할 수 있다.

단계 S9에서, 제어 프로세스는 아바타(10)를 계속 모니터링하여 아바타(10)가 웹 브라우징 화면(11) 상에서 어느 지역으로 접근하는가를 판단한다. 본 발명에서는 접근한 지역이 쇼핑 존(11d)이라고 가정한다. 접근한 지역이 쇼핑 존(11d)인 경우, 제어 프로세스는 단계 S10에서 쇼핑 존(11d)에 대한 상세 정보를 제공함과 동시에, 지역 데이터로서의 쇼핑 존 데이터 및 쇼핑 존에 입장한 이벤트 시각을 검출한다. 사용자의 ID 데이터와 함께 검출된 지역 데이터 및 입장 이벤트 시각을 포함하는 정보는 실시간으로 네트웍(20)을 통해 웹 서버(32)에 전송되며, 전송된 정보는 데이터베이스(31)에 축적된다.

그 다음, 제어 프로세스는 접근 지역으로부터 사용자가 퇴장하였는가를 판단하고(단계 S12), 퇴장하였으면, 퇴장 이벤트 시각을 검출한다(단계 S13). 검출된 퇴장 이벤트 시각 데이터는 사용자 ID 데이터와 함께 네트웍(20)을 통해 웹 서버(32)에 전송되어 데이터베이스(31) 내에 축적된다(단계 S14).

본 발명의 웹 서버(32)는 데이터베이스(31)에 저장된 사용자 행동 정보, 즉, 사용자 ID, 지역 데이터, 입장 이벤트 시각 데이터, 퇴장 이벤트 시각 데이터에 기초하여, 사용자별로 접근한 지역 데이터, 해당 지역 데이터로의 접근 회수, 이용 시간을 자동으로 분석하여, 도 4에 도시한 바와 같은 일련의 정보를 사용자 행동 패턴 정보로서 데이터베이스(31)에 저장한다(단계 S15).

도 4에서 사용자는 다른 지역에 비해 쇼핑몰에 가장 큰 관심을 보인다는 것을 알 수 있다. 데이터베이스(31)에 저장된 사용자 행동 패턴 정보는 해당 사용자가 가장 관심을 갖는 분야에 대한 정보를 우선적으로 서비스하기 위한 맞춤 정보로서 이용된다. 즉, 사용자가 3차원 웹 브라우저를 기동할 경우, 본 발명의 웹 서버(32)는 해당 사용자에 대응하는 사용자 행동 패턴 정보를 데이터베이스(31)로부터 추출하여, 해당 사용자가 가장 관심을 갖는 분야에 대한 정보, 즉, 쇼핑 정보를 우선적으로 서비스할 수 있다(단계 S16).

또한, 사용자가 3차원 웹 브라우저를 기동하여 쇼핑 존(11d)으로 엑세스할 경우, 본 발명의 웹 서버(32)는 데이터베이스(31) 내에 저장된 해당 사용자에 대응하는 사용자 행동 패턴 정보에 기초하여, 쇼핑 존(11d) 내에서 해당 사용자가 가장 관심을 갖는 특정 상품에 대한 정보, 예를 들면, 컴퓨터 정보를 우선적으로 서비스할 수 있다(단계 S17). 도 4에 도시한 바와 같이, 사용자는 쇼핑 정보 다음에, 비즈니스, 커뮤니티, 게임 등의 순으로 관심을 갖고 있으므로, 본 발명의 웹 서버(32)는 사용자에게 제공되는 웹 브라우징 화면 정보의 비중을 상기 순서로 할당할 수 있음은 물론이다.

상기에 있어서, 본 발명의 바람직한 실시예에 대해서 설명하였는데, 본 발명의 특허 청구 범위를 이탈하지 않으면서 당업자는 다양한 변경을 행할 수 있음은 물론이다.

발명의 효과

따라서, 본 발명에 따르면, 인터넷을 통한 웹 브라우징시에, 사용자에 대응하는 가상 이미지를 이용하여 해당 가상 이미지의 네비게이션 정보를 분석하고, 분석된 정보를 사용자 행동 패턴 정보로서 데이터베이스화함으로써, 사용자가 가장 큰 관심을 갖는 분야의 정보를 다른 분야의 정보에 비하여 우선적으로 서비스할 수 있다.

또한, 본 발명에 따르면, 웹 브라우저 상에서 사용자의 ID 등의 기본 정보만을 갖는 가상 이미지를 이용하여 사용자에 관한 네비게이션 정보를 분석하므로, 사용자의 개인 프라이버시를 침해하지 않으면서 사용자의 행동 패턴을 효과적으로 분석할 수 있다.

또한, 본 발명에 따르면, 웹 브라우징시 사용자에 대한 네비게이션 정보를 웹 서버에 저장하여 관리하므로, 웹 서버의 용량에 따라 사용자 네비게이션 정보를 임의로 확장할 수 있다.

또한, 본 발명에 따르면, 3차원 웹 브라우징 화면 상에서 사용자에 대응하는 가상 이미지의 이동 경로를 분석함으로써, 가상 이미지가 주로 이동하는 경로 상에 특정한 정보를 할당하여 정보 이용의 효율을 극대화할 수 있다.

(57) 청구의 범위

청구항 1.

네트웍을 통한 웹 브라우징시에 발생하는 사용자의 네비게이션 정보를 분석하고, 분석 결과에 따라 해당 사용자에게 적합한 맞춤 정보를 서비스하는 시스템에 있어서,

웹 브라우저가 내장된 사용자 컴퓨터와,

복수의 가상 이미지를 저장하는 저장 수단을 포함하며, 상기 웹 브라우저의 기동시, 네트웍을 통해 상기 사용자 컴퓨터 상에 상기 복수의 가상 이미지를 제공하는 서버를 포함하며,

상기 서버는

상기 복수의 가상 이미지 중에서 상기 사용자에 의해 선택된 가상 이미지가 존재하는지를 판단하는 수단과,

선택된 가상 이미지가 존재하면, 상기 선택된 가상 이미지에 대응하는 사용자 식별 데이터를 입력하도록 요구하고; 그렇치 않으면, 기설정된 가상 이미지를 제공하고 상기 기설정된 가상 이미지에 대응하는 사용자 식별 데이터를 입력하도록 요구하는 수단과,

상기 선택된 가상 이미지가 등장하는 웹 브라우징 화면을 상기 사용자 컴퓨터 상에 제공하여, 상기 가상 이미지에 의한 네비게이션이 가능하도록 하는 수단과,

상기 가상 이미지에 의한 네비게이션 정보를 실시간으로 분석하고 분석된 정보를 저장 수단에 축적하는 분석 수단과,

상기 웹 브라우저의 기동시, 상기 저장 수단에 축적된 상기 분석 정보에 기초하여 상기 사용자에 적합한 맞춤 정보를 서비스하는 수단을 포함하는 사용자 행동 패턴 분석 시스템.

청구항 2.

제 1 항에 있어서,

상기 분석 수단은

상기 가상 이미지를 계속 모니터링하여 상기 가상 이미지가 상기 웹 브라우징 화면 상에서 어느 페이지로 접근하는가를 판단하는 수단과,

접근한 페이지가 존재하는 경우, 접근한 페이지의 데이터, 해당 페이지에 대한 입장 시각을 검출하는 수단과,

일련의 상기 사용자 식별 데이터, 상기 검출된 페이지 데이터 및 상기 검출된 입장 시각 데이터를 상기 네비게이션 정보로서 출력하는 수단을 포함하는 사용자 행동 패턴 분석 시스템.

청구항 3.

제 2 항에 있어서,

상기 분석 수단은

상기 가상 이미지가 상기 페이지로부터 퇴장한 시각을 검출하는 수단과,

일련의 상기 사용자 식별 데이터 및 상기 검출된 퇴장 시각 데이터를 상기 네비게이션 정보로서 출력하는 수단을 더 포함하는 사용자 행동 패턴 분석 시스템.

청구항 4.

제 3 항에 있어서,

상기 데이터 분석 수단은

상기 네비게이션 정보에 기초하여, 상기 접근 페이지에 대한 입장 횟수 및 이용 시간을 계산하는 수단을 더 포함하는 사용자 행동 패턴 분석 시스템.

청구항 5.

제 1 항 내지 제 3 항 중의 어느 한 항에 있어서,

상기 가상 이미지는 상기 사용자를 나타내는 그래픽 아이콘인 사용자 행동 패턴 분석 시스템.

청구항 6.

제 1 항에 있어서,

상기 서버는 복수의 지역으로 분할된 웹 브라우징 화면을 사용자 컴퓨터 상에 제공하고,

상기 분석 수단은

상기 가상 이미지를 계속 모니터링하여 상기 가상 이미지의 x, y 및 z축의 위치 데이터를 실시간으로 검출하는 수단과,

상기 가상 이미지가 웹 브라우징 화면의 상기 복수의 지역 중에서 어느 지역으로 접근하는가를 판단하는 수단과,

접근한 지역이 존재하는 경우, 접근한 지역의 데이터, 해당 지역에 대한 입장 시각을 검출하는 수단과,

일련의 상기 사용자 식별 데이터, 상기 검출된 위치 데이터 및 상기 검출된 입장 시각 데이터를 상기 네비게이션 정보로서 출력하는 수단을 포함하는 사용자 행동 패턴 분석 시스템.

청구항 7.

제 6 항에 있어서,

상기 분석 수단은

상기 가상 이미지가 상기 접근 지역으로부터 퇴장한 시각을 검출하는 수단과,

일련의 상기 사용자 식별 데이터, 상기 검출된 위치 데이터 및 상기 검출된 퇴장 시각 데이터를 상기 네비게이션 정보로서 출력하는 수단을 더 포함하는 사용자 행동 패턴 분석 시스템.

청구항 8.

제 7 항에 있어서,

상기 분석 수단은

상기 네비게이션 정보에 기초하여, 상기 접근 지역에 대한 입장 횟수 및 이용 시간을 계산하는 수단을 더 포함하는 사용자 행동 패턴 분석 시스템.

청구항 9.

제 5 항 내지 제 7 항 중의 어느 한 항에 있어서,

상기 가상 이미지는 상기 사용자를 나타내는 아바타(avatar)인 사용자 행동 패턴 분석 시스템.

청구항 10.

제 1 항 내지 제 3 항, 제 6 항, 제 7 항 중의 어느 한 항에 있어서,

상기 사용자 식별 데이터는 사용자 ID 번호 및/또는 이름을 포함하는 사용자 행동 패턴 분석 시스템.

청구항 11.

제 6 항에 있어서,

상기 분석 수단은

상기 검출된 위치 데이터에 기초하여, 상기 가상 이미지가 상기 웹 브라우징 화면 상에서 주로 이동하는 경로를 분석하는 수단과,

상기 분석된 이동 경로 정보를 상기 저장 수단에 축적하는 수단을 더 포함하는 사용자 행동 패턴 분석 시스템.

청구항 12.

제 11 항에 있어서,

상기 분석 수단은

상기 분석된 이동 경로 상에 상기 맞춤 정보를 제공하는 수단을 더 포함하는 사용자 행동 패턴 분석 시스템.

청구항 13.

복수의 지역(zone)으로 할당된 화면을 갖는 웹 브라우저가 내장된 사용자 컴퓨터와 복수의 가상 이미지를 저장하는 저장 수단을 구비하는 서버를 포함하는 시스템에서, 네트웍을 통한 웹 브라우징시에 발생하는 사용자의 네비게이션 정보를 분석하여 분석 결과에 따라 해당 사용자에게 적합한 맞춤 정보를 서비스하는 방법에 있어서,

(a) 웹 브라우저의 기동시, 네트웍을 통해 사용자 컴퓨터 상에 상기 복수의 가상 이미지를 제공하는 단계와,

(b) 상기 복수의 가상 이미지 중에서 상기 사용자에 의해 선택된 가상 이미지가 존재하는지를 판단하는 단계와,

(c) 선택된 가상 이미지가 존재하면, 상기 선택된 가상 이미지에 대응하는 사용자 식별 데이터를 입력하도록 요구하고 ; 그렇치 않으면, 기설정된 가상 이미지를 제공하여 상기 기설정된 가상 이미지에 대응하는 사용자 식별 데이터를 입력하도록 요구하는 단계와,

(d) 상기 선택된 가상 이미지가 등장하는 웹 브라우징 화면을 상기 사용자 컴퓨터 상에 제공하여, 상기 가상 이미지에 의한 네비게이션이 가능하도록 하는 단계와,

(e) 상기 가상 이미지에 의한 네비게이션 정보를 실시간으로 분석하고 분석된 정보를 상기 저장 수단에 축적하는 단계와,

(f) 상기 웹 브라우저의 기동시, 상기 저장 수단에 축적된 상기 분석 정보에 기초하여 상기 사용자에 적합한 맞춤 정보를 서비스하는 단계를 포함하는 사용자 행동 패턴 분석 방법.

청구항 14.

제 13 항에 있어서,

상기 단계 (e)는

(e1) 상기 가상 이미지를 계속 모니터링하여 상기 가상 이미지의 x, y 및 z축의 위치 데이터를 실시간으로 검출하는 단계와,

(e2) 상기 가상 이미지가 상기 웹 브라우징 화면의 상기 복수의 지역 중에서 어느 지역으로 접근하는가를 판단하는 단계와,

(e3) 접근한 지역이 존재하는 경우, 접근 지역 데이터, 해당 지역에 대한 입장 시각을 검출하는 단계와,

(e4) 일련의 상기 사용자 식별 데이터, 상기 검출된 위치 데이터 및 상기 검출된 입장 시각 데이터를 상기 네비게이션 정보로서 출력하는 단계를 포함하는 사용자 행동 패턴 분석 방법.

청구항 15.

제 14 항에 있어서,

상기 단계 (e)는

(e5) 상기 가상 이미지가 상기 접근 지역으로부터 퇴장한 시각을 검출하는 단계와,

(e6) 일련의 상기 사용자 식별 데이터, 상기 검출된 위치 데이터 및 상기 검출된 퇴장 시각 데이터를 상기 네비게이션 정보로서 출력하는 단계를 더 포함하는 사용자 행동 패턴 분석 방법.

청구항 16.

제 15 항에 있어서,

상기 단계 (e)는

(e7) 상기 네비게이션 정보에 기초하여, 상기 접근 지역에 대한 입장 횟수 및 이용 시간을 계산하는 수단을 더 포함하는 사용자 행동 패턴 분석 방법.

청구항 17.

제 13 항 내지 제 15 항 중의 어느 한 항에 있어서,

상기 가상 이미지는 상기 사용자를 나타내는 그래픽 아이콘인 사용자 행동 패턴 분석 방법.

청구항 18.

제 13 항 내지 제 15 항 중의 어느 한 항에 있어서,

상기 사용자 식별 데이터는 사용자 ID 번호 및/또는 이름을 포함하는 사용자 행동 패턴 분석 방법.

청구항 19.

제 13 항에 있어서,

상기 단계 (e)는

(e8) 상기 검출된 위치 데이터에 기초하여, 상기 가상 이미지가 상기 웹 브라우징 화면 상에서 주로 이동하는 경로를 분석하는 단계와,

(e9) 상기 분석된 이동 경로 정보를 상기 저장 수단에 축적하는 단계를 더 포함하는 사용자 행동 패턴 분석 방법.

청구항 20.

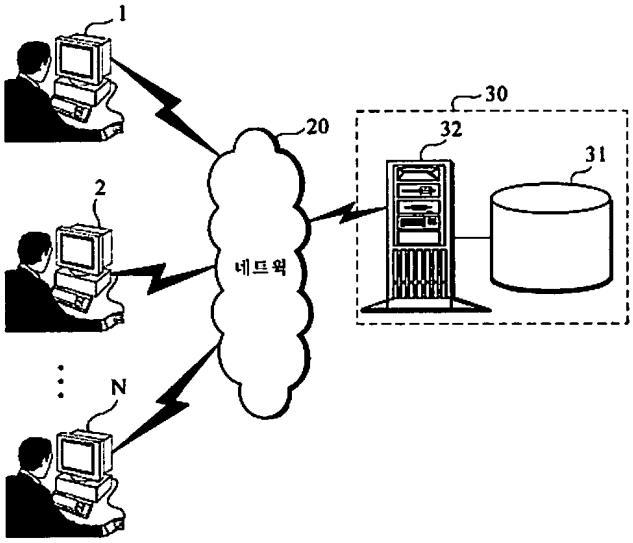
제 19 항에 있어서,

상기 단계 (e)는

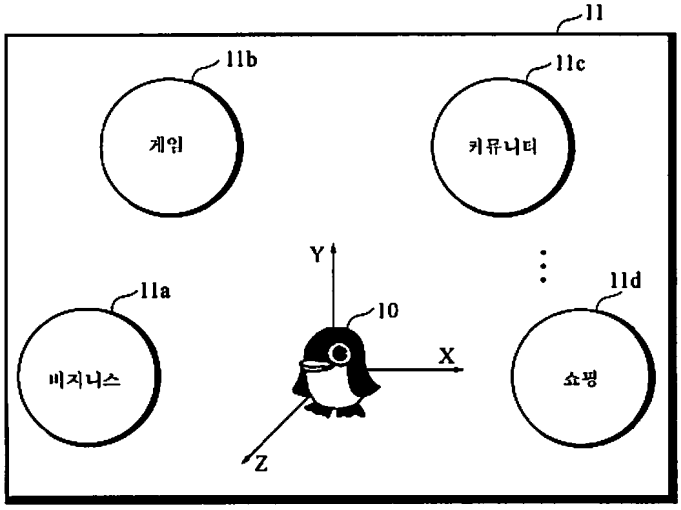
(e9) 상기 분석된 이동 경로 상에 상기 맞춤 정보를 제공하는 단계를 더 포함하는 사용자 행동 패턴 분석 방법.

도면

도면 1



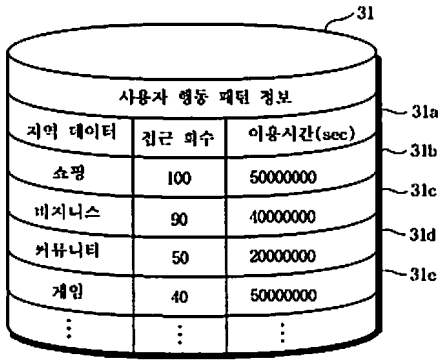
도면 2



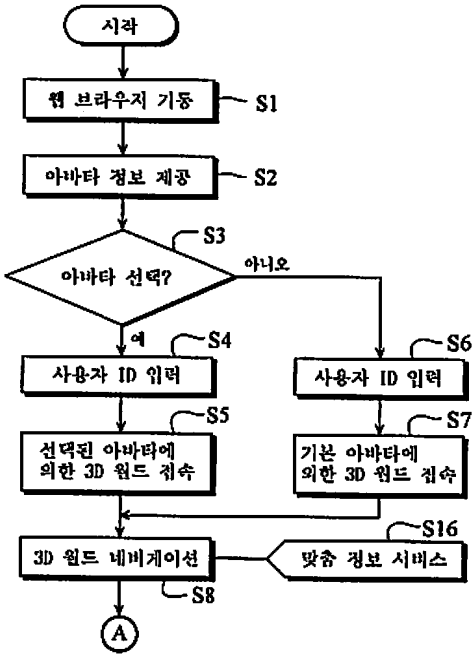
도면 3

사용자 행동 패턴 정보					
사용자 ID	위치 데이터 (X,Y,Z)	지역 데이터 (쇼핑 존)	입장 시각 데이터	상세 데이터 (컴퓨터)	퇴장 시각 데이터

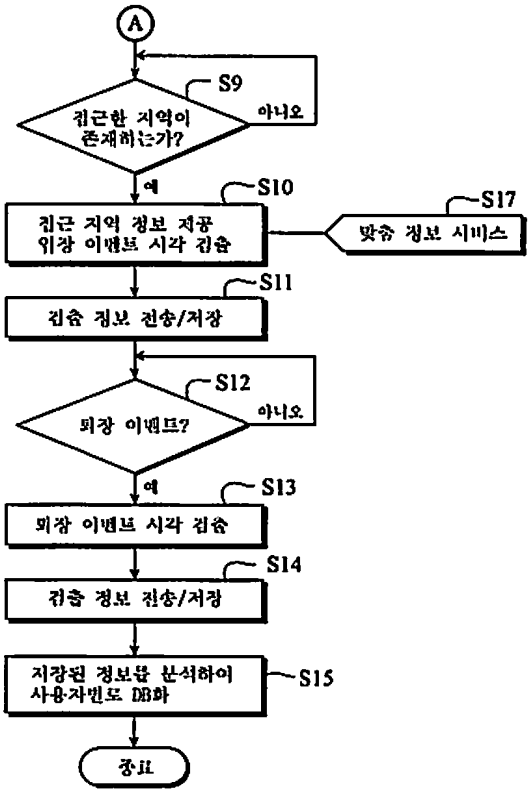
도면 4



도면 5


시작
웹 브라우저 기동
S1
아바타 정보 제공
S2
S3
아바타 선택?
아니오
예
S4
사용자 ID 입력
S5
선택된 아바타에
의한 3D 월드 접속
S6
사용자 ID 입력
S7
기본 아바타에
의한 3D 월드 접속
S16
3D 월드 네비게이션
맞춤 정보 서비스
S8
A

도면 6


A
S9
인근한 지역이 존재하는가?
아니오
예
S10
인근 지역 정보 제공 위장 이벤트 시각 검출
S17
맞춤 정보 시비스
S11
검출 정보 전송/저장
S12
위장 이벤트?
아니오
예
S13
위장 이벤트 시각 검출
S14
검출 정보 전송/저장
지장된 정보를 분석하여 사용자별로 DB화
S15
종료

Method of making Cordyceps sinensis wine with filtrate of cordyceps sinensis mycelium

Publication number: CN1274747 (A)
Publication date: 2000-11-29
Inventor(s): ZHANG XINLE [CN]; MENG NINGSHENG [CN]; LAI MAOLIN [CN]
Applicant(s): DONGXIN SCIENCE & TECHNOLOGY D [CN]
Classification:
- International: *C12G3/02;* C12G3/02; (IPC1-7): C12G3/02
- European:
Application number: CN20001012869 20000421
Priority number(s): CN20001012869 20000421

Abstract of CN 1274747 (A)

A cordyceps wine is made by using cordyceps mycelium filtrate, edible alcohol and active enzyme as raw material and through sealed fermentation at 20-30 deg.c for 25-40 days, the first filtering, sealed fermentation at 34-36 deg.c for other 25-40 days, the second filtering, blending, sealed storage for 80-110 days. The present invention changes cordyceps mycelium filtrate as one kind of waste into cordyceps wine, which contains 18 kinds of amino acids, trace elements, vitamins and other active cordyceps components and has the health functions of tonifying lung, invigorating kidney, raising body's disease-resisting immunity, etc.

[19]中华人民共和国国家知识产权局

[51]Int. Cl[7]
C12G 3/02

[12]发明专利申请公开说明书

[21]申请号 00112869.8

[43]公开日 2000年11月29日

[11]公开号 CN 1274747A

[22]申请日 2000.4.21 [21]申请号 00112869.8
[71]申请人 绵阳市东信科技开发有限公司
地址 621000 四川省绵阳市西河东路9号
[72]发明人 张新乐 孟宁生 赖茂林 姚忠良
[74]专利代理机构 绵阳市专利事务所
代理人 杨荫茂

权利要求书1页 说明书4页 附图页数1页

[54]发明名称 虫草菌丝体滤液制造虫草饮用酒的方法

[57]摘要

一种虫草菌丝体滤液制造虫草饮用酒的方法，其特征是包括：滤液与食用酒精或粮食酒和活性酶在20～30℃下密封发酵25～40天，过滤，滤液与活性酶和食用酒精或食用酒在34～36℃下密封发酵25～40天，过滤，调配，密封储存80～110天等步骤。本发明将视为废液的虫草菌丝体滤液变成富含18种氨基酸、多种虫草有效成分、多种微量元素及维生素等，具有益肺补肾、提高人体抗病免疫力等营养保健功效、色香味俱全的虫草饮用酒，变废为宝，生产工艺简单，成本低，利于环境保护。

ISSN1008-4274

知识产权出版社出版

权 利 要 求 书

1．一种虫草菌丝体滤液制造虫草饮用酒的方法，其特征是包括下列步骤：

a．发酵：将虫草菌丝体滤液过滤，取滤液，与食用酒精或粮食酒和适量活性酶混合均匀，置于容器中在20~30℃下密封发酵25~40天；

b．过滤：将发酵后的混合物经过滤器过滤，弃去滤渣，滤液待用；

c．发酵：取适量活性酶和食用酒精或粮食酒加入滤液中混合均匀，置于容器中在34~36℃下密封发酵25~40天；

d．过 滤：将发酵后的混合物经过滤器过滤，弃取滤渣，滤液待用；

e．调配：取食用酒精或粮食酒或/和无盐水，加入滤液中，使混合液中乙醇含量为12%~46%（体积）；

f．贮存：将混合液置于容器内密封储存80~110天。

2．按权利要求 1 所述的虫草菌丝体滤液制造虫草饮用酒的方法，其特征是还包括下列步骤：

g．勾兑：根据成品的色香味需要，将芳香物质、有机酸或/和有机酯加入贮存后的混合液中混合均匀；

h．精过滤：将勾兑过的混合液经精过滤器过滤，弃取滤渣，滤液即为虫草饮用酒产品；

i．滤液经检验后罐装或/和分装为成品。

3．按权利要求 1 所述的虫草菌丝体滤液制造虫草饮用酒的方法，其特征是：所述步骤a和步骤c中的混合液中，乙醇含量为12%~46%（体积）。

4．按权利要求 1、2 或 3 所述的虫草菌丝体滤液制造虫草饮用酒的方法，其特征是：所述步骤a中活性酶为水解酶。

5．按权利要求 1、2 或 3 所述的虫草菌丝体滤液制造虫草饮用酒的方法，其特征是：所述步骤c中的活性酶为分解蛋白酶。

6．按权利要求 5 所述的虫草菌丝体滤液制造虫草饮用酒的方法，其特征是：所述步骤c中的活性酶为分解蛋白酶。

7．按权利要求 1、2、3 或 6 所述的虫草菌丝体滤液制造虫草饮用酒的方法，其特征是：所述步骤d中过滤器内置有硅藻土或/和活性炭。

8．按权利要求 2 所述的虫草菌丝体滤液制造虫草饮用酒的方法，其特征是：所述步骤h的精过滤器内置有硅藻土、活性炭或/和高分子树脂。

9．按权利要求 1、2 或 3 所述的虫草菌丝体滤液制造虫草饮用酒的方法，其特征是：所述容器、过滤器和精过滤器均为不锈钢材料制成。

说　明　书

虫草菌丝体滤液制造虫草饮用酒的方法

本发明属于含有来源于植物的材料等的其它酒精饮料的制备，涉及一种虫草菌丝体滤液制造虫草饮用酒的方法。

冬虫夏草，作为集营养与药效于一体的保健佳品，被受到普遍关注和利用。现有的虫草饮用酒，均采用虫草子实体或虫草菌丝体浸泡于酒中勾兑而成，例如中国专利申请号为94101327提供的“虫草保健酒及其制作方法”等等，这种方式不仅需要大量珍稀独特的冬虫夏草资源，而且饮用后人体吸收虫草中有效成份较少。现有技术中，虫草菌丝体的制备，较多的是将虫草植入液体培养基中经培养、发酵而成，例如申请号为93104709、公开号为CN1095103公开的“虫草菌丝体生产工艺”等，液体培养基由大部份水与少量碳原、氮原、微量元素、维生素等（例如：葡萄糖、蛋白胨、奶粉、蚕蛹、磷酸氢二钠、磷酸二氢钾等）混合而成，虫草菌丝发酵完成后，液体（滤液）弃去；由于滤液被弃去，不仅使滤液中的氨基酸、微量元素、虫草酸、虫草多糖等多种易被人体吸收的有效成份得不到利用，而且造成土壤、地表水、地下水和大气（受污染后所形成腐植质释放大量臭气）严重污染，不利于环境保护。

本发明之目的旨在克服上述现有技术的不足，通过将滤液与乙醇、活性酶混合发酵等方式，从而提供一种变废为宝、低成本的虫草菌丝体滤液制造虫草饮用酒的方法。

本发明的内容是：一种虫草菌丝体滤液制造虫草饮用酒的方法，其特征之处是包括下列步骤：

a. 发酵：将虫草菌丝体滤液过滤，取滤液，与食用酒精或粮食酒和适量活性酶混合均匀，置于容器中在20~30℃下密封发酵25~40天，充分发生生物化学反应，并使蛋白质与水溶合，增加渗透压，抑制杂菌生长；

b. 过滤：将发酵后的混合物经过滤器过滤，弃去滤渣，滤液待用；

c. 发酵：取适量活性酶和食用酒精或粮食酒加入滤液中混合均匀，置于容器中在34~36℃下密封发酵25~40天，充分发生生物化学反应，并使溶于水的蛋白质分解成氨基酸，抑制

杂菌生长；

d. 过 滤：将发酵后的混合物经过滤器过滤，弃取滤渣，滤液待用；

e. 调配：取食用酒精或粮食酒或/和无盐水（即已除去无机离子、细菌的净化水），加入滤液中，使混合液中乙醇含量为 12%~46%（体积），根据市场需要确定混合液中乙醇含量；

f. 贮存：将混合液置于容器内储存 80~110 天。

本发明内容中还包括下列步骤：

g. 勾兑：根据成品的色香味需要，将芳香物质、有机酸或/和有机酯加入贮存后的混合液中混合均匀；

h. 精过滤：将勾兑过的混合液经精过滤器过滤，弃取滤渣，滤液即为虫草饮用酒产品；

i. 滤液经检验后罐装或/和分装为成品。

本发明内容中：所述步骤 a 和步骤 c 中的混合液中，乙醇含量为 12%~46%（体积）。

本发明内容中：所述步骤 a 中活性酶为水解酶。

本发明内容中：所述步骤 c 中的活性酶为分解蛋白酶。

本发明内容中：所述步骤 d 中过滤器内置有硅藻土或/和活性炭。

本发明内容中：所述步骤 h 的精过滤器内置有硅藻土、活性炭或/和高分子树脂。

本发明内容中：所述容器、过滤器和精过滤器均为不锈钢材料制成。

本发明具有下列特点：

（1） 针对虫草菌丝体滤液中含有虫草酸等多种虫草浸出物、多种蛋白质、淀粉等高分子物质以及微量元素、维生素等状况，在无杂菌污染的条件下，采用以生物技术为主、兼以生物化学技术、生物物理技术，对滤液进行处理，经两次发酵、过滤、贮存、调配及勾兑等工序，使滤液中有效成份被分解、稳定、保留和利用，滤渣、悬浮物、无效成份被分离、除去，从而将滤液开发生产成兼备保健营养功效，益肺补肾、提高人体抗病免疫力、色香味俱全的虫草饮用酒，增加了饮用酒的种类，且生产工艺简单，成本低；

（2） 采用多种活性酶使滤液中不稳定的蛋白质分解成较稳定的氨基酸，除去不稳定和有害物质，将虫草菌丝体滤液变成富含 18 种氨基酸（门冬氨酸、苏氨酸、丝氨酸、谷氨酸、

甘氨酸、丙氨酸、胱氨酸、缬氨酸、蛋氨酸、异亮氨酸、亮氨酸、酪氨酸、苯丙氨酸、赖氨酸、氨、组氨酸、精氨酸、脯氨酸)、虫草酸等多种虫草中有效成份，维生素、锌、锶、碘、硒等多种微量元素的高级营养保健酒，从而使大量视为废液、利用技术难度大、贮存难的虫草菌丝体滤液得到综合利用，变废为宝；

（3） 节省了大量虫草及其菌丝体资源，避免了虫草菌丝体滤液排放而造成的土壤、地上水、地下水和大气的污染，利于环境保护，实用性强，具有较大的推广价值，经济效益和社会效益显著。

图 1 是本发明的生产工艺流程示意图。

下面通过实施例对本发明作进一步描述：

实施例 1。

一种虫草菌丝体滤液制造虫草饮用酒的方法，包括下列步骤：

a. 发酵：将虫草菌丝体滤液过滤，取滤液，与食用酒精或粮食酒和适量水解酶混合均匀，并使混合液中的乙醇含量为30%（体积）。置于容器中在 25℃下密封发酵 30 天，充分发生生物化学反应，并使蛋白质与水溶合，增加渗透压，抑制杂菌生长；

b. 过滤：将发酵后的混合物经过滤器过滤，弃去滤渣，滤液待用；

c. 发酵：取适量分解蛋白酶和食用酒精或粮食酒加入滤液中混合均匀，并使混合液中的乙醇含量为 30%（体积）。置于容器中在 35℃下密封发酵 30 天，充分发生生物化学反应，并使溶于水的蛋白质分解成氨基酸，抑制杂菌生长；

d. 过 滤：将发酵后的混合物经过滤器（过滤器内置有硅藻土或/和活性炭）过滤，滤除悬浮物、碳酸盐物质、杂醇油、色素、高级脂肪酸等物质，弃取滤渣，滤液待用；

e. 调配：取食用酒精或粮食酒或/和无盐水，加入滤液中，使混合液中乙醇含量为 32%（体积），或根据市场需要，调整乙醇含量；

f. 贮存：将混合液置于容器内密封储存 90 天，使酒体老熟；

g. 勾兑：根据成品的色香味需要，将芳香物质、有机酸或/和有机酯加入贮存后的混合液中混合均匀；

h. 精过滤：将勾兑过的混合液经精过滤器（精过滤器内置有硅藻土、活性炭或/和高分子树脂）过滤，除去异味异色物质、残余高级脂肪酸、杂醇油等，弃取滤渣，滤液即为虫草饮用酒产品；

i. 滤液经检验后罐装或/和分装为成品。

上述各步骤中，所述容器、过滤器和精过滤器均为不锈钢材料制成。

实施例 2~7。

一种虫草菌丝体滤液制造虫草饮用酒的方法，其生产工艺步骤同实施例 1，略；各步骤中工艺条件和参数见下表：

实施例编号 / 工艺条件和参数 / 步骤	2	3	4	5	6	7
a发酵	20℃发酵40天乙醇含量46%	30℃发酵25天乙醇含量12%	28℃发酵28天乙醇含量20%	26℃发酵30天乙醇含量32%	24℃发酵33天乙醇含量42%	22℃发酵38天乙醇含量45%
c发酵	34℃发酵40天乙醇含量46%	36℃发酵25天乙醇含量12%	35℃发酵28天乙醇含量20%	34℃发酵30天乙醇含量32%	35℃发酵33天乙醇含量42%	36℃发酵38天乙醇含量45%
e调配	乙醇含量46%	乙醇含量12%	乙醇含量20%	乙醇含量32%	乙醇含量42%	乙醇含量45%
f贮存	110天	80天	90天	100天	105天	95天

注：①表未列出的步骤中工艺条件和参数同实施例 1；

②表中各步骤中，未写出的组分、工艺条件和参数同实施例 1；

③步骤 e 调配中，混合液中乙醇含量可根据市场需要调整。

本发明不限于上述实施例，本发明内容所述均可实施，并具有较好效果。

说 明 书 附 图

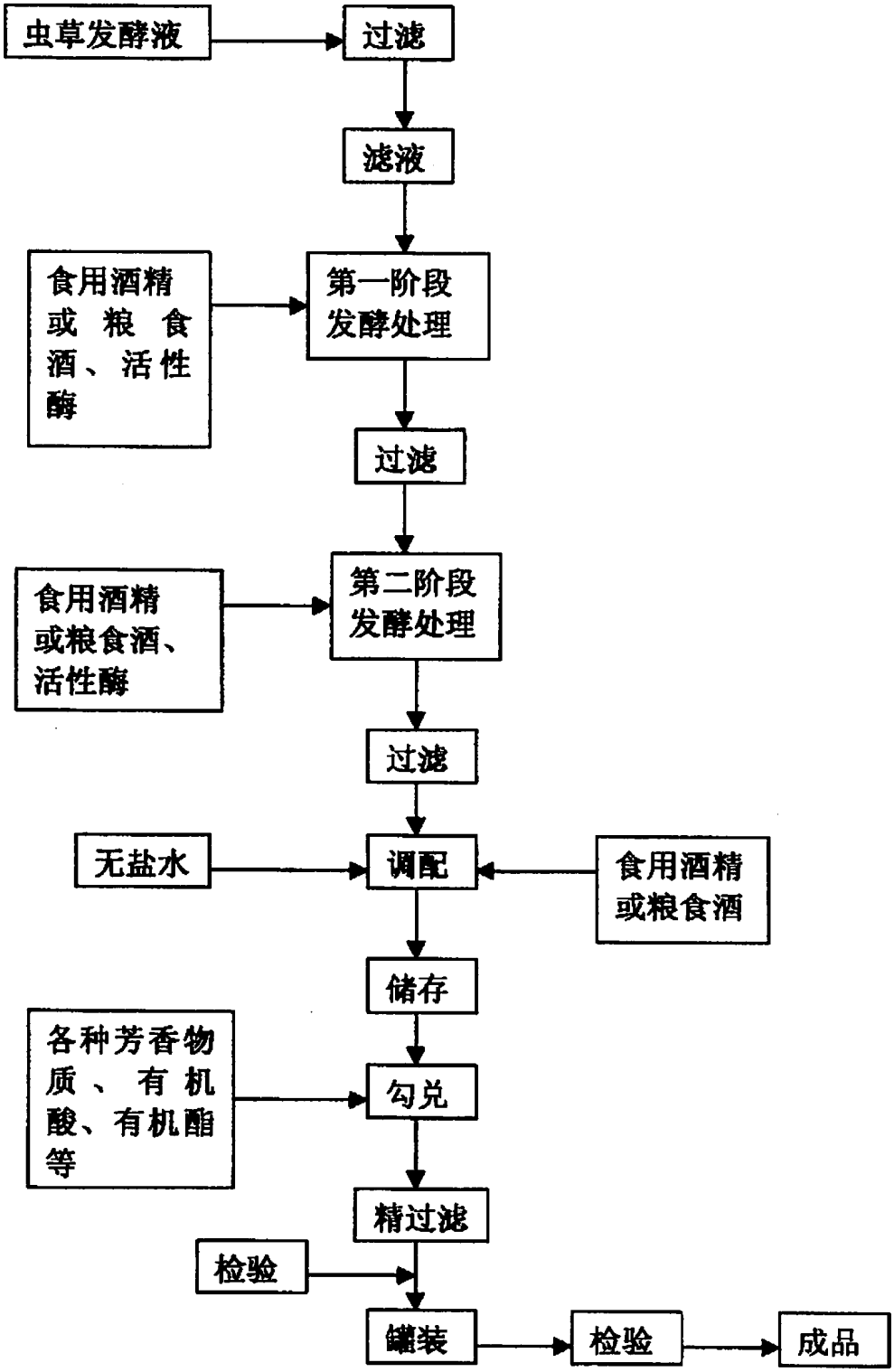

虫草发酵液
过滤
滤液
食用酒精或粮食酒、活性酶
第一阶段发酵处理
过滤
食用酒精或粮食酒、活性酶
第二阶段发酵处理
过滤
无盐水
调配
食用酒精或粮食酒
储存
各种芳香物质、有机酸、有机酯等
勾兑
精过滤
检验
罐装
检验
成品

图 1

Electronic Patent Application Fee Transmittal

Application Number:	10743476
Filing Date:	23-Dec-2003
Title of Invention:	Proactive user interface
First Named Inventor/Applicant Name:	Jong-Goo Lee
Filer:	Paul J. Farrell/Deanna Marman
Attorney Docket Number:	678-1264

Filed as Large Entity

Utility under 35 USC 111(a) Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Petition fee- 37 CFR 1.17(h) (Group III)	1464	1	130	130
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
Submission- Information Disclosure Stmt	1806	1	180	180
		Total in USD ($)		**310**

Electronic Acknowledgement Receipt

EFS ID:	4538166
Application Number:	10743476
International Application Number:	
Confirmation Number:	9615
Title of Invention:	Proactive user interface
First Named Inventor/Applicant Name:	Jong-Goo Lee
Customer Number:	66547
Filer:	Paul J. Farrell/Deanna Marman
Filer Authorized By:	Paul J. Farrell
Attorney Docket Number:	678-1264
Receipt Date:	30-DEC-2008
Filing Date:	23-DEC-2003
Time Stamp:	16:26:23
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Credit Card
Payment was successfully received in RAM	$310
RAM confirmation Number	1769
Deposit Account	504053
Authorized User	FARRELL,PAUL J.

The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows:

Charge any Additional Fees required under 37 C.F.R. Section 1.17 (Patent application and reexamination processing fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.20 (Post Issuance fees)

Charge any Additional Fees required under 37 C.F.R. Section 1.21 (Miscellaneous fees and charges)					
File Listing:					
Document Number	**Document Description**	**File Name**	**File Size(Bytes)/ Message Digest**	**Multi Part /.zip**	**Pages (if appl.)**
1	Information Disclosure Statement Letter	Supplemental_IDS.pdf	80790 b44d46797a0026221bda62d8b846446991e6cb52	no	3
Warnings:					
Information:					
2	Petition for review by the Office of Petitions.	Petition.pdf	77890 7e1313222ac68736fe7805a6875759c62d36186e	no	2
Warnings:					
Information:					
3	NPL Documents	RU_OA.pdf	252234 4de9d1d39af8fa7d3c4d0f3bba8ddcc668da6f45	no	7
Warnings:					
Information:					
4	Foreign Reference	KR_1020020024824.pdf	505585 0b6668043834d4797dffcc780942bcb9a16ea801	no	15
Warnings:					
Information:					
5	Foreign Reference	CN_1274747.pdf	352194 708b0f6a6c5d3f72ea3627f54cbb55a6fbb85519	no	8
Warnings:					
Information:					
6	Fee Worksheet (PTO-06)	fee-info.pdf	31446 e110f5f4af86986834c1d790387fc05bf51e7f01	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	1300139		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Jong-Goo LEE et al.

Serial No.: 10/743,476 Docket: 678-1264

Filed: December 23, 2003 Dated: December 30, 2008

For: PROACTIVE USER INTERFACE

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT

Sir:

Pursuant to Applicant's continuing duty of disclosure, it is respectfully requested that the references listed in the attached form PTO-1449 be considered by the Examiner and made of record in the above-identified application. A copy of each reference is attached hereto.

The citation of the listed items is not a representation that they constitute a complete or exhaustive listing of the relevant art or that the references are prior art. The items listed are submitted in good faith, but are not intended to substitute for the Examiner's search. It is hoped, however, that in addition to apprising the Examiner of these particular items, they will assist in identifying fields of search and in making as full and complete a search as possible.

The listed items were cited by the Russian Patent Office in a counterpart application, namely Appln. No. 2006110932. A copy of the Russian Office Action dated September 30, 2008 is attached hereto.

The filing of this Supplemental Information Disclosure Statement is not an admission that the information cited herein is, or is considered to be, material to patentability as defined in 37 C.F.R. § 1.56(b).

The claims of the application as now presented are believed to patentably distinguish over the prior art and to be in condition for allowance. Early and favorable consideration of the case is respectfully requested.

CERTIFICATION UNDER 37 C.F.R. §1.97(e)(2)

Applicant submits that each item of information contained in the Supplemental Information Disclosure Statement was cited in a communication from a foreign patent office in a counterpart foreign application no more than three months prior to the filing of the Statement.

Respectfully submitted,

Paul J. Farrell
Reg. No. 33,494
Attorney for Applicant

THE FARRELL LAW FIRM
333 Earle Ovington Blvd., Suite 701
Uniondale, NY 11553
(516) 228-3565

Form PTO-1449 U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE	ATTY. DOCKET NO. 678-1264	SERIAL NO. 10/743,476
INFORMATION DISCLOSURE STATEMENT BY APPLICANT (Use several sheets if necessary)	APPLICANT Jong-Goo LEE et al.	
	FILING DATE December 23, 2003	GROUP ART UNIT

U.S. PATENT DOCUMENTS

EXAMINER INITIAL		DOCUMENT NUMBER	DATE	NAME	CLASS	SUBCLASS	FILING DATE IF APPROPRIATE

FOREIGN PATENT DOCUMENTS

		DOCUMENT NUMBER	DATE	COUNTRY	CLASS	SUBCLASS	TRANSLATION YES	TRANSLATION NO
		KR 1020020024824	April 3, 2002	Korea			X	
		CN 1274747	November 29, 2000	China			X	

OTHER PRIOR ART (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER	DATE CONSIDERED

* EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw citation if not in conformance and not considered. Include copy of this form with next communication to ap

DOCKET: **678-1264 (P11404)**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

APPLICANT(S): LEE, Jong-Goo, et al. **ART UNIT:** 2179

APPLICATION NO.: 10/743,476 **EXAMINER:** THERIAULT, Steven B.

FILED: December 23, 2003 **DATED:** December 8, 2008

FOR: **PROACTIVE USER INTERFACE**

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPELLANT'S REPLY BRIEF

Sir:

In response to the Examiner's Answer having a mailing date of October 7, 2008, Appellants respectfully submit that based on at least the arguments provided in the Appeal Brief of July 7, 2008, Claims 1-7 and 109-141 are patentable over the applied reference. The following comments are respectfully submitted to address statements made in the Examiner's Answer.

The allegations raised in the Examiner's Answer do not change the fact that Claims 1-7 and 109-141 are patentable over U.S. Patent No. 6,400,996 to Hoffberg et al. (herein "Hoffberg").

1. The Alternative Embodiments of Hoffberg Cited in the Examiner's Answer Fail to Anticipate the Pending Claims

The Examiner's prior analysis relied upon Fig. 15 and 16 of the 143-page Hoffberg reference, aside from an extensive listing of patents that Hoffberg 'incorporates by reference,' which is discussed in Section 2 below. The Examiner's Answer adds an analysis of embodiments discussed at Col. 97-98 and at Col. 100, lines 1-21. Like the originally cited embodiment of Figs. 15-16 of Hoffberg, these new embodiments also fail to disclose *proactively altering at least one function of said interface unit according to said detected pattern*, and accordingly do not anticipate any of Claims 1-7 and 109-141 under 35 U.S.C. § 102(b).

Fig. 15 of Hoffberg discloses a process to "eliminate extra keypresses" by providing to the user "[f]requently used choices for program selection." (Hoffberg, Col. 85, lines 16 and 18-19.) The Examiner incorrectly alleges that a "skilled artisan would determine that a reprogramming of the interface is an altering of at least one function of the interface" to be taken "[i]n combination with cited features of tracking and modifying the interface." (Examiner's Answer, bottom of page 7.) The Examiner also incorrectly argues that a "'behavior of use' pattern" allegedly disclosed in Figs. 15-16 of Hoffberg can be taken "[i]n combination with the description of the interface example in column 97-98" of Hoffberg (Examiner's Answer, bottom of page 7 to top of page 8).

To the contrary, "Example 5" of Hoffberg functions "[b]y determining the skill of the user" (Hoffberg, Col. 97, lines 65-66) and accordingly differs from the embodiment(s) of Figs. 15-16. Moreover, the "Example 5" discussed at the newly cited Col. 97-98 of Hoffberg is not expressly related to the example of Figs. 15-16, and the Examiner improperly seeks to combine such alleged disclosure from different embodiments in an anticipation rejection. The need for the

Examiner to cite various aspects of different embodiments highlights the Examiner's inability to satisfy the requirement of showing how Hoffberg describes all of the elements of the claimed invention, arranged as in the pending claims. See, *Net MoneyIN, Inc. v. VeriSign, Inc.*, 2008 U.S. App. LEXIS 21827, *22 (2007-1565 Oct. 20, 2008), in which the Federal Circuit "reemphasized the importance of the requirement that the reference describe not only the elements of the claimed invention, but also that it describe those elements 'arranged as in the claim'," citing *Connell v. Sears, Roebuck & Co.*, 722 F.2d 1542, 1548 (Fed. Cir. 1983).[1]

For at least the above reasons, the rejection must be withdrawn.

2. The Examiner Incorrectly Cites More than One Reference In the Alleged Anticipation Rejection

The Examiner alleges that since "the rejection of claim does not refer to any ***specific*** patent or reference other then [sic: than] Hoffberg" it is appropriate to base the rejection on Col. 42, lines 20-67, which consists of a listing of fifty-one issued patents that Hoffberg seeks to incorporate by reference. (Examiner's Answer, page 11, emphasis supplied.) The rationale advanced by the Examiner is that this listing of other patents is properly considered in this anticipation rejection since the other patents are only listed "**as a demonstration** to the state of the art and what was know at the time of the filing of Hoffberg." (Examiner's Answer, page 11, emphasis in original.) To the contrary, MPEP 2131.01, which addresses citation of multiple references in anticipation rejections, does not include a 'state of the art' exception.

For this additional reason, the rejection must be withdrawn.

[1] A printout of the recent *Net MoneyIN, Inc. v. VeriSign, Inc.* decision is enclosed for the Examiner's convenience.

3. Conclusion

Appellant has shown that there are claimed features not disclosed by the cited reference, and has thus shown that Claims 1-7 and 109-141 have been erroneously rejected under 35 U.S.C. §102(b). For at least this reason the Examiner has not established a *prima facie* showing of anticipation and the rejection must be withdrawn.

Dated: December 4, 2008

Respectfully submitted,

By: _____

Paul J. Farrell
Reg. No.: 33,494
Attorney for Applicant(s)

THE FARRELL LAW FIRM, P.C.
333 Earle Ovington Blvd., Suite 701
Uniondale, New York 11553
(516) 228-3565 (tel)

Enclosure: *Net MoneyIN, Inc. v. VeriSign, Inc.*, 2008 U.S. App. LEXIS 21827 (Fed. Cir. 2007-1565 Oct. 20, 2008).

Electronic Acknowledgement Receipt

EFS ID:	4413314
Application Number:	10743476
International Application Number:	
Confirmation Number:	9615
Title of Invention:	Proactive user interface
First Named Inventor/Applicant Name:	Jong-Goo Lee
Customer Number:	66547
Filer:	Paul J. Farrell/Deanna Marman
Filer Authorized By:	Paul J. Farrell
Attorney Docket Number:	678-1264
Receipt Date:	08-DEC-2008
Filing Date:	23-DEC-2003
Time Stamp:	15:38:38
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
1	Reply Brief Filed	Reply_Brief.pdf	131692 f0d3d36e41074c2e3b563dd5c560aa970fbd04ee	no	4

Warnings:	
Information:	

2	NPL Documents	NET_MONEYIN__INC_V_VERISIG.pdf	739460 e9c121c75e4840c1c93eff05ee65c2ec48c7c949	no	9

Warnings:	
Information:	
Total Files Size (in bytes):	871152

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/743,476	12/23/2003	Jong-Goo Lee	678-1264	9615

66547 7590 10/07/2008

THE FARRELL LAW FIRM, P.C.
333 EARLE OVINGTON BOULEVARD
SUITE 701
UNIONDALE, NY 11553

EXAMINER	
THERIAULT, STEVEN B	
ART UNIT	**PAPER NUMBER**
2179	

MAIL DATE	DELIVERY MODE
10/07/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/743,476
Filing Date: December 23, 2003
Appellant(s): LEE ET AL.

Paul J. Farrell
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 07/07/2008 appealing from the Office action mailed 02/07/2008.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,400,996	Hoffberg	6-2002

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-7, 109-141 are rejected under 35 U.S.C. 102(b) as being anticipated by Hoffberg et al. (hereinafter Hoffberg) U.S. Patent No. 6400996 issued June 4, 2002.

In regard to claims 1-7 and 109-121, claims 1-7 and 109-121 reflect the interface comprising computer readable instructions for performing the method steps of claims 122-141, respectively, and are rejected along the same rationale.

In regard to **Independent claim 122,** Hoffberg teaches a method for a proactive interaction between a user and a computational device through a user interface, the computational device having an operating system, the method comprising:

- Detecting a pattern of user behavior according to at least one interaction of the user with the user interface by using a learning module (See column 10, lines 15-31 and several incorporate pattern recognition patents (See column 42, lines 20-67 and column 50, lines 50-67)).
- Proactively altering at least one function of the user interface according to said pattern (See Figure 15, and column 85, lines 5-67). Hoffberg teaches an interface that the user interacts with and that the system modifies based on the detected input pattern

With respect to **dependent claim 123,** Hoffberg teaches the method wherein said at least one pattern is selected from the group consisting of a pattern determined according to at least one previous interaction of the user with said user interface, and a predetermined pattern, or a combination thereof (See column 51, lines 7-15).

With respect to **dependent claim 124,** Hoffberg teaches the method wherein said user interface features a graphical display and said altering at least one function of said user interface comprises altering at least a portion of said graphical display (See Figure 15 and column 51, lines 55-67 and column 52, lines 35-45).

With respect to **dependent claim 125,** Hoffberg teaches the method wherein said altering at least a portion of said graphical display comprises:

selecting a menu for display according to said detected pattern; and displaying said menu (See column 144, lines 25-42 and Figure 15 and column 90, lines 20-40).

With respect to **dependent claim 126,** Hoffberg teaches the method wherein said selecting said menu comprises: constructing a menu from a plurality of menu options (See column 116, lines 15-67 and Figures 15-18).

With respect to **dependent claim 127,** Hoffberg teaches the method wherein said user interface

features an audio display and said altering at least one function of said user interface comprises altering at least one audible sound produced by the computational device (See column 60, lines 59-67, column 94, lines 43-65 and column 119, lines 20-67).

With respect to **dependent claim 128,** Hoffberg teaches the method wherein the computational device is selected from the group consisting of a regular computer, an ATM, a cellular telephone, a mobile information device, a PDA, or a consumer appliance having an operating system (See column 51, lines 40-45, column 39, lines 35-45 and column 147, lines 30-35).

With respect to **dependent claim 129,** Hoffberg teaches the method wherein said learning module comprises a knowledge base, and the method further comprises holding information gathered as a result of interactions with the user and/or the operating system by using said knowledge base (See Figure 18, 1807 and column 117, lines 20-67).

With respect to **dependent claim 130,** Hoffberg teaches the method wherein said knowledge base comprises a plurality of integrated knowledge determined from the behavior of the user and from preprogrammed information (See column 56, lines 40-51).

With respect to **dependent claim 131,** Hoffberg teaches the method wherein said learning module further comprises a plurality of sensors, and uses said sensors to perceive a state of the operating system (See column 99, lines 1-15 and 40-55).

With respect to **dependent claim 132,** Hoffberg teaches the method wherein said learning module further comprises a perception unit, and uses said perception unit to process output from said sensors and determine a state of the operating system and a state of said user interface (See figures 15-18 and column 50, lines 50-67 and column 125, lines 30-67).

With respect to **dependent claim 133,** Hoffberg teaches the method wherein said learning module further comprises a reasoning system, and uses said reasoning system to update said knowledge base and learn an association between an alteration of said user interface and a state of the operating system (See column 126, lines 44-67 and Example 12, column 119).

With respect to **dependent claim 134,** Hoffberg teaches the method wherein said learning

module further comprises at least one of an artificial intelligence algorithm and a machine learning algorithm, and the method is performed by the learning module (See column 42, lines 27-67 and column 132, lines 10-20).

With respect to **dependent claim 135,** Hoffberg teaches the method wherein said learning module maximizes a percentage of proactive alterations leading to a direct user selection from said alteration (See column 51, lines 63-67 and 52, lines 1-26).

With respect to **dependent claim 136,** Hoffberg teaches the method wherein said maximization is performed through learning reinforcement (See column 51, lines 63-67 and 52, lines 1-26 and column 55, lines 58-67 and column 56, lines 1-22).

With respect to **dependent claim 137,** Hoffberg teaches the method wherein said learning reinforcement is performed through an iterative learning process (See column 51, lines 63-67 and 52, lines 1-26 and column 55, lines 58-67 and column 56, lines 1-22).

With respect to **dependent claim 138,** Hoffberg teaches the method wherein each iteration of said learning process is performed after said alteration has been performed (See column 53, lines 19-40 and Examples 12-14).

With respect to **dependent claim 139,** Hoffberg teaches the method wherein said proactively altering at least one function of said user interface comprises activating an additional software application through the operating system (See column 131, medial devices that interact with the system have additional software installed to measure the bio feeds of the user.)

With respect to **dependent claim 140,** Hoffberg teaches the method wherein the method is performed using an intelligent agent capable of communicating with a human user (See example 12-14, column 119-120.

With respect to **dependent claim 141,** Hoffberg teaches the method wherein said intelligent agent controls at least one interaction of the computational device over a network (See Examples 12-14 and Example 17, column 125-126).

(10) Response to Argument

Beginning on page 11 of Appellant's brief (hereinafter Brief), Appellant argues specific issues, which are accordingly addressed below. Appellant has elected the grouped claims and not argued the claims individually and thus the Examiner will present arguments based on the grouped claims.

Claims 1-7 and 109-141

Appellant's argument that the prior art of Hoffberg does not teach the claim limitation recited in claim 1 and 122

Appellant argues that the prior art of Hoffberg does not teach the "proactive altering of at least one function of said interface unit according to detected pattern" because it appears the Appellant does not interpret the function cited in Hoffberg where providing of frequently used choices for program selections, alters the interface (See Brief page 11, middle).

The Examiner respectfully disagrees.

First, as a matter of clarity the Examiner refers to the final rejection mailed 8/29/2008 (page 3, middle) and notes that in the rejection the Examiner not only cited column 85, lines 5-67 but also cited column 50, lines 50-67 and column 42, lines 20-67. The incorporated patents in column 42 will be discussed below in the second argument.

Turning to the first argument, the office position is that Hoffberg teaches an adaptive interface that predicts the desired user function by monitoring the user's history, interface context and machine status (See abstract) and then

changes the interface based on the predicted function. Further Hoffberg teaches that a pattern recognition system is used for a multimedia device where the pattern recognition is used to study the behavior of the user, the user's mood as well as the preferences of the users to determine the familiarity with the operation and functionality of the system. The purpose of the pattern recognition **is to provide a tailored interface adapted to the characteristics of the user** thus adaptively providing access to various features in a hierarchical manner such that the **most likely** feature is presented on the interface rather then the more unlikely feature (See column 97, lines 28-40 and 56-67 and column 98, lines 1-67). The section 97-98 of Hoffberg specifically recites the user interface description used in the preferred embodiments and that a VCR is modeled for program entry.

In the rejection, The Examiner referred to column 85, lines 5-67 as an example of the preferred embodiment VCR program where the specific text recites "the interface system provides an easily accessible CHANGE, CANCEL, or UNDO feature, which facilitates backtracking or **reprogramming the immediately previously** entered information rather then forcing the user to repeat all or substantially all of the programming steps". Therefore, the skilled artisan would determine that a reprogramming of the interface is an altering of at least one function of the interface. In combination with cited feature of tracking and modifying the interface based on "frequently used settings", which is a pattern, the interface is adapted based on "the behavior of use" pattern. It is clear

that the system of Hoffberg detects a pattern of behavior because it tracks frequently used settings and then modifies the interface based on the settings. In combination with the description of the interface example in column 97-98 it is clear that the purpose of Hoffberg's prediction routine is to track the user's history/settings while interacting with the interface to adapt the interface to present likely functions.

As mentioned above, the Examiner also referred to column 50, lines 50-67, which specifically recites that the invention of Hoffberg provides an adaptive interface that changes in response to context, past history and status of the system. The interface provides a predictive algorithm that is modeled after user interactions and the model is adapted to the user pattern. Therefore, as further support, it is shown that the system of Hoffberg is clearly directed to adapting an interface function based on at least one pattern and then altering the interface. The feature of reprogramming the interface is shown for the purposes of presenting the likely feature to the user while they perform the CHANGE, CANCEL or UNDO functions on the interface and the examiner referred to this function in the rejection.

To complete the analysis of the rejection of the entire claim, in the second limitation, the Examiner refers to Figure 15 and column 85, lines 5-67. In figure 15 and the accompanying text (See column 85-87, specifically column 86, lines 1-11 and 49-60) where the predictive function of "analyzing the program sequence to predict the next action" is shown as teaching that the " intelligent

interface stores data concerning programming, user preferences and by means of a logic method the system predicts the entry by a user to generate actions to display to the user". Thus it is shown that the cited section in the final office action has referred to the section of the reference that teaches the claim limitations. It is also shown in column 86, lines 49-60, that accompanies figure 15 that the interface is adapted by using a "teach" mode that acquires the preferences of the user or monitoring actual choices by the user during the operation of the interface to enter in a time into the VCR.

In addition, the present application specification (See PGPUB Para 94-95) states the following:

[0094] The proactive user interface of the present invention is preferably able to control and/or be associated with any type of computational device, in order to actively make suggestions to the user, **based upon prior experience with a particular user and/or various preprogrammed patterns from which the computational device could select, depending upon user behavior**. These suggestions could optionally be made by altering the appearance of at least a portion of the display, for example by changing a menu or a portion thereof; providing different menus for display; and/or altering touch screen functionality. The suggestions could also optionally be made audibly.

[0095] The proactive user interface is preferably implemented for a computational device, as previously described, which includes an operating system. The interface optionally and preferably includes a user interface for communicating between the user and the operating system. The interface is preferably able to detect at least one pattern of interaction of the user with the user interface, for example through operation of a learning module and is therefore preferably able to proactively alter at least one function of the user interface according to the detected pattern.

Therefore, the proactive user interface can anticipate the requests of the user and thereby assist the user in selecting a desired function of the computational device.

Thus, the present application specification defines the predictive altering of the interface to be based on user's behavior and interactions with pre-programmed patterns of the device. In the examples above and the final rejection, it is clear that Hoffberg monitors a users past history of interactions with the VCR and then predicts the next function to present to the user based on the history of interaction with the VCR. By predicting the next function the system is altering the function of the device to allow for the user history of interaction. Thus it appears the structure taught in the present application is shown in the prior art and is the basis for the anticipation rejection presented by the examiner.

Moreover, as background to the argument, Hoffberg states that "intelligent or learning systems are known" (See column 1, lines 30-35) and that "optimization schemes optimize the mechanical elements of the system to provide a universally optimized interface" (See column 2, lines 8-21) are also known. Further, (See the present application specification Para 96 of the PGPUB) where Appellant states that learning algorithms are known in the art. Additionally, Hoffberg shows that the preferred embodiment is to a VCR interface and a programming preference prediction mechanism to alter the interface. Hoffberg shows how the device monitors the users input over time to predict the next function (See column 100, lines 1-21), which was specifically cited by the examiner in the final rejection. Finally, the VCR example provides a smart screen that is an adaptive interface that allows the interface to anticipate or

predict the intent of the user to provide a user choice by default, which is the most likely choice, and then display the choice to the user (See column 11, lines 3-25) and yet another example as to how the teachings of Hoffberg anticipate the claims.

Appellant's argument that the Examiner improperly used two references for a 102 Rejection

Appellant argues that the Examiner refers to a cited section of the reference that shows Incorporated by Reference patents and improperly based the rejection on two references rather than a single reference to anticipate the claims (See Brief page 12).

The Examiner disagrees.

First, the Examiner has reviewed the final rejection and other than mention of column 42, lines 20-67 in the rejection of claim 1 and 122, nowhere in the rejection is there a mention of any reference used in the rejection other than Hoffberg. Further, the rejection of claim 1 does not refer to any specific patent or reference other then Hoffberg. Simply, the rejection is not based on a dual reference rejection it is based on the teachings of Hoffberg as a single reference. As mentioned in the advisory action mailed 04/21/2008 and in the arguments attached to the final rejection mailed the Examiner referred to the section 42, **as a demonstration** to the state of the art and what was known at the time of the filing of Hoffberg, and the references where something for the Appellant to consider. Hoffberg clearly shows that the patents properly incorporated by

reference are relevant to the art of pattern recognition. At the beginning of the background section (See column 1, lines 30-67) Hoffberg clearly states that the incorporate by reference patents are shown to detail how the invention builds on the prior art and how various problems are addressed with intelligent systems and interfaces. Hoffberg specifically states that adaptive and human responsive interface are well known (See column 2, lines 10-16). While the examiner pointed to the incorporated by reference patents, the rejection points only to disclosure of Hoffberg and Hoffberg clearly shows that the prior art as of the filing (Feb 1, 1999) stated that adaptive interfaces are well known and therefore provides a basis that future adaptive interface applications should consider. Therefore, the rejection clearly shows **both** a reference to what the prior art teaches and what is known in the prior art of pattern recognition, and a specific teaching as to the claim limitations within the cited sections of Hoffberg. In summary, the Examiner understands the MPEP section 2131.01 and 2163.07. The examiner did not write a dual rejection or even mention a specific patent in the rejection therefore 2131 is not applicable. Instead the Examiner refers to 2167.07 to show that patents with incorporated by reference patents have material that should be considered as a part of the incorporating reference because incorporated by reference patents provide subject matter that support the written description of the incorporating reference and can be used if the need arises to support a rejection.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Steven B. Theriault/

Conferees:

/Weilun Lo/
Supervisory Patent Examiner, Art Unit 2179

/Ba Huynh/

Primary Examiner, Art Unit 2179

THE FARRELL LAW FIRM, PC
333 Earle Ovington Blvd., Suite 701
Uniondale, New York 11553