

(19) World Intellectual Property
Organization
International Bureau



(43) International Publication Date
9 December 2004 (09.12.2004)

PCT

(10) International Publication Number
WO 2004/107150 A1

(51) International Patent Classification⁷: **G06F 3/00**

(21) International Application Number:
PCT/JP2004/007905

(22) International Filing Date: 1 June 2004 (01.06.2004)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
2003-156807 2 June 2003 (02.06.2003) JP

(71) Applicant (for all designated States except US): **CANON KABUSHIKI KAISHA** [JP/JP]; 3-30-2, Shimomaruko, Ohta-ku, Tokyo, 1468501 (JP).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **IKEDA, Hiromi** [JP/JP]; c/o CANON KABUSHIKI KAISHA, 3-30-2, Shimomaruko, Ohta-ku, Tokyo, 1468501 (JP). **HI-ROTA, Makoto** [JP/JP]; c/o CANON KABUSHIKI KAISHA, 3-30-2, Shimomaruko, Ohta-ku, Tokyo, 1468501 (JP). **NAKAGAWA, Kenichirou** [JP/JP]; c/o CANON KABUSHIKI KAISHA, 3-30-2, Shimomaruko, Ohta-ku, Tokyo, 1468501 (JP).

(74) Agent: **OHTSUKA, Yasunori**; Shuwa Kioicho Park Bldg. 7th FL., 3-6, Kioicho, Chiyoda-Ku, Tokyo, 1020094 (JP).

(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, KE, KG, KP, KR, 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.

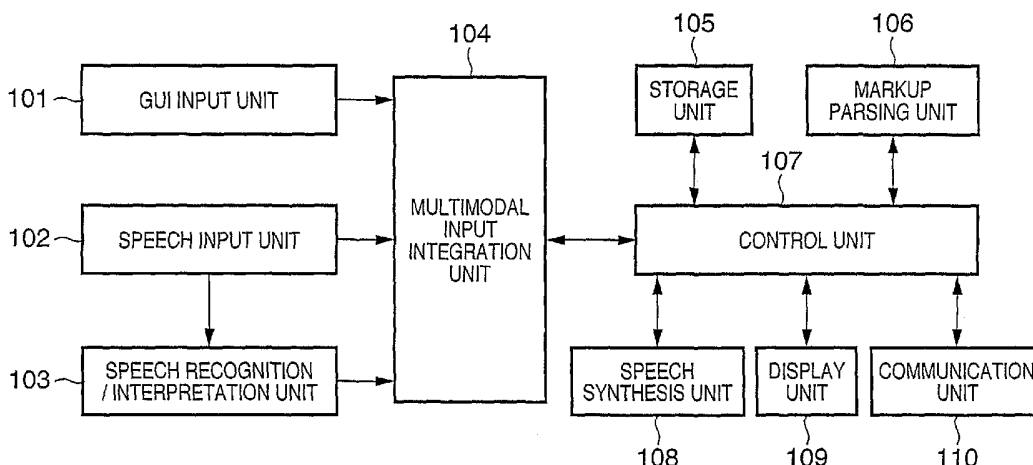
(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, 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).

Published:

- with international search report
- before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

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.

(54) Title: INFORMATION PROCESSING METHOD AND APPARATUS



(57) Abstract: In an information processing method for processing a user's instruction on the basis of a plurality of pieces of input information which are input by a user using a plurality of types of input modalities, each of the plurality of types of input modalities has a description including correspondence between the input contents and semantic attributes. Each input content is acquired by parsing each of the plurality of pieces of input information which are input using the plurality of types of input modalities, and semantic attributes of the acquired input contents are acquired from the description. A multimodal input integration unit integrates the acquired input contents on the basis of the acquired semantic attributes.

DESCRIPTION

INFORMATION PROCESSING METHOD AND APPARATUS

TECHNICAL FIELD

5 The present invention relates to a so-called multimodal user interface used to issue instructions using a plurality of types of input modalities.

BACKGROUND ART

10 A multimodal user interface which allows to input using a desired one of a plurality of types of modalities (input modes) such as a GUI input, speech input, and the like is very convenient for the user. Especially, high convenience is obtained upon making
15 inputs by simultaneously using a plurality of types of modalities. For example, when the user clicks a button indicating an object on a GUI while uttering an instruction word such as "this" or the like, even the user who is not accustomed to a technical language such
20 as commands or the like can freely operate the objective device. In order to attain such operations, a process for integrating inputs by means of a plurality of types of modalities is required.

 As examples of the process for integrating inputs
25 by means of a plurality of types of modalities, a method of applying language interpretation to a speech recognition result (Japanese Patent Laid-Open

No. 9-114634), a method using context information
(Japanese Patent Laid-Open No. 8-234789), a method of
combining inputs with approximate input times, and
outputting them as a semantic interpretation unit
5 (Japanese Patent Laid-Open No. 8-263258), and a method
of making language interpretation and using a semantic
structure (Japanese Patent Laid-Open No. 2000-231427)
have been proposed.

Also, the IBM et al. have formulated a
10 specification "XHTML + Voice Profile", and this
specification allows to describe a multimodal user
interface in a markup language. Details of this
specification are described in the W3C Web site
(<http://www.w3.org/TR/xhtml+voice/>). The SALT Forum
15 has published a specification "SALT", and this
specification allows to describe a multimodal user
interface in a markup language as in XHTML + Voice
Profile above. Details of this specification are
described in the SALT Forum Web site (The Speech
20 Application Language Tags: <http://www.saltforum.org/>).

However, these prior arts require complicated
processes such as language interpretation and the like
upon integrating a plurality of types of modalities.
Even when such complicated process is done, the meaning
25 of inputs that the user intended cannot sometimes be
reflected in an application due to an interpretation
error and the like of language interpretation.

Techniques represented by XHTML + Voice Profile and
SALT, and the conventional description method using a
markup language have no scheme that handles a
description of semantic attributes which represent
5 meanings of inputs.

DISCLOSURE OF INVENTION

The present invention has been made in
consideration of the above situation, and has as its
10 object to implement multimodal input integration that
the user intended by a simple process.

More specifically, it is another object of the
present invention to implement integration of inputs
that the user or designer intended by a simple
15 interpretation process by adopting a new description
such as a description of semantic attributes that
represent meanings of inputs in a description for
processing inputs from a plurality of types of
modalities.

20 It is still another object of the present
invention to allow an application developer to describe
semantic attributes of inputs using a markup language
or the like.

In order to achieve the above objects, according
25 to one aspect of the present invention, there is
provided an information processing method for
recognizing a user's instruction on the basis of a

plurality of pieces of input information which are input by a user using a plurality of types of input modalities, the method having a description including correspondence between input contents and a semantic attribute for each of the plurality of types of input modalities, the method comprising: an acquisition step of acquiring an input content by parsing each of the plurality of pieces of input information which are input using the plurality of types of input modalities, and acquiring semantic attributes of the acquired input contents from the description; and an integration step of integrating the input contents acquired in the acquisition step on the basis of the semantic attributes acquired in the acquisition step.

Other features and advantages of the present invention will be apparent from the following description taken in conjunction with the accompanying drawings, in which like reference characters designate the same or similar parts throughout the figures thereof.

BRIEF DESCRIPTION OF DRAWINGS

The accompanying drawings, which are incorporated in and constitute a part of the specification, illustrate embodiments of the invention and, together with the description, serve to explain the principles of the invention.

Fig. 1 is a block diagram showing the basic arrangement of an information processing system according to the first embodiment;

Fig. 2 shows a description example of semantic attributes by a markup language according to the first embodiment;

Fig. 3 shows a description example of semantic attributes by a markup language according to the first embodiment;

Fig. 4 is a flowchart for explaining the flow of the process of a GUI input processor in the information processing system according to the first embodiment;

Fig. 5 is a table showing a description example of grammar (rules of grammar) for speech recognition according to the first embodiment;

Fig. 6 shows a description example of the grammar (rules of grammar) for speech recognition using a markup language according to the first embodiment;

Fig. 7 shows a description example of the speech recognition/interpretation result according to the first embodiment;

Fig. 8 is a flowchart for explaining the flow of the process of a speech recognition/interpretation processor 103 in the information processing system according to the first embodiment;

Fig. 9A is a flowchart for explaining the flow of the process of a multimodal input integration unit 104

in the information processing system according to the first embodiment;

Fig. 9B is a flowchart showing details of step S903 in Fig. 9A;

5 Fig. 10 shows an example of multimodal input integration according to the first embodiment;

Fig. 11 shows an example of multimodal input integration according to the first embodiment;

10 Fig. 12 shows an example of multimodal input integration according to the first embodiment;

Fig. 13 shows an example of multimodal input integration according to the first embodiment;

Fig. 14 shows an example of multimodal input integration according to the first embodiment;

15 Fig. 15 shows an example of multimodal input integration according to the first embodiment;

Fig. 16 shows an example of multimodal input integration according to the first embodiment;

20 Fig. 17 shows an example of multimodal input integration according to the first embodiment;

Fig. 18 shows an example of multimodal input integration according to the first embodiment;

Fig. 19 shows an example of multimodal input integration according to the first embodiment;

25 Fig. 20 shows a description example of semantic attributes using a markup language according to the second embodiment;

Fig. 21 shows a description example of grammar (rules of grammar) for speech recognition according to the second embodiment;

Fig. 22 shows a description example of the speech
5 recognition/interpretation result according to the second embodiment;

Fig. 23 shows an example of multimodal input integration according to the second embodiment;

Fig. 24 shows a description example of semantic
10 attributes including "ratio" using a markup language according to the second embodiment;

Fig. 25 shows an example of multimodal input integration according to the second embodiment;

Fig. 26 shows a description example of the
15 grammar (rules of grammar) for speech recognition according to the second embodiment; and

Fig. 27 shows an example of multimodal input integration according to the second embodiment.

20 BEST MODE FOR CARRYING OUT THE INVENTION

Preferred embodiments of the present invention will now be described in detail in accordance with the accompanying drawings.

[First Embodiment]

25 Fig. 1 is a block diagram showing the basic arrangement of an information processing system according to the first embodiment. The information

processing system has a GUI input unit 101, speech
input unit 102, speech recognition/interpretation unit
103, multimodal input integration unit 104, storage
unit 105, markup parsing unit 106, control unit 107,
5 speech synthesis unit 108, display unit 109, and
communication unit 110.

The GUI input unit 101 comprises input devices
such as a button group, keyboard, mouse, touch panel,
pen, tablet, and the like, and serves as an input
10 interface used to input various instructions from the
user to this apparatus. The speech input unit 102
comprises a microphone, A/D converter, and the like,
and converts user's utterance into a speech signal.
The speech recognition/interpretation unit 103
15 interprets the speech signal provided by the speech
input unit 102, and performs speech recognition. Note
that a known technique can be used as the speech
recognition technique, and a detailed description
thereof will be omitted.

20 The multimodal input integration unit 104
integrates information input from the GUI input unit
101 and speech recognition/interpretation unit 103.
The storage unit 105 comprises a hard disk drive device
used to save various kinds of information, a storage
25 medium such as a CD-ROM, DVD-ROM, and the like used to
provide various kinds of information to the information
processing system and a drive, and the like. The hard

disk drive device and storage medium store various application programs, user interface control programs, various data required upon executing the programs, and the like, and these programs are loaded onto the system
5 under the control of the control unit 107 (to be described later).

The markup parsing unit 106 parses a document described in a markup language. The control unit 107 comprises a work memory, CPU, MPU, and the like, and
10 executes various processes for the whole system by reading out the programs and data stored in the storage unit 105. For example, the control unit 107 passes the integration result of the multimodal input integration unit 104 to the speech synthesis unit 108 to output it
15 as synthetic speech, or passes the result to the display unit 109 to display it as an image. The speech synthesis unit 108 comprises a loudspeaker, headphone, D/A converter, and the like, and executes a process for generating speech data based on read text, D/A-converts
20 the data into analog data, and externally outputs the analog data as speech. Note that a known technique can be used as the speech synthesis technique, and a detailed description thereof will be omitted. The display unit 109 comprises a display device such as a
25 liquid crystal display or the like, and displays various kinds of information including an image, text, and the like. Note that the display unit 109 may adopt

a touch panel type display device. In this case, the display unit 109 also has a function of the GUI input unit (a function of inputting various instructions to this system). The communication unit 110 is a network
5 interface used to make data communications with other apparatuses via networks such as the Internet, LAN, and the like.

Mechanisms (GUI input and speech input) for making inputs to the information processing system with
10 the above arrangement will be described below.

A GUI input will be explained first. Fig. 2 shows a description example using a markup language (XML in this example) used to present respective components. Referring to Fig. 2, an <input> tag
15 describes each GUI component, and a type attribute describes the type of component. A value attribute describes a value of each component, and a ref attribute describes a data model as a bind destination of each component. Such XML document complies with the
20 specification of W3C (World Wide Web Consortium), i.e., it is a known technique. Note that details of the specification are described in the W3C Web site (XHTML: <http://www.w3.org/TR/xhtml11/>, XForms: <http://www.w3.org/TR/xforms/>).

25 In Fig. 2, a meaning attribute is prepared by expanding the existing specification, and has a structure that can describe a semantic attribute of

each component. Since the markup language is allowed to describe semantic attributes of components, an application developer himself or herself can easily set the meaning of each component that he or she intended.

5 For example, in Fig. 2, a meaning attribute "station" is given to "SHIBUYA", "EBISU", and "JIYUGAOKA". Note that the semantic attribute need not always use a unique specification like the meaning attribute. For example, a semantic attribute may be described using an

10 existing specification such as a class attribute in the XHTML specification, as shown in Fig. 3. The XML document described in the markup language is parsed by the markup parsing unit 106 (XML parser).

The GUI input processing method will be described

15 using the flowchart of Fig. 4. When the user inputs, e.g., an instruction of a GUI component from the GUI input unit 101, a GUI input event is acquired (step S401). The input time (time stamp) of that instruction is acquired, and the semantic attribute of the

20 designated GUI component is set to be that of the input with reference to the meaning attribute in Fig. 2 (or the class attribute in Fig. 3) (step S402).

Furthermore, the bind destination of data and input value of the designated component are acquired from the

25 aforementioned description of the GUI component. The bind destination, input value, semantic attribute, and time stamp acquired for the data of the component are

output to the multimodal input integration unit 104 as input information (step S403).

A practical example of the GUI input process will be described below with reference to Figs. 10 and 11.

5 Fig. 10 shows a process executed when a button with a value "1" is pressed via the GUI. This button is described in the markup language, as shown in Fig. 2 or 3, and it is understood by parsing this markup language that the value is "1", the semantic attribute is
10 "number", and the data bind destination is "/Num".
Upon depression of the button "1", the input time (time stamp; "00:00:08" in Fig. 10) is acquired. Then, the value "1", semantic attribute "number", and data bind destination "/Num" of the GUI component, and the time
15 stamp are output to the multimodal input integration unit 104 (Fig. 10: 1002).

Likewise, when a button "EBISU" is pressed, as shown in Fig. 11, a time stamp ("00:00:08" in Fig. 11), a value "EBISU" obtained by parsing the markup language
20 in Fig. 2 or 3, a semantic attribute "station", and a data bind destination "-" (no bind)" is output to the multimodal input integration unit 104 (Fig. 11: 1102).
With the above process, the semantic attribute that the application developer intended can be handled as
25 semantic attribute information of the inputs on the application side.

The speech input process from the speech input unit 102 will be described below. Fig. 5 shows grammar (rules of grammar) required to recognize speech. Fig. 5 shows grammar that describes rules for

5 recognizing speech inputs such as "from here", "to EBISU", and the like and outputting interpretation results from="@unknown", to="EBISU", and the like. In Fig. 5, an input string is input speech, and has a structure that describes a value corresponding to the

10 input speech in a value string, a semantic attribute in a meaning string, and a data model of the bind destination in a DataModel string. Since the grammar (rules of grammar) required to recognize speech can describe a semantic attribute (meaning), the

15 application developer himself or herself can easily set the semantic attribute corresponding to each speech input, and the need for complicated processes such as language interpretation and the like can be obviated.

In Fig. 5, the value string describes a special

20 value (@unknown in this example) for an input such as "here" or the like which cannot be processed if it is input alone, and requires correspondence with an input by means of another modality. By specifying this special value, the application side can determine that

25 such input cannot be processed alone, and can skip processes such as language interpretation and the like. Note that the grammar (rules of grammar) may be

described using the specification of W3C, as shown in Fig. 6. Details of the specification are described in the W3C Web site (Speech Recognition Grammar Specification: <http://www.w3.org/TR/speech-grammar/>,

5 Semantic Interpretation for Speech Recognition:

<http://www.w3.org/TR/semantic-interpretation/>). Since the W3C specification does not have a structure that describes the semantic attribute, colon (:) and the semantic attribute are appended to the interpretation

10 result. Hence, a process for separating the interpretation result and semantic attribute is required later. The grammar described in the markup language is parsed by the markup parsing unit 106 (XML parser).

15 The speech input/interpretation process method will be described below using the flowchart of Fig. 8. When the user inputs speech from the speech input unit 102, a speech input event is acquired (step S801). The input time (time stamp) is acquired, and a speech
20 recognition/interpretation process is executed (step S802). Fig. 7 shows an example of the interpretation process result. For example, when a speech processor connected to a network is used, the interpretation result is obtained as an XML document shown in Fig. 7.
25 In Fig. 7, an `<nlsml: interpretation>` tag indicates one interpretation result, and a confidence attribute indicates its confidence. Also, an `<nlsml: input>` tag

indicates texts of input speech, and an <nlsml:
instance> tag indicates the recognition result. The
W3C has published the specification required to express
the interpretation result, and details of the
5 specification are described in the W3C Web site
(Natural Language Semantics Markup Language for the
Speech Interface Framework:
<http://www.w3.org/TR/nl-spec/>). As in the grammar, the
speech interpretation result (input speech) can be
10 parsed by the markup parsing unit 106 (XML parser). A
semantic attribute corresponding to this interpretation
result is acquired from the description of the rules of
grammar (step S803). Furthermore, a bind destination
and input value corresponding to the interpretation
15 result are acquired from the description of the rules
of grammar, and are output to the multimodal input
integration unit 104 as input information together with
the semantic attribute and time stamp (step S804).

A practical example of the aforementioned speech
20 input process will be described below using Figs. 10
and 11. Fig. 10 shows a process when speech "to EBISU
" is input. As can be seen from the grammar (rules of
grammar) in Fig. 6, when speech "to EBISU" is input,
the value is "EBISU", the semantic attribute is
25 "station", and the data bind destination is "/To".
When speech "to EBISU" is input, its input time (time
stamp; "00:00:06" in Fig. 10) is acquired, and is

output to the multimodal input integration unit 104 together with the value "EBISU", semantic attribute "station", and data bind destination "/To" (Fig. 10: 1001). Note that the grammar (grammar for speech recognition) in Fig. 6 allows a speech input as a combination of one of "here", "SHIBUYA", "EBISU", "JIYUGAOKA", "TOKYO", and the like bounded by <one-of> and </one-of> tags, and "from" or "to" (for example, "from here" and "to EBISU"). Also, such combinations can also be combined (for example, "from SHIBUYA to JIYUGAOKA" and "to here, from TOKYO"). A word combined with "from" is interpreted as a from value, a word combined with "to" is interpreted as a to value, and contents bounded by <item>, <tag>, </tag>, and </item> are returned as an interpretation result. Therefore, when speech "to EBISU" is input, "EBISU: station" is returned as a to value, and when speech "from here" is input, "@unknown: station" is returned as a from value. When speech "from EBISU to TOKYO" is input, "EBISU: station" is returned as a from value, and "TOKYO: station" is returned as a to value.

Likewise, when speech "from here" is input, as shown in Fig. 11, a time stamp "00:00:06", and an input value "@unknown", semantic attribute "station", and data bind destination "/From", which are acquired based on the grammar (rules of grammar) in Fig. 6, are output to the multimodal input integration unit 104 (Fig. 11:

1101). With the above process, in the speech input process, the semantic attribute that the application developer intended can be handled as semantic attribute information of the inputs on the application side.

5 The operation of the multimodal input integration unit 104 will be described below with reference to Figs. 9A to 19. Note that this embodiment will explain a process for integrating input information (multimodal inputs) from the aforementioned GUI input unit 101 and
10 speech input unit 102.

 Fig. 9A is a flowchart showing the process method for integrating input information from the respective input modalities in the multimodal input integration unit 104. When the respective input modalities output
15 a plurality of pieces of input information (data bind destination, input value, semantic attribute, and time stamp), these pieces of input information are acquired (step S901), and all pieces of input information are sorted in the order of time stamps (step S902). Next,
20 a plurality of pieces of input information with the same semantic attribute are integrated in correspondence with their input order (step S903). That is, a plurality of pieces of input information with the same semantic attribute are integrated
25 according to their input order. More specifically, the following process is done. That is, for example, when inputs "from here (click SHIBUYA) to here (click

EBISU)" are input, a plurality of pieces of speech input information are input in the order of:

(1) here (station) ← "here" of "from here"

(2) here (station) ← "here" of "to here"

5 Also, a plurality of pieces of GUI input (click) information are input in the order of:

(1) SHIBUYA (station)

(2) EBISU (station)

Then, inputs (1) and inputs (2) are respectively
10 integrated.

As conditions required to integrate a plurality of pieces of input information,

(1) the plurality of pieces of information require an integration process;

15 (2) the plurality of pieces of information are input within a time limit (e.g., the time stamp difference is 3 sec or less);

(3) the plurality of pieces of information have the same semantic attribute;

20 (4) the plurality of pieces of information do not include any input information having a different semantic attribute when they are sorted in the order of time stamps;

(5) "bind destination" and "value" have a
25 complementary relationship; and

(6) information, which is input earliest, of those which satisfy (1) to (4), is to be integrated. A

plurality of pieces of input information which satisfy these integration conditions are to be integrated.

Note that the integration conditions are an example, and other conditions may be set. For example, a

5 spatial distance (coordinates) of inputs may be adopted.

Note that the coordinates of the TOKYO station, EBISU station, and the like on the map may be used as the coordinates. Also, some of the above integration conditions may be used as the integration conditions

10 (for example, only conditions (1) and (3) are used as the integration conditions). In this embodiment, inputs of different modalities are integrated, but inputs of an identical modality are not integrated.

Note that condition (4) is not always necessary.

15 However, by adding this condition, the following advantages are expected.

For example, when speech "from here, two tickets, to here" is input, if it is considered as click timings and integration interpretations that

20 (a) "(click) from here, two tickets, to here" → it is natural to integrate click and "here (from)";

(b) "from (click) here, two tickets, to here" → it is natural to integrate click and "here (from)";

(c) "from here (click), two tickets, to here" →
25 it is natural to integrate click and "here (from)";

(d) "from here, two (click) tickets, to here" →
it is hard to say even for humans whether click is to
be integrated with "here (from)" or "here (to)"; and

(e) "from here, two tickets, (click) to here" →
5 it is natural to integrate click and "here (to)",
when condition (4) is not used, i.e., when a different
semantic attribute can be included, click and "here
(from)" are integrated in (e) above if they have close
timings. However, it is obvious for those who are
10 skilled in the art that such conditions may change
depending on the use purposes of an interface.

Fig. 9B is a flowchart for explaining the
integration process in step S903 in more detail. After
a plurality of pieces of input information are sorted
15 in the chronological order in step S902, the first
input information is selected in step S911. It is
checked in step S912 if the selected input information
requires integration. In this case, if at least one of
the bind destination and input value of the input
20 information is not settled, it is determined that
integration is required; if both the bind destination
and input values are settled, it is determined that
integration is not required. If it is determined that
integration is not required, the flow advances to step
25 S913, and the multimodal input integration unit 104
outputs the bind destination and input value of that
input information as a single input. At the same time,

a flag indicating that the input information is output is set. The flow then jumps to step S919.

On the other hand, if it is determined that integration is required, the flow advances to step S914
5 to search for input information, which is input before the input information of interest, and satisfies the integration conditions. If such input information is found, the flow advances from step S915 to step S916 to integrate the input information of interest with the
10 found input information. This integration process will be described later using Figs. 10 to 19. The flow advances to step S917 to output the integration result, and to set a flag indicating that the two pieces of input information are integrated. The flow then
15 advances to step S919.

If the search process cannot find any input information that can be integrated, the flow advances to step S918 to hold the selected input information intact. The next input information is selected (steps
20 S919 and S920, and the aforementioned processes are repeated from step S912. If it is determined in step S919 that no input information to be processed remains, this process ends.

Examples of the multimodal input integration
25 process will be described in detail below with reference to Figs. 10 to 19. In the description of each process, the step numbers in Fig. 9B are described

in parentheses. Also, the GUI inputs and grammar for speech recognition are defined, as shown in Fig. 2 or 3, and Fig. 6.

An example of Fig. 10 will be explained. As described above, speech input information 1001 and GUI input information 1002 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp (in Fig. 10, circled numbers indicate the order). In the speech input information 1001, all the data bind destination, semantic attribute, and value are settled. For this reason, the multimodal input integration unit 104 outputs the data bind destination "/To" and value "EBISU" as a single input (Fig. 10: 1004, S912, S913 in Fig. 9B). Likewise, since all the data bind destination, semantic attribute, and value are settled in the GUI input information 1002, the multimodal input integration unit 104 outputs the data bind destination "/Num" and value "1" as a single input (Fig. 10: 1003).

An example of Fig. 11 will be described below. Since speech input information 1101 and GUI input information 1102 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp, the speech input information 1101 is processed first. The speech input information 1101 cannot be processed as a single input and requires an integration process, since its value is "@unknown".

As information to be integrated, GUI input information input before the speech input information 1101 is searched for an input that similarly requires an integration process (in this case, information whose
5 bind destination is not settled). In this case, since there is no input before the speech input information 1101, the process of the next GUI input information 1102 starts while holding the information. The GUI input information 1102 cannot be processed as a single
10 input and requires an integration process (S912), since its data model is "- (no bind)".

In case of Fig. 11, since input information that satisfies the integration conditions is the speech input information 1101, the GUI input information 1102
15 and speech input information 1101 are selected as information to be integrated (S915). The two pieces of information are integrated, and the data bind destination "/From" and value "EBISU" are output (Fig. 11: 1103) (S916).

20 An example of Fig. 12 will be described below. Speech input information 1201 and GUI input information 1202 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp. The speech input information 1201
25 cannot be processed as a single input and requires an integration process, since its value is "@unknown". As information to be integrated, GUI input information

input before the speech input information 1201 is searched for an input that similarly requires an integration process. In this case, since there is no input before the speech input information 1201, the process of the next GUI input information 1202 starts while holding the information. The GUI input information 1202 cannot be processed as a single input and requires an integration process, since its data model is "- (no bind)". As information to be integrated, speech input information input before the GUI input information 1202 is searched for input information that satisfies the integration condition (S912, S914). In this case, the speech input information 1201 input before the GUI input information 1202 has a different semantic attribute from that of the information 1202, and does not satisfy the integration condition. Therefore, the integration process is skipped, and the next process starts while holding the information as in the speech input information 1201 (S914, S915 - S918).

An example of Fig. 13 will be described below. Speech input information 1301 and GUI input information 1302 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp. The speech input information 1301 cannot be processed as a single input and requires an integration process (S912), since its value is

"@unknown". As information to be integrated, GUI input information input before the speech input information 1301 is searched for an input that similarly requires an integration process (S914). In this case, since
5 there is no input before the speech input information 1301, the process of the next GUI input information 1302 starts while holding the information. Since all the data bind destination, semantic attribute, and value are settled in the GUI input information 1302,
10 the data bind destination "/Num" and value "1" are output as a single input (Fig. 13: 1303) (S912, S913). Hence, the speech input information 1301 is kept held.

An example of Fig. 14 will be described below. Speech input information 1401 and GUI input information
15 1402 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp. Since all the data bind destination (/To), semantic attribute, and value are settled in the speech input information 1401, the data
20 bind destination "/To" and value "EBISU" are output as a single input (Fig. 14: 1404) (S912, S913). Next, in the GUI input information 1402 as well, the data bind destination "/To" and value "JIYUGAOKA" are output as a single input (Fig. 14: 1403) (S912, S913). As a result,
25 since 1403 and 1404 have the same data bind destination "/To", the value "JIYUGAOKA" of 1403 is overwritten on the value "EBISU" of 1404. That is, the contents of

1404 are output, and those of 1403 are then output.
Such state is normally considered as "contention of
information" since "EBISU" is received as one input and
"JIYUGAOKA" is received as the other input, although
5 identical data are to be input in the same time band.
In this case, which of information is to be selected is
a problem. A method of processing information after a
chronologically close input is waited may be used.
However, much time is required with this method until
10 the processing result is obtained. Hence, this
embodiment executes a process for outputting data in
turn without waiting for such input.

An example of Fig. 15 will be described below.
Speech input information 1501 and GUI input information
15 1502 are sorted in the order of time stamps, and are
processed in turn from input information with an
earlier time stamp. In this case, since the two pieces
of input information have the same time stamp, the
processes are done in the order of a speech modality
20 and GUI modality. As for this order, these pieces of
information may be processed in the order that they
arrive the multimodal input integration unit, or in the
order of input modalities set in advance in a browser.
As a result, since all the data bind destination,
25 semantic attribute, and value of the speech input
information 1501 are settled, the data bind destination
"/To" and value "EBISU" are output as a single input

(Fig. 15: 1504). Next, when the GUI input information 1502 is processed, the data bind destination "/To" and value "JIYUGAOKA" are output as a single input (Fig. 15: 1503). As a result, since 1503 and 1504 have
5 the same data bind destination "/To", the value "JIYUGAOKA" of 1503 is overwritten on the value "EBISU" of 1504.

An example of Fig. 16 will be described below.

Speech input information 1601, speech input information
10 1602, GUI input information 1603, and GUI input information 1604 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp (indicated by circled numbers 1 to 4 in Fig. 16). The speech input information 1601
15 cannot be processed as a single input and requires an integration process (S912), since its value is "@unknown". As information to be integrated, GUI input information input before the speech input information 1601 is searched for an input that similarly requires
20 an integration process (S914). In this case, since there is no input before the speech input information 1601, the process of the next GUI input information 1602 starts while holding the information (S915, S918 - S920). The GUI input information 1603 cannot be
25 processed as a single input and requires an integration process (S912), since its data model is "- (no bind)". As information to be integrated, speech input

information input before the GUI input information 1603 is searched for input information that satisfies the integration condition (S914). In case of Fig. 16, since the speech input information 1601 and GUI input information 1603 satisfy the integration conditions, the GUI information 1603 and speech input information 1601 are integrated (S916). After these two pieces of information are integrated, the data bind destination "/From" and value "SHIBUYA" are output (Fig. 16: 1606) (S917), and the process of the speech input information 1602 as the next information starts (S920). The speech input information 1602 cannot be processed as a single input and requires an integration process (S912), since its value is "@unknown". As information to be integrated, GUI input information input before the speech input information 1602 is searched for an input that similarly requires an integration process (S914). In this case, the GUI input information 1603 has already been processed, and there is no GUI input information that requires an integration process before the speech input information 1602. Hence, the process of the next GUI information 1604 starts while holding the speech input information 1602 (S915, S918 - S920). The GUI input information 1604 cannot be processed as a single input and requires an integration process, since its data model is "- (no bind)" (S912). As information to be integrated, speech input information input before

the GUI input information 1604 is searched for input information that satisfies the integration condition (S914). In this case, since input information that satisfies the integration condition is the speech input information 1602, the GUI input information 1604 and speech input information 1602 are integrated. These two pieces of information are integrated, and the data bind destination "/To" and value "EBISU" are output (Fig. 16: 1605) (S915 - S917).

10 An example of Fig. 17 will be described below. Speech input information 1701, speech input information 1702, and GUI input information 1703 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp. The speech input information 1701 as the first input information cannot be processed as a single input and requires an integration process, since its value is "@unknown". As information to be integrated, GUI input information input before the speech input information 1701 is searched for an input that similarly requires an integration process (S912, S914). In this case, since there is no input before the speech input information 1701, the process of the next speech input information 1702 starts while holding this information (S915, S918 - S920). Since all the data bind destination, semantic attribute, and value of the speech input information 1702 are settled, the data

bind destination "/To" and value "EBISU" are output as a single input (Fig. 17: 1704) (S912, S913).

Subsequently, the process of the GUI input information 1703 as the next input information starts.

5 The GUI input information 1703 cannot be processed as a single input and requires an integration process, since its data model is "- (no bind)". As information to be integrated, speech input information input before the GUI input information 1703 is searched for input
10 information that satisfies the integration condition. As input information that satisfies the integration condition, the speech input information 1701 is found. Hence, the GUI input information 1703 and speech input information 1701 are integrated and, as a result, the
15 data bind destination "/From" and value "SHIBUYA" are output (Fig. 17: 1705) (S915 - S917).

An example of Fig. 18 will be described below.

Speech input information 1801, speech input information 1802, GUI input information 1803, and GUI input
20 information 1804 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp. In case of Fig. 18, these pieces of input information are processed in the order of 1803, 1801, 1804, and 1802.

25 The first GUI input information 1803 cannot be processed as a single input and requires an integration process, since its data model is "- (no bind)". As

information to be integrated, speech input information input before the GUI input information 1803 is searched for input information that satisfies the integration condition. In this case, since there is no input

5 before the GUI input information 1803, the process of the speech input information 1801 as the next input information starts while holding the information (S912, S914, S915). The speech input information 1801 cannot be processed as a single input and requires an

10 integration process, since its value is "@unknown". As information to be integrated, GUI input information input before the speech input information 1801 is searched for an input that similarly requires an integration process (S912, S914). In this case, the

15 GUI input information 1803 input before the speech input information 1801 is present, but it reaches a time-out (the time stamp difference is 3 sec or more) and does not satisfy the integration conditions. Hence, the integration process is not executed. As a result,

20 the process of the next GUI information 1804 starts while holding the speech input information 1801 (S915, S918 - S920).

The GUI input information 1804 cannot be processed as a single input and requires an integration

25 process, since its data model is "- (no bind)". As information to be integrated, speech input information input before the GUI input information 1804 is searched

for input information that satisfies the integration condition (S912, S914). In case of Fig. 18, since the speech input information 1801 satisfies the integration conditions, the GUI information 1804 and speech input
5 information 1801 are integrated. After these two pieces of information are integrated, the data bind destination "/From" and value "EBISU" are output (Fig. 18: 1805) (S915 - S917).

After that, the process of the speech input
10 information 1802 starts. The speech input information 1802 cannot be processed as a single input and requires an integration process, since its value is "@unknown". As information to be integrated, GUI input information input before the speech input information 1802 is
15 searched for an input that similarly requires an integration process (S912, S914). In this case, since there is no input before the speech input information 1802, the next process starts while holding the information (S915, S918 - S920).

20 An example of Fig. 19 will be described below. Speech input information 1901, speech input information 1902, and GUI input information 1903 are sorted in the order of time stamps, and are processed in turn from input information with an earlier time stamp. In case
25 of Fig. 19, these pieces of input information are sorted in the order of 1901, 1902, and 1903.

The speech input information 1901 cannot be processed as a single input and requires an integration process, since its value is "@unknown". As information to be integrated, GUI input information input before

5 the speech input information 1901 is searched for an input that similarly requires an integration process (S912, S914). In this case, since there is no GUI input information input before the speech input information 1901, the integration process is skipped,

10 and the process of the next speech input information 1902 starts while holding information (S915, S918 - S920). Since all the data bind destination, semantic attribute, and value of the speech input information 1902 are settled, the data bind destination "/Num" and

15 value "2" are output as a single input (Fig. 19: 1904) (S912, S913). Next, the process of the GUI input information 1903 starts (S920). The GUI input information 1903 cannot be processed as a single input and requires an integration process, since its data

20 model is "- (no bind)". As information to be integrated, speech input information input before the GUI input information 1903 is searched for input information that satisfies the integration condition (S912, S914). In this case, the speech input

25 information 1901 does not satisfy the integration conditions, since the input information 1902 with a different semantic attribute is present between them.

Hence, the integration process is skipped, and the next process starts while holding the information (S915, S918 - S920).

As described above, since the integration process is executed based on the time stamps and semantic attributes, a plurality of pieces of input information from respective input modalities can be normally integrated. As a result, when the application developer sets a common semantic attribute in inputs to be integrated, his or her intention can be reflected on the application.

As described above, according to the first embodiment, an XML document and grammar (rules of grammar) for speech recognition can describe a semantic attribute, and the intention of the application developer can be reflected on the system. When the system that comprises the multimodal user interface exploits the semantic attribute information, multimodal inputs can be efficiently integrated.

20 [Second Embodiment]

The second embodiment of an information processing system according to the present invention will be described below. In the example of the aforementioned first embodiment, one semantic attribute is designated for one input information (GUI component or input speech). The second embodiment will exemplify

a case wherein a plurality of semantic attributes can be designated for one input information.

Fig. 20 shows an example of an XHTML document used to present respective GUI components in the information processing system according to the second embodiment. In Fig. 20, an <input> tag, type attribute, value attribute, ref attribute, and class attribute are described by the same description method as that of Fig. 3 in the first embodiment. However, unlike in the first embodiment, the class attribute describes a plurality of semantic attributes. For example, a button having a value "TOKYO" describes "station area" in its class attribute. The markup parsing unit 106 parses this class attribute as two semantic attributes "station" and "area" which have a white space character as a delimiter. More specifically, a plurality of semantic attributes can be described by delimiting them using a space.

Fig. 21 shows grammar (rules of grammar) required to recognize speech. The grammar in Fig. 21 is described by the same description method as that in Fig. 7, and describes rules required for recognizing speech inputs "weather of here", "weather of TOKYO", and the like, and outputting an interpretation result such as area="@unknown". Fig. 22 shows an example of the interpretation result obtained when both the grammar (rules of grammar) shown in Fig. 21 and that

shown in Fig. 7 are used. For example, when a speech processor connected to a network is used, the interpretation result is obtained as an XML document shown in Fig. 22. Fig. 22 is described by the same description method as that in Fig. 7. According to
5 Fig. 22, the confidence level of "weather of here" is 80, and that of "from here" is 20.

The processing method upon integrating a plurality of pieces of input information each having a
10 plurality of semantic attributes will be described below taking Fig. 23 as an example. In Fig. 23, "DataModel" of GUI input information 2301 is a data bind destination, "value" is a value, "meaning" is a semantic attribute, "ratio" is the confidence level of
15 each semantic attribute, and "c" is the confidence level of the value. These "DataModel", "value", "meaning", and "ratio" are obtained by parsing the XML document shown in Fig. 20 by the markup parsing unit 106. Note that "ratio" of these data assumes a value
20 obtained by dividing 1 by the number of semantic attributes if it is not specified in the meaning attribute (or class attribute) (hence, for TOKYO, "ratio" of each of station and area is 0.5). Also, "c" is the confidence level of the value, and this value is
25 calculated by the application when the value is input. For example, in case of the GUI input information 2301, "c" is the confidence level when a point at which the

probability that the value is TOKYO is 90% and the probability that the value is KANAGAWA is 10% is designated (for example, when a point on a map is designated by drawing a circle with a pen, and that
5 circle includes TOKYO 90% and KANAGAWA 10%).

Also, in Fig. 23, "c" of speech input information 2302 is the confidence level of a value, which uses a normalization likelihood (recognition score) for each recognition candidate. The speech input information
10 2302 is an example when the normalization likelihood (recognition score) of "weather of here" is 80 and that of "from here" is 20. Fig. 23 does not describe any time stamp, but the time stamp information is utilized as in the first embodiment.

15 The integration conditions according to the second embodiment include:

(1) the plurality of pieces of information require an integration process;

(2) the plurality of pieces of information are
20 input within a time limit (e.g., the time stamp difference is 3 sec or less);

(3) at least one of semantic attributes of information matches that of information to be integrated;

25 (4) the plurality of pieces of information do not include any input information having semantic

attributes, none of which match, when they are sorted in the order of time stamps;

(5) "bind destination" and "value" have a complementary relationship; and

- 5 (6) information, which is input earliest, of those which satisfy (1) to (4), is to be integrated. Note that the integration conditions are an example, and other conditions may be set. Also, some of the above integration conditions may be used as the
- 10 integration conditions (for example, only conditions (1) and (3) are used as the integration conditions). In this embodiment as well, inputs of different modalities are integrated, but inputs of an identical modality are not integrated.
- 15 The integration process of the second embodiment will be described below using Fig. 23. The GUI input information 2301 is converted into GUI input information 2303 to have a confidence level "cc" obtained by multiplying the confidence level "c" of the
- 20 value and the confidence level "ratio" of the semantic attribute in Fig. 23. Likewise, the speech information 2302 is converted into speech input information 2304 to have a confidence level "cc" obtained by multiplying the confidence level "c" of the value and the
- 25 confidence level "ratio" of the semantic attribute in Fig. 23 (in Fig. 23, the confidence level of the semantic attribute is "1" since each speech recognition

result has only one semantic attribute; for example, when a speech recognition result "TOKYO" is obtained, it includes semantic attributes "station" and "area", and their confidence levels are 0.5). The integration
5 method of respective pieces of speech input information is the same as that in the first embodiment. However, since one input information includes a plurality of semantic attributes and a plurality of values, a plurality of integration candidates are likely to
10 appear in step S916, as indicated by 2305 in Fig. 23.

Next, a value obtained by multiplying the confidence levels of matched semantic attributes is set as a confidence level "ccc" in the GUI input information 2303 and speech input information 2304 to
15 generate a plurality of pieces of input information 2305. Of the plurality of pieces of input information 2305, input information with the highest confidence level (ccc) is selected, and a bind destination "/Area" and value "TOKYO" of the selected data (data of ccc =
20 3600 in this example) are output (Fig. 23: 2306). If a plurality of pieces of information have the same confidence level, information which is processed first is preferentially selected.

A description example of the confidence level
25 (ratio) of the semantic attribute using the markup language will be explained. In Fig. 24, semantic attributes are designated in the class attribute as in

Fig. 22. In this case, colon (:) and the confidence level are appended to each semantic attribute. As shown in Fig. 24, a button having a value "TOKYO" has semantic attributes "station" and "area", the
5 confidence level of the semantic attribute "station" is "55", and that of the semantic attribute "area" is "45". The markup parsing unit 106 (XML parser) separately parses the semantic attribute and confidence level, and outputs the confidence level of the semantic attribute
10 as "ratio" in GUI input information 2501 in Fig. 25. In Fig. 25, the same process as in Fig. 23 is done to output a data bind destination "/Area" and value "TOKYO" (Fig. 25: 2506).

In Figs. 24 and 25, only one semantic attribute
15 is described in the grammar (rules of grammar) for speech recognition for the sake of simplicity. However, as shown in Fig. 26, a plurality of semantic attributes may be designated by a method using, e.g., List type. As shown in Fig. 26, an input "here" has a value
20 "@unknown", semantic attributes "area" and "country", the confidence level "90" of the semantic attribute "area", and the confidence level "10" of the semantic attribute "country".

In this case, the integration process is executed,
25 as shown in Fig. 27. The output from the speech recognition/interpretation unit 103 has contents 2602. The multimodal input integration unit 104 calculates

confidence levels ccc, as indicated by 2605. As for the semantic attribute "country", since no input from the GUI input unit 101 has the same semantic attribute, its confidence level is not calculated.

5 Figs. 23 and 25 show examples of the integration process based on the confidence levels described in the markup language. Alternatively, the confidence level may be calculated based on the number of matched semantic attributes of input information having a
10 plurality of semantic attributes, and information with the highest confidence level may be selected. For example, if GUI input information having three semantic attributes A, B, and C, GUI input information having three semantic attributes A, D, and E, and speech input
15 information having four semantic attributes A, B, C, and D are to be integrated, the number of common semantic attributes between the GUI input information having semantic attributes A, B, and C and the speech input information having semantic attributes A, B, C,
20 and D is 3. On the other hand, the number of common semantic attributes between the GUI input information having semantic attributes A, D, and E and the speech input information having semantic attributes A, B, C, and D is 2. Hence, the number of common semantic
25 attributes is used as the confidence level, and the GUI input information having semantic attributes A, B, and C, and speech input information A, B, C, and D, which

have the high confidence level, are integrated and output.

As described above, according to the second embodiment, an XML document and grammar (rules of grammar) for speech recognition can describe a plurality of semantic attributes, and the intention of the application developer can be reflected on the system. When the system that comprises the multimodal user interface exploits the semantic attribute information, multimodal inputs can be efficiently integrated.

As described above, according to the above embodiments, an XML document and grammar (rules of grammar) for speech recognition can describe a semantic attribute, and the intention of the application developer can be reflected on the system. When the system that comprises the multimodal user interface exploits the semantic attribute information, multimodal inputs can be efficiently integrated.

As described above, according to the present invention, since a description required to process inputs from a plurality of types of input modalities adopts a description of a semantic attribute, integration of inputs that the user or developer intended can be implemented by a simple analysis process.

Furthermore, the invention can be implemented by supplying a software program, which implements the functions of the foregoing embodiments, directly or indirectly to a system or apparatus, reading the
5 supplied program code with a computer of the system or apparatus, and then executing the program code. In this case, so long as the system or apparatus has the function of the program, the mode of implementation need not rely upon a program.

10 Accordingly, since the functions of the present invention are implemented by computer, the program code installed in the computer also implements the present invention. In other words, the claims of the present invention also cover a computer program for the purpose
15 of implementing the functions of the present invention.

In this case, so long as the system or apparatus has the functions of the program, the program may be executed in any form, such as an object code, a program executed by an interpreter, or script data supplied to
20 an operating system.

Examples of storage media that can be used for supplying the program are a floppy disk, a hard disk, an optical disk, a magneto-optical disk, a CD-ROM, a CD-R, a CD-RW, a magnetic tape, a non-volatile type
25 memory card, a ROM, and a DVD (DVD-ROM and a DVD-R).

As for the method of supplying the program, a client computer can be connected to a website on the

Internet using a browser of the client computer, and the computer program of the present invention or an automatically-installable compressed file of the program can be downloaded to a recording medium such as a hard disk. Further, the program of the present invention can be supplied by dividing the program code constituting the program into a plurality of files and downloading the files from different websites. In other words, a WWW (World Wide Web) server that downloads, to multiple users, the program files that implement the functions of the present invention by computer is also covered by the claims of the present invention.

It is also possible to encrypt and store the program of the present invention on a storage medium such as a CD-ROM, distribute the storage medium to users, allow users who meet certain requirements to download decryption key information from a website via the Internet, and allow these users to decrypt the encrypted program by using the key information, whereby the program is installed in the user computer.

Besides the cases where the aforementioned functions according to the embodiments are implemented by executing the read program by computer, an operating system or the like running on the computer may perform all or a part of the actual processing so that the

functions of the foregoing embodiments can be implemented by this processing.

Furthermore, after the program read from the storage medium is written to a function expansion board
5 inserted into the computer or to a memory provided in a function expansion unit connected to the computer, a CPU or the like mounted on the function expansion board or function expansion unit performs all or a part of the actual processing so that the functions of the
10 foregoing embodiments can be implemented by this processing.

As many apparently widely different embodiments of the present invention can be made without departing from the spirit and scope thereof, it is to be
15 understood that the invention is not limited to the specific embodiments thereof except as defined in the appended claims.

CLAIMS

1. An information processing method for recognizing a user's instruction on the basis of a plurality of pieces of input information which are
5 input by a user using a plurality of types of input modalities,

said method having a description including correspondence between input contents and a semantic attribute for each of the plurality of types of input
10 modalities,

said method comprising: an acquisition step of acquiring an input content by parsing each of the plurality of pieces of input information which are input using the plurality of types of input modalities,
15 and acquiring semantic attributes of the acquired input contents from the description; and

an integration step of integrating the input contents acquired in the acquisition step on the basis of the semantic attributes acquired in the acquisition
20 step.

2. The method according to claim 1, wherein one of the plurality of types of input modalities is an instruction of a component via a GUI,

the description includes a description of
25 correspondence between respective components of the GUI and semantic attributes, and

the acquisition step includes a step of detecting an instructed component as an input content, and acquiring a semantic attribute corresponding to the instructed component from the description.

5 3. The method according to claim 2, wherein the description describes the GUI using a markup language.

4. The method according to claim 1, wherein one of the plurality of types of input modalities is a speech input,

10 the description includes a description of correspondence between speech inputs and semantic attributes, and

the acquisition step includes a step of applying a speech recognition process to speech information to
15 obtain input speech as an input content, and acquiring a semantic attribute corresponding to the input speech from the description.

5. The method according to claim 4, wherein the description includes a description of a grammar rule
20 for speech recognition, and

the speech recognition step includes a step of applying the speech recognition process to the speech information with reference to the description of the grammar rule.

25 6. The method according to claim 5, wherein the grammar rule is described using a markup language.

7. The method according to claim 1, wherein the acquisition step includes a step of further acquiring an input time of the input content, and

the integration step includes a step of
5 integrating a plurality of input contents on the basis of the input times of the input contents, and the semantic attributes acquired in the acquisition step.

8. The method according to claim 7, wherein the acquisition step includes a step of acquiring
10 information associated with a value and bind destination of the input content, and

the integration step includes a step of checking based on the information associated with the value and bind destination of the input content if integration is
15 required, outputting, if integration is not required, the input contents intact, integrating the input contents, which require integration, on the basis of the input times and semantic attributes, and outputting the integration result.

20 9. The method according to claim 8, wherein the integration step includes a step of integrating the input contents which have a input time difference that falls within a predetermined range, and matched semantic attributes, of the input contents that require
25 integration.

10. The method according to claim 8, wherein the integration step includes a step of outputting, when

the input contents or the integration result, which have the input time difference that falls within the predetermined range and the same bind destination, are to be output, the input contents or integration result
5 in the order of input times.

11. The method according to claim 8, wherein the integration step includes a step of selecting, when the input contents or the integration result, which have the input time difference that falls within the
10 predetermined range and the same bind destination, are to be output, the input content or integration result, which is input according to an input modality with higher priority, in accordance with priority of input modalities, which is set in advance, and outputting the
15 selected input content or integration result.

12. The method according to claim 8, wherein the integration step includes a step of integrating input contents in ascending order of input time.

13. The method according to claim 8, wherein the
20 integration step includes a step of inhibiting integration of input contents which include input contents with a different semantic attribute when the input contents are sorted in the order of input times.

14. The method according to claim 1, wherein the
25 description describes a plurality of semantic attributes for one input content, and

the integration step includes a step of determining, when a plurality of types of information are likely to be integrated on the basis of the plurality of semantic attributes, input contents to be
5 integrated on the basis of weights assigned to the respective semantic attributes.

15. The method according to claim 1, wherein the integration step includes a step of determining, when a plurality of input contents are acquired for input
10 information in the acquisition step, input contents to be integrated on the basis of confidence levels of the input contents in parsing.

16. An information processing apparatus for recognizing a user's instruction on the basis of a
15 plurality of pieces of input information which are input by a user using a plurality of types of input modalities, comprising:

a holding unit for holding a description including correspondence between input contents and a
20 semantic attribute for each of the plurality of types of input modalities,

an acquisition unit for acquiring an input content by parsing each of the plurality of pieces of input information which are input using the plurality
25 of types of input modalities, and acquiring semantic attributes of the acquired input contents from the description; and

an integration unit for integrating the input contents acquired by said acquisition unit on the basis of the semantic attributes acquired by said acquisition unit.

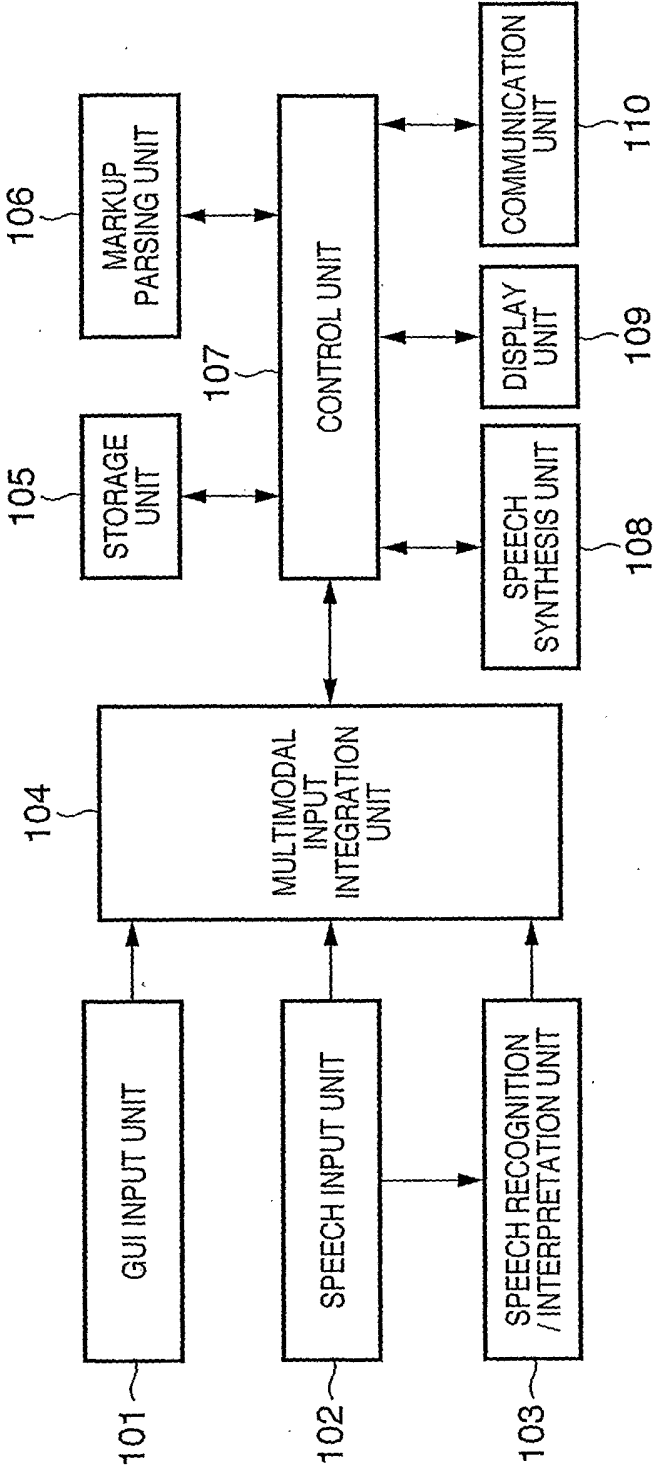
5 17. A description method of describing a GUI, characterized by describing semantic attributes corresponding to respective GUI components using a markup language.

10 18. A grammar rule for recognizing speech input information input by speech, characterized by describing semantic attributes corresponding to respective speech inputs in the grammar rule.

15 19. A storage medium storing a control program for making a computer execute an information processing method of claim 1.

20. A control program for making a computer execute an information processing method of claim 1.

FIG. 1



2/28

FIG. 2

.

.

.

```
<input type="button" value="SHIBUYA" ct:meaning="station" />  
<input type="button" value="EBISU" ct:meaning="station" />  
<input type="button" value="JIYUGAOKA" ct:meaning="station" xf:ref="/ To" />  
<input type="button" value="1" ct:meaning="number" xf:ref="/ Num" />  
<input type="button" value="2" ct:meaning="number" />
```

.

.

.

3/28

FIG. 3

.

.

.

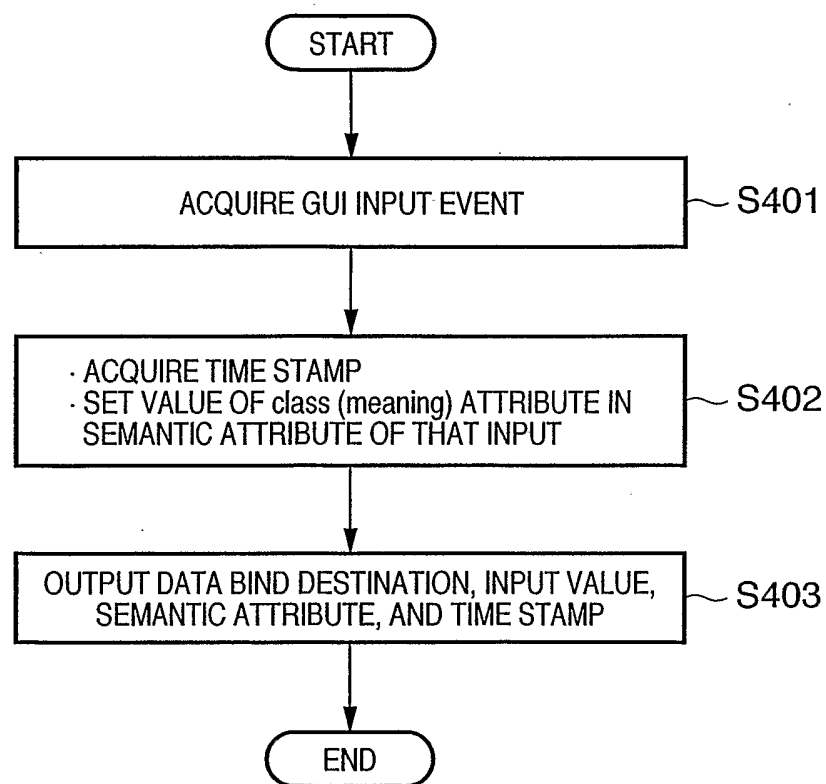
```
<input type="button" value="SHIBUYA" class="station" />
<input type="button" value="EBISU" class="station" />
<input type="button" value="JIYUGAOKA" class="station" xf:ref="/ To" />
<input type="button" value="1" class="number" xf:ref="/ Num" />
<input type="button" value="2" class="number" />
```

.

.

.

4/28

FIG. 4

5/28

FIG. 5

DataModel	meaning	value	input
—	from	—	from
—	to	—	to
/From	station	@unknown	here
/To	station	@unknown	here
/From	station	EBISU	ebise
/To	station	EBISE	ebise
...	...	...	...

6/28

FIG. 6

```

<?xml version="1.0" encoding="Shift_JIS"?>
<!DOCTYPE grammar SYSTEM "http://www.w3.org/
TR/speech-grammar/grammar.dtd">
<grammar xmlns="http://www.w3.org/2001/06/grammar"
xml:lang="ja" version="1.0"
mode="voice" root="from_to">
  <rule id="from_to">
    <ruleref uri="#from"/>
    <ruleref uri="#to"/>
    <tag>
      $.from=#from.res;
      $.to=#to.res;
    </tag>
  </rule>
  <rule id="from">
    <ruleref uri="#stations"/>
    <tag>
      $.res=$stations;
    </tag>
    <item repeat="0-1"> from </item>
  </rule>
  <rule id="to">
    <ruleref uri="#stations"/>
    <tag>
      $.res=$stations;
    </tag>
    <item repeat="0-1"> to </item>
  </rule>
  <rule id="stations">
    <one-of>
      <item> here <tag>$="@unknown : station" </tag> </item>
      <item> shibuya <tag>$="SHIBUYA : station" </tag> </item>
      <item> ebisu <tag>$="EBISU : station" </tag> </item>
      <item> jiyugaoka <tag>$="JIYUGAOKA : station" </tag> </item>
      <item> tokyo <tag>$="TOKYO : station" </tag> </item>
      ...
    </one-of>
  </rule>
</grammar>

```

7/28

FIG. 7

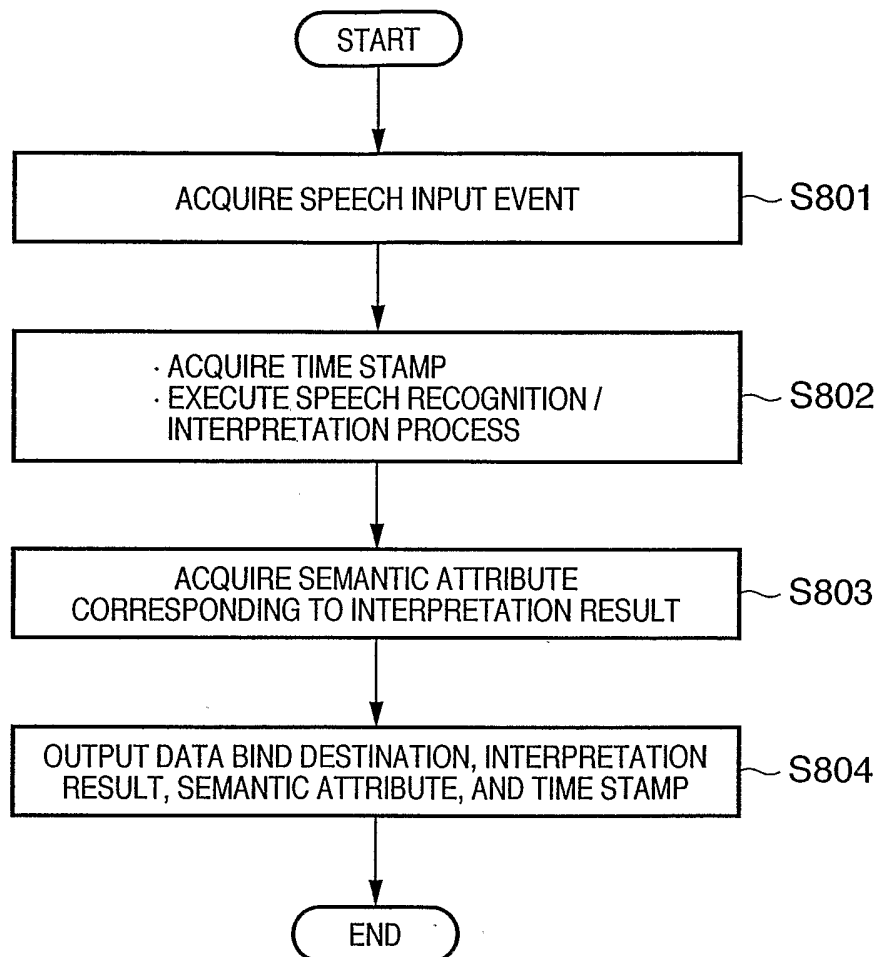
```
<dsr:SpeechRecognitionResponse>
  <nlsml:result>

    <nlsml:interpretation confidence="80">
      <nlsml:input> from here to jiyugaoka </nlsml:input>
      <nlsml:instance>
        <my:From>@unknown : station</my:From>
        <my:To> JIYUGAOKA : station</my:To>
      </nlsml:instance>
    </nlsml:interpretation>

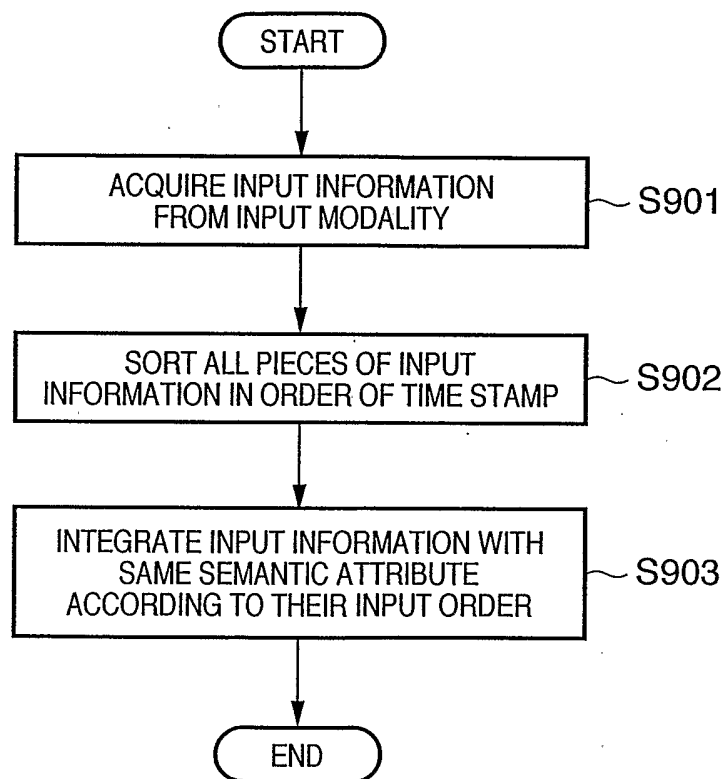
    <nlsml:interpretation confidence="20">
      <nlsml:input> from here to here </nlsml:input>
      <nlsml:instance>
        <my:From>@unknown : station</my:From>
        <my:To>@unknown : station</my:To>
      </nlsml:instance>
    </nlsml:interpretation>

  </nlsml:result>
</dsr:SpeechRecognitionResponse>
```

8/28

FIG. 8

9/28

FIG. 9A

10/28

FIG. 9B

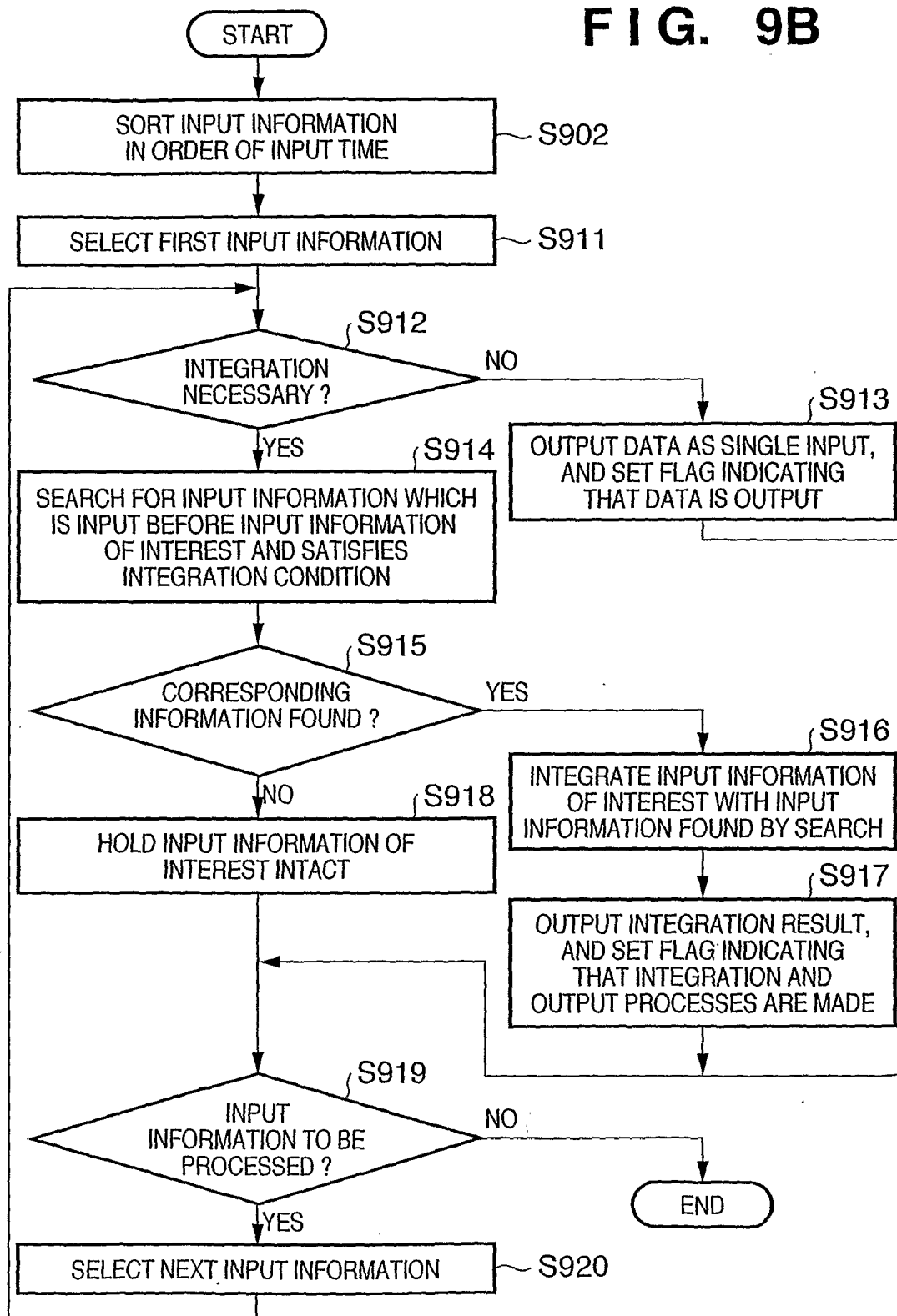


FIG. 10

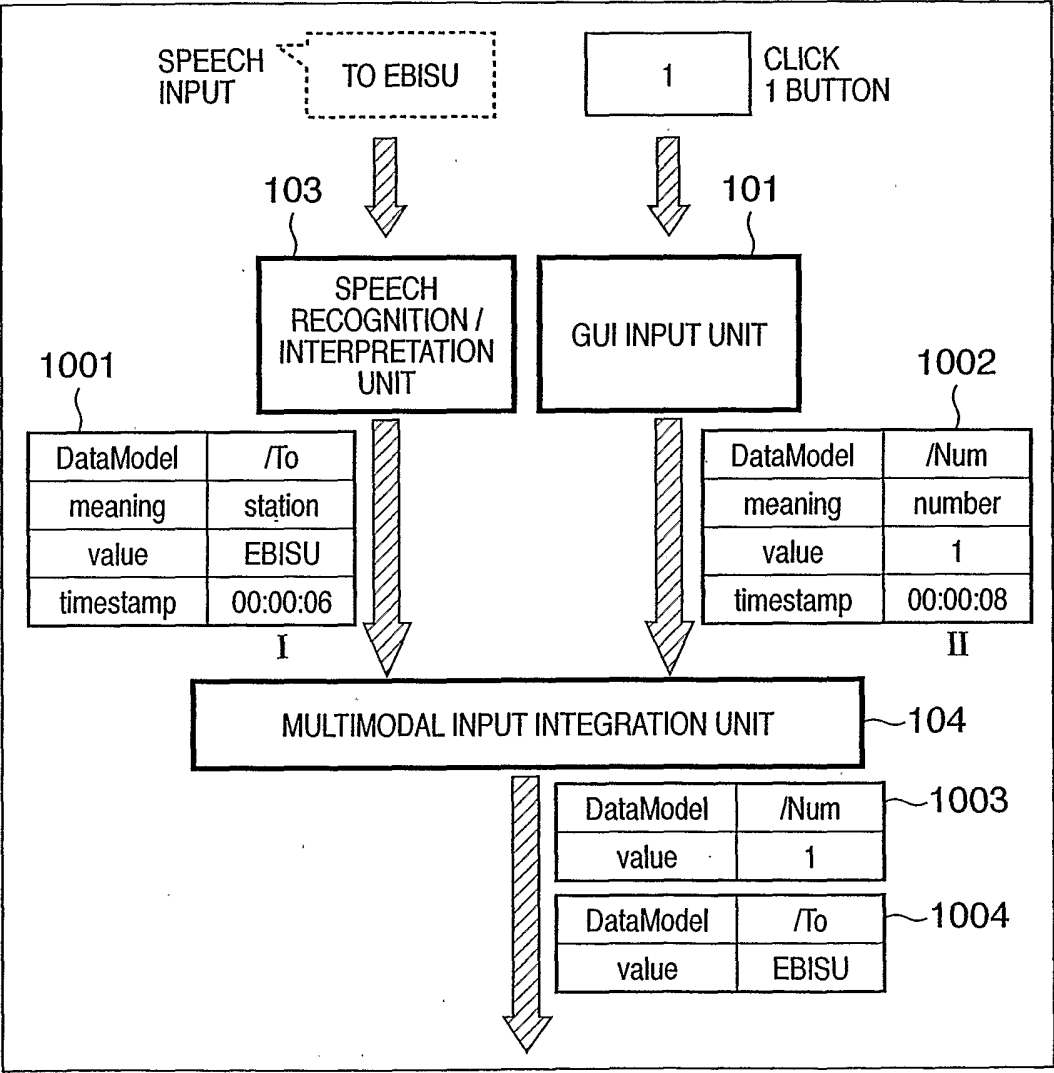


FIG. 11

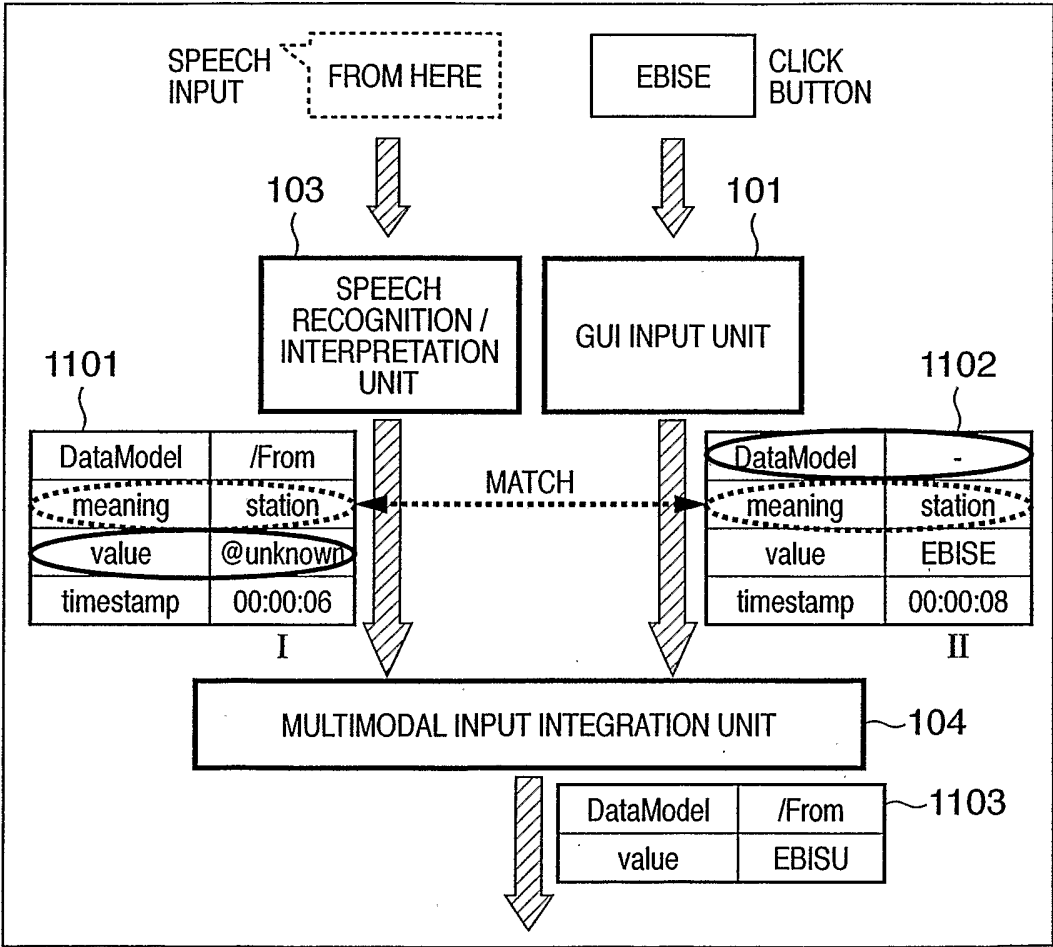


FIG. 12

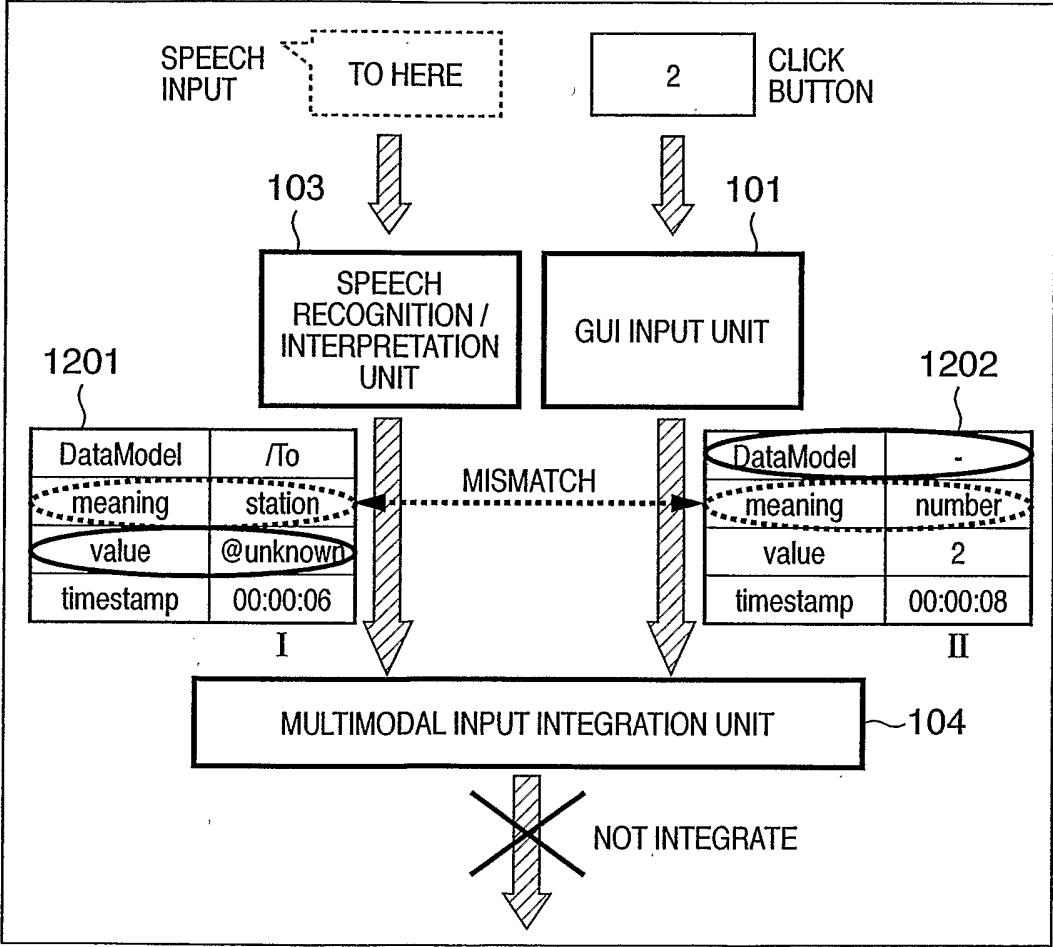


FIG. 13

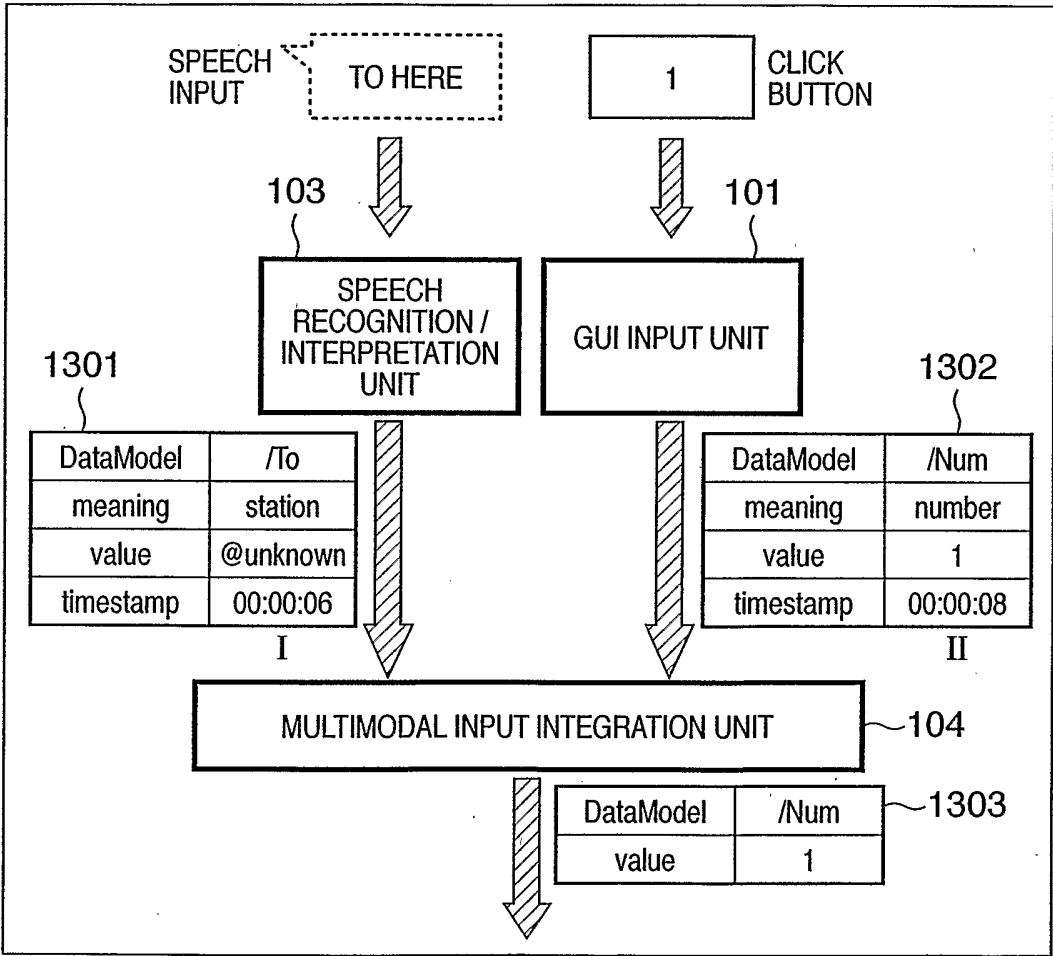


FIG. 14

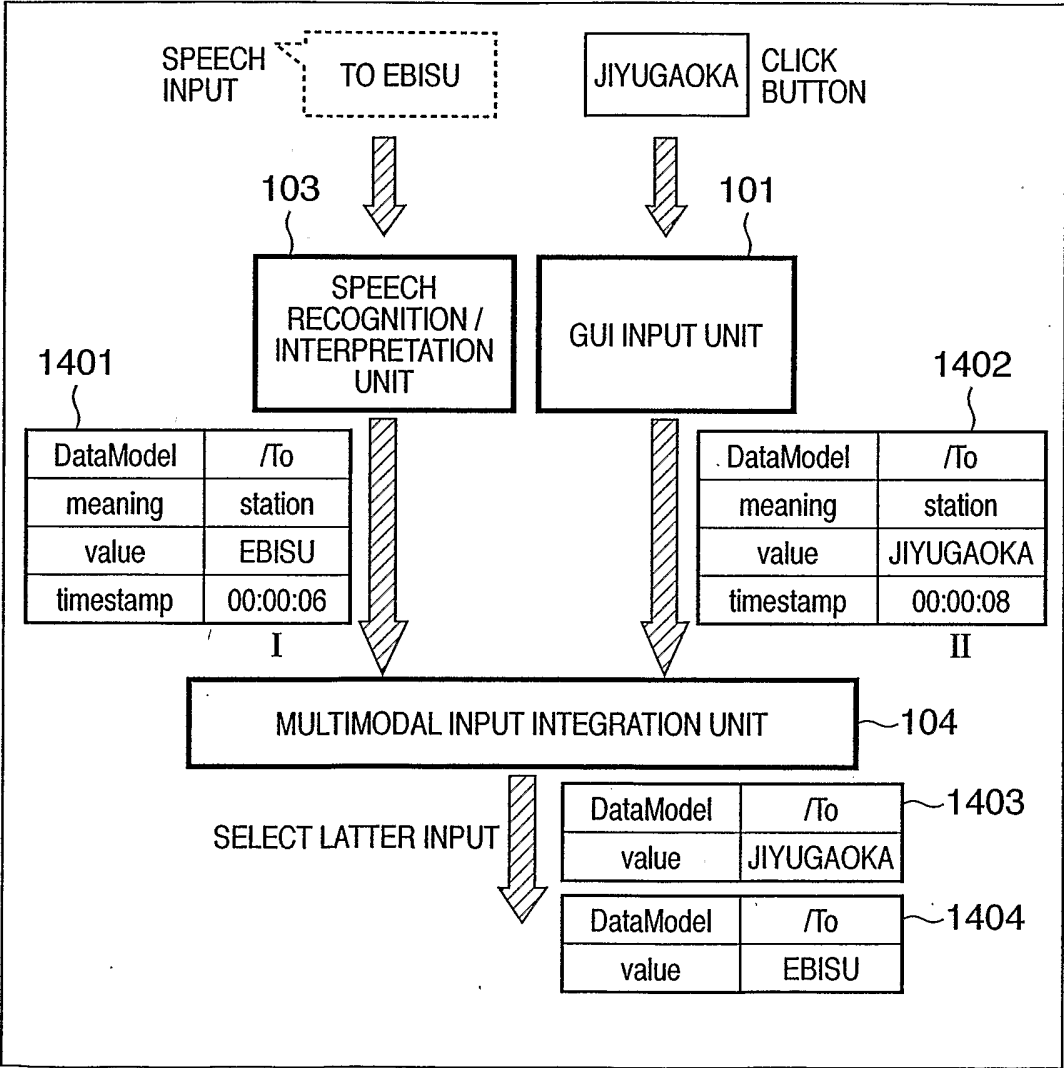
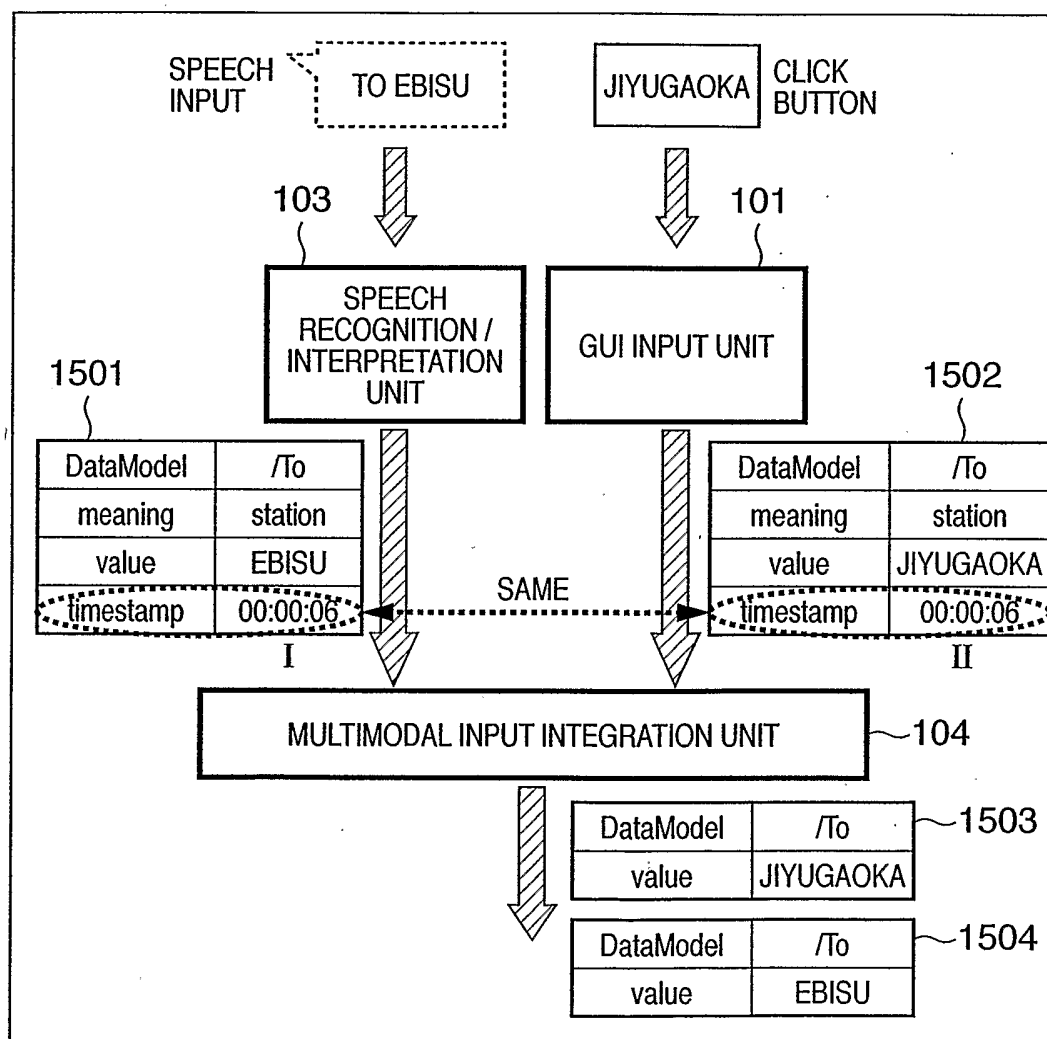


FIG. 15



17/28

FIG. 16

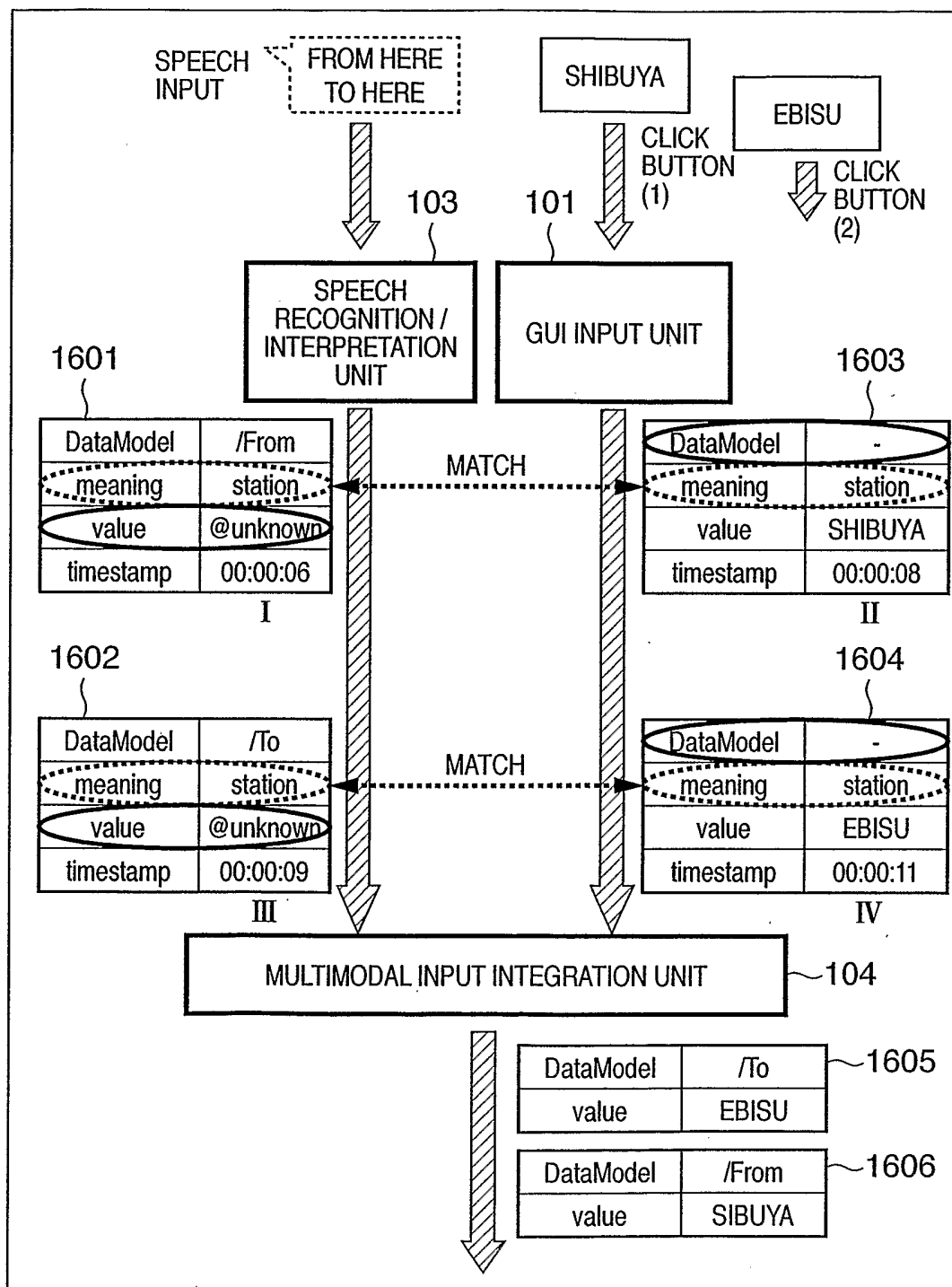
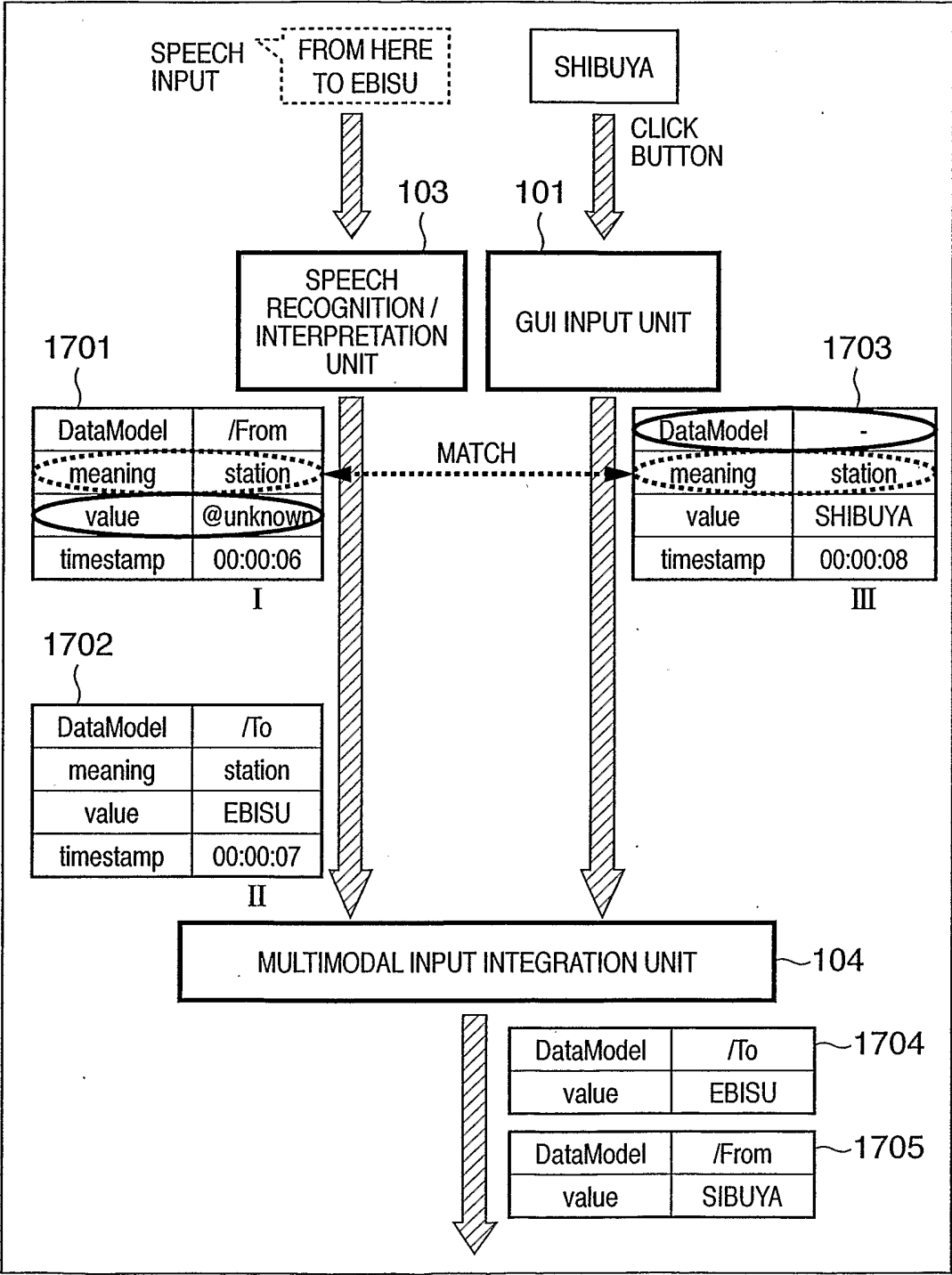
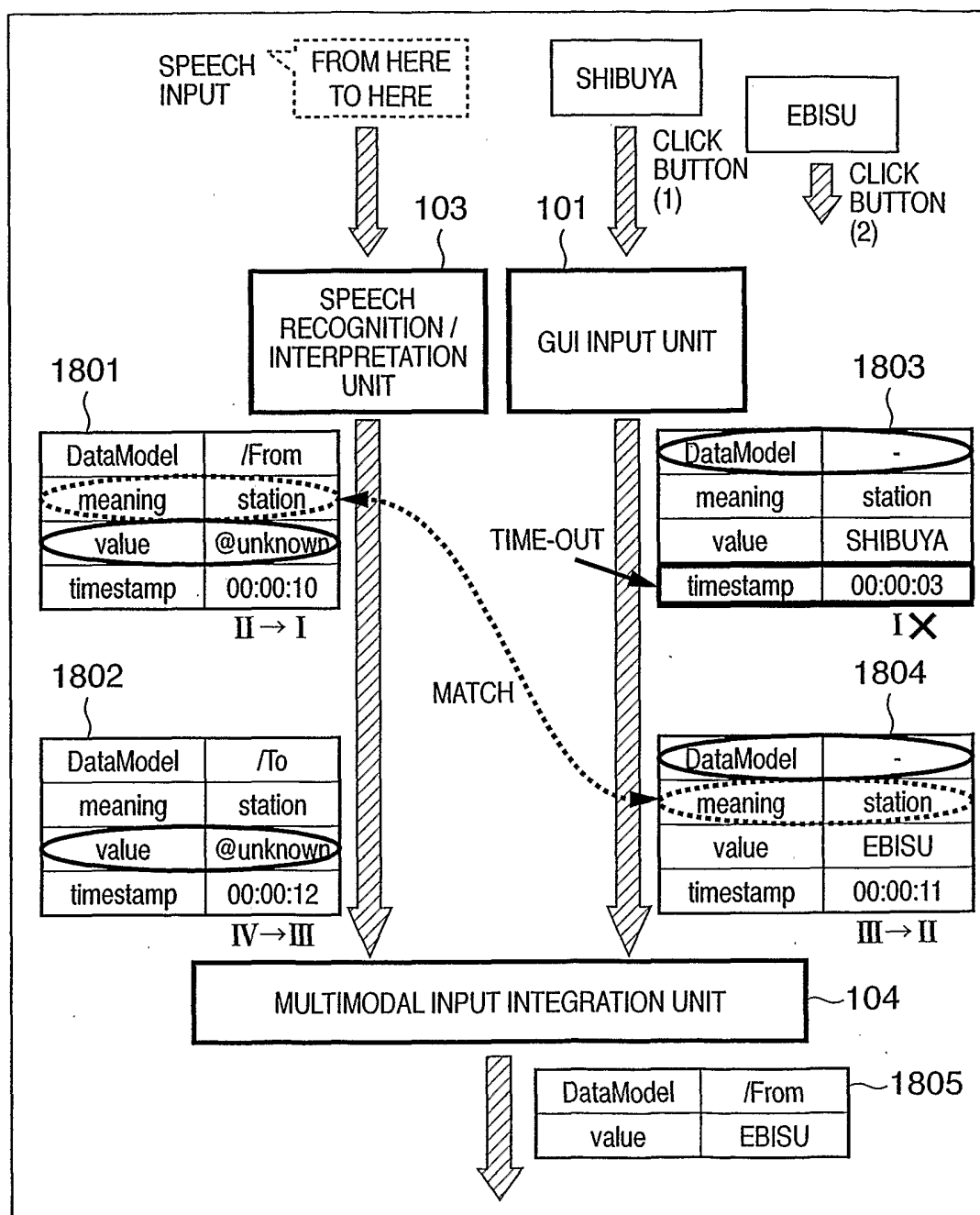


FIG. 17



19/28

FIG. 18



20/28

FIG. 19

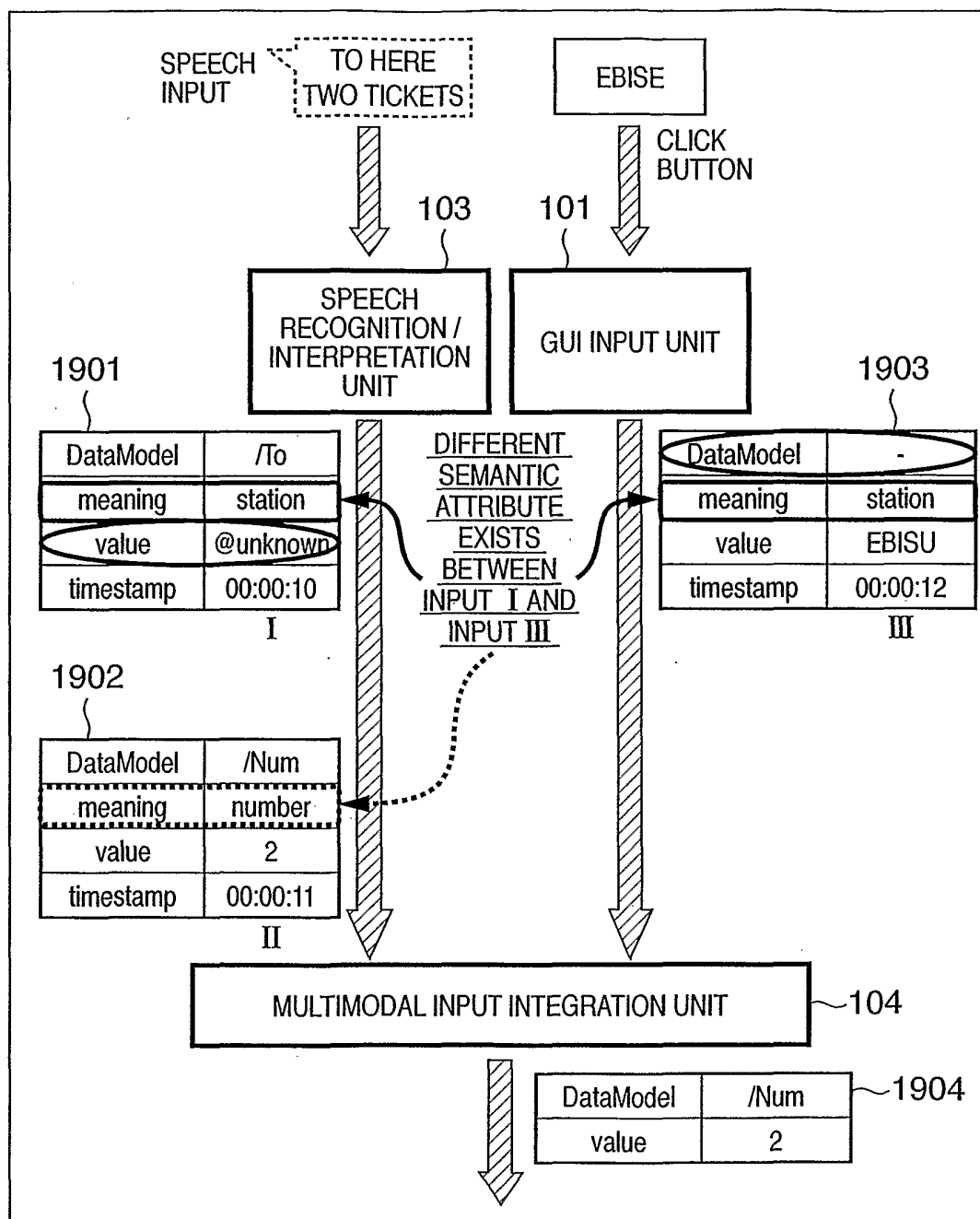


FIG. 20

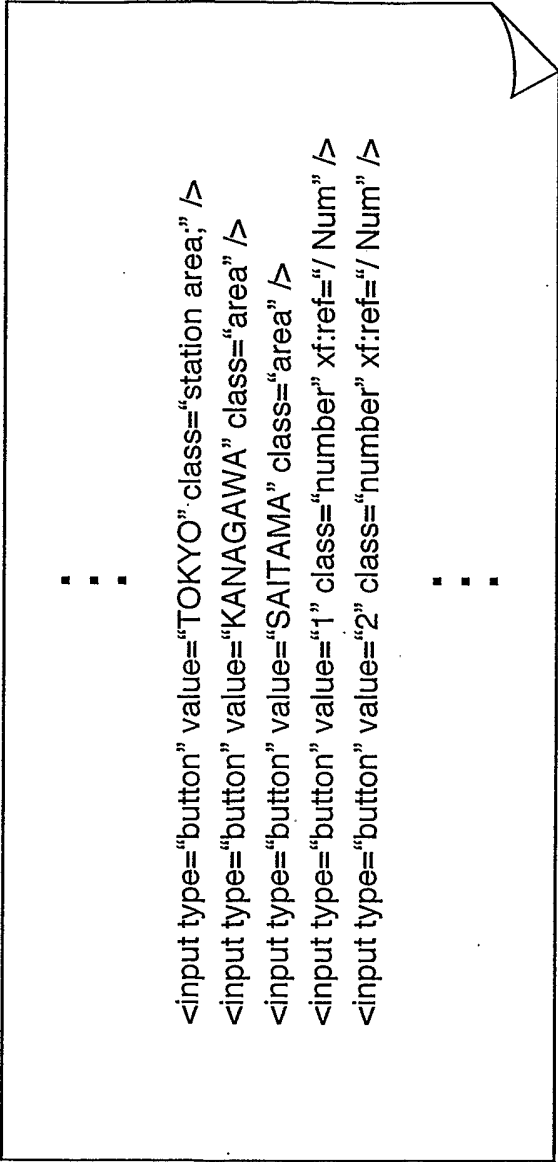


FIG. 21

```

<?xml version="1.0" encoding="Shift_JIS"?>
<!DOCTYPE grammar SYSTEM "http://www.w3.org/TR/speech-grammar/grammar.dtd">
<grammar xmlns="http://www.w3.org/2001/06/grammar" xml:lang="ja" version="1.0"
mode="voice" root="weather">
  <rule id="weather">
    <ruleref uri="#area"/>
  </tag>
  <tag>
    $.res=$area;
  </tag>
  <item repeat="0-1"> weather of </item>
</rule>
<rule id="area">
  <one-of>
    <item> here <tag>$="@unknown : area" </tag> </item>
    <item> tokyo <tag>$="TOKYO : area" </tag> </item>
    <item> kanagawa <tag>$="KANAGAWA : area" </tag> </item>
    <item> saitama <tag>$="SAITAMA : area" </tag> </item>
    <item> chiba <tag>$="CHIBA : area" </tag> </item>
    ...
  </one-of>
</rule>
</grammar>

```

FIG. 22

```
<dsr:SpeechRecognitionResponse>
  <nlsml:result>

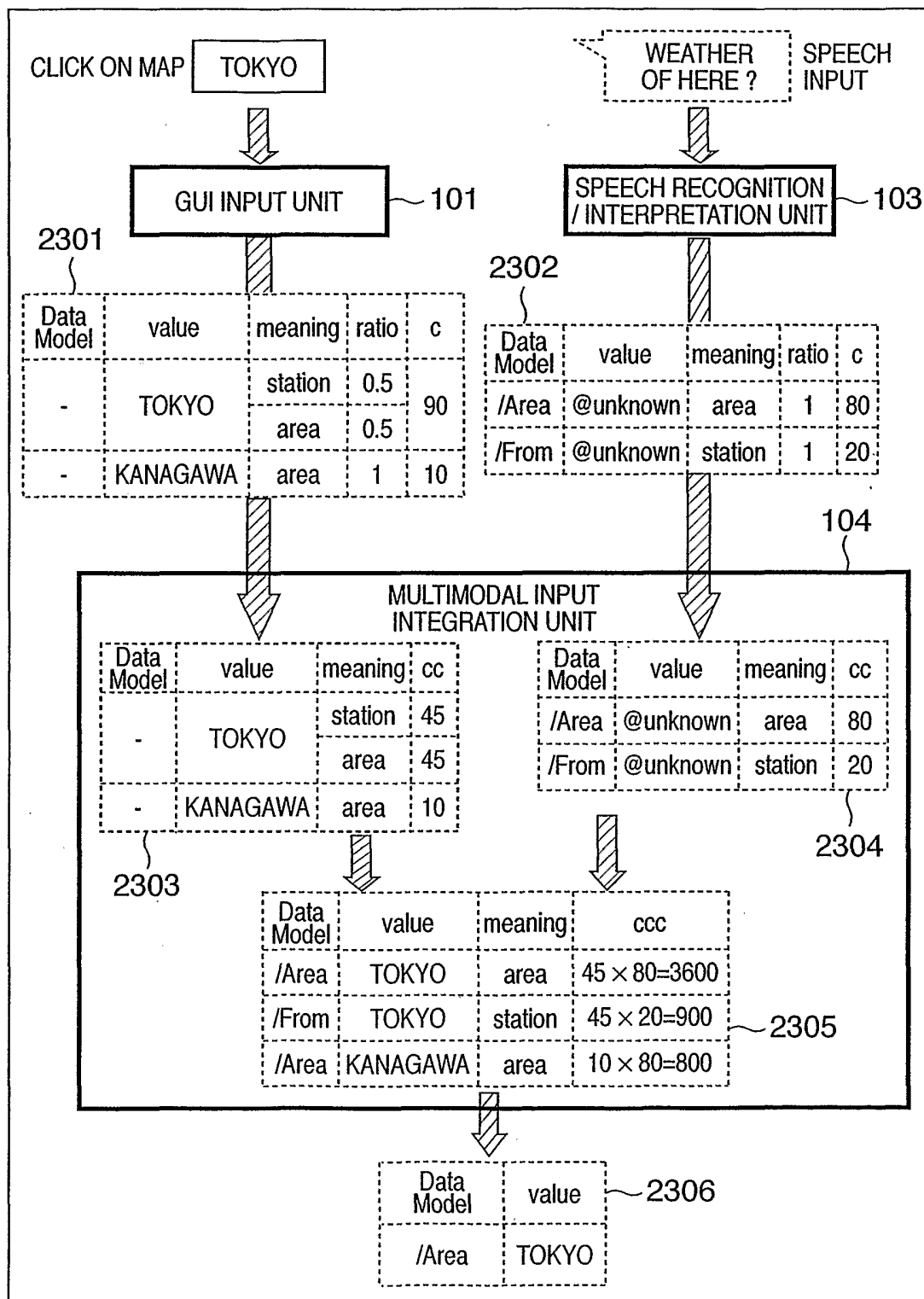
    <nlsml:interpretation confidence="80">
      <nlsml:input> weather of here </nlsml:input>
      <nlsml:instance>
        <my:Area>@unknown : area</my:Area>
      </nlsml:instance>
    </nlsml:interpretation>

    <nlsml:interpretation confidence="20">
      <nlsml:input> from here </nlsml:input>
      <nlsml:instance>
        <my:From>@unknown : station</my:From>
      </nlsml:instance>
    </nlsml:interpretation>

  </nlsml:result>
</dsr:SpeechRecognitionResponse>
```

24/28

FIG. 23



25/28

FIG. 24

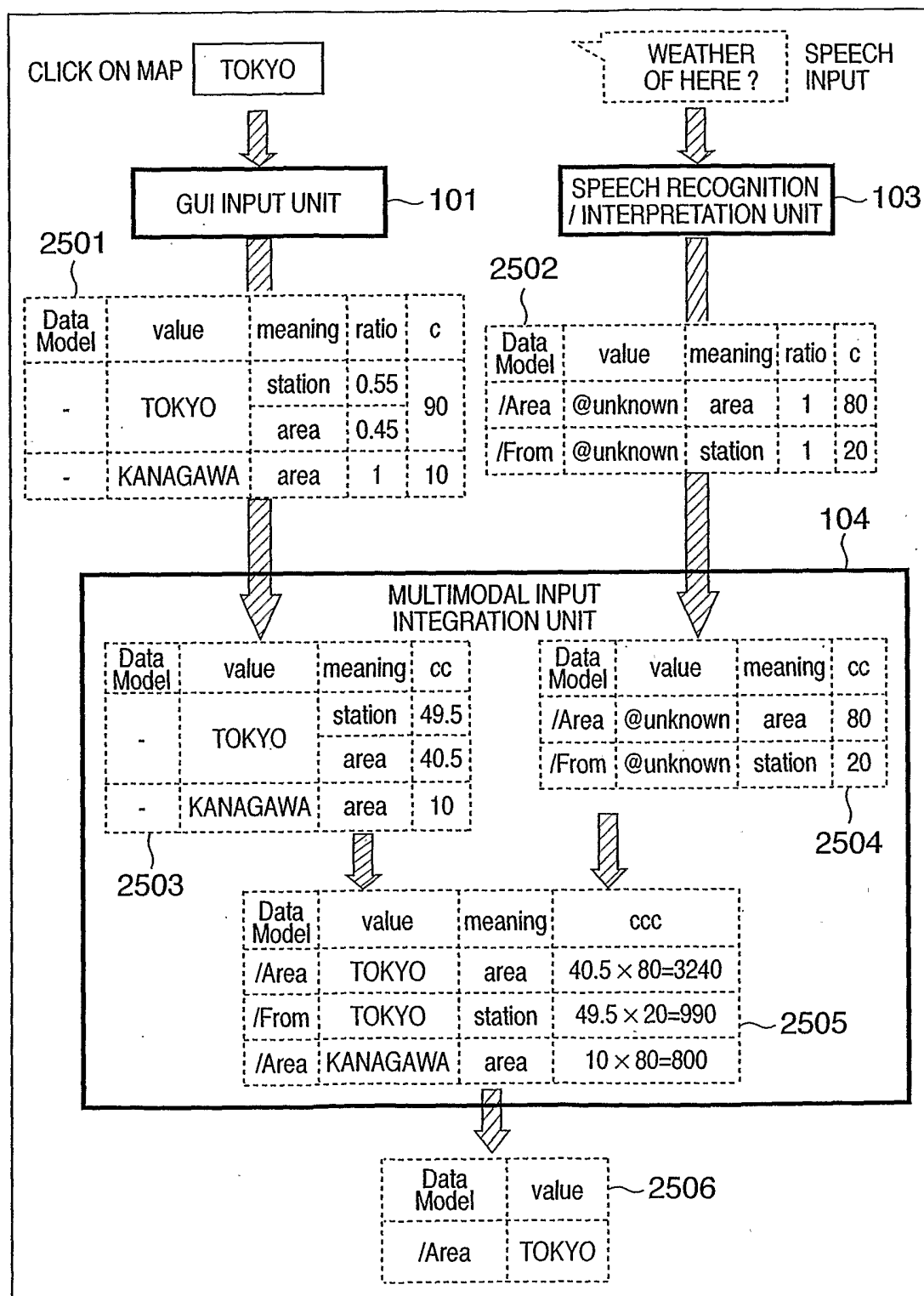
:

```
<input type="button" value="TOKYO" class="station:55 area:45;" />  
<input type="button" value="KANAGAWA" class="area" />  
<input type="button" value="SAITAMA" class="area" />  
<input type="button" value="1" class="number" xf:ref="/ Num" />  
<input type="button" value="2" class="number" xf:ref="/ Num" />
```

:

26/28

FIG. 25



27/28

FIG. 26

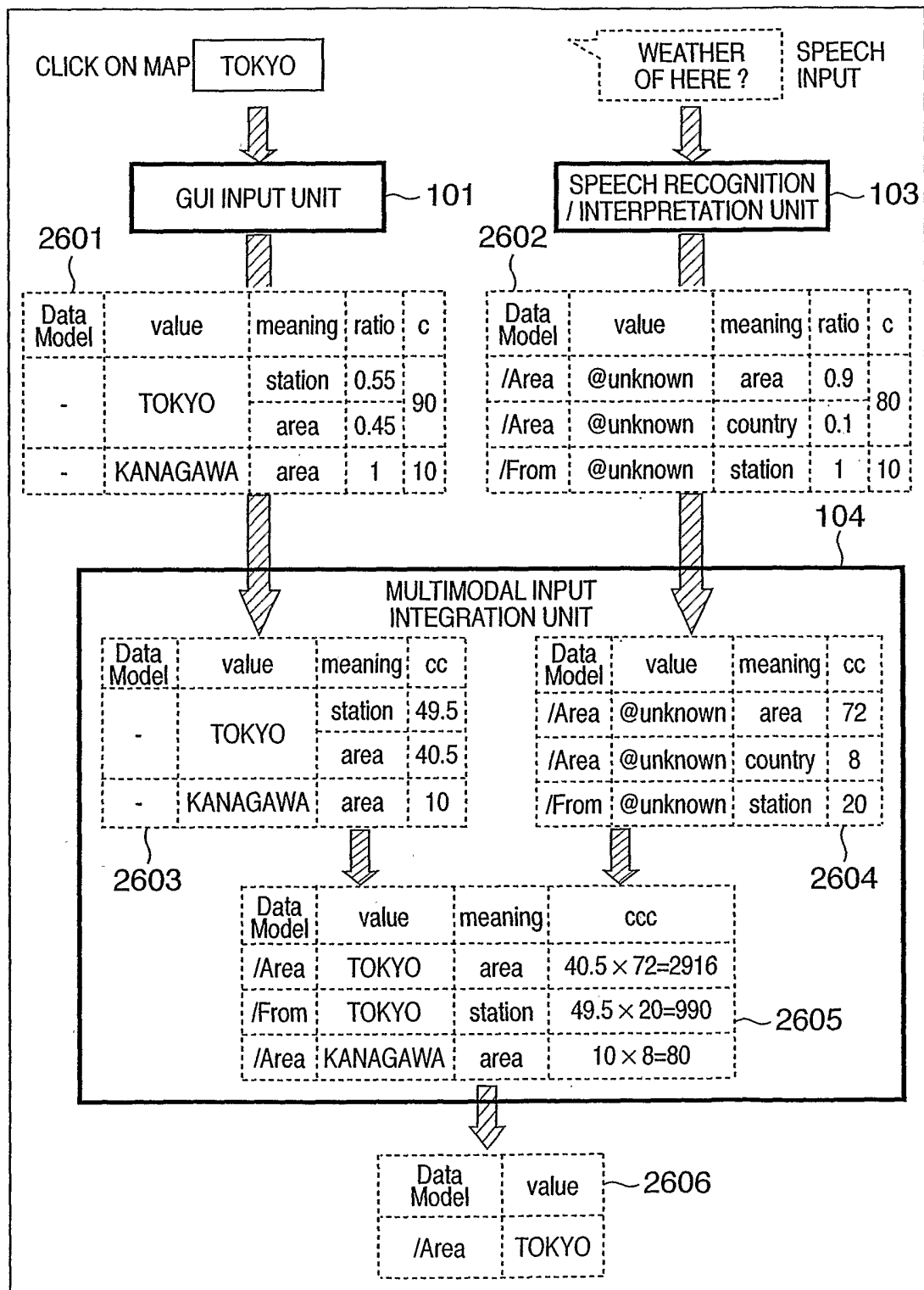
```

<?xml version="1.0" encoding="Shift_JIS"?>
<!DOCTYPE grammar SYSTEM "http://www.w3.org/TR/speech-grammar/grammar.dtd">
<grammar xmlns="http://www.w3.org/2001/06/grammar" xml:lang="ja" version="1.0"
mode="voice" root="weather">
  <rule id="weather">
    <ruleref uri="#area"/>
  </tag>
  $.res=$area;
</tag>
<item repeat="0-1"> weather of </item>
</rule>
<rule id="area">
  <one-of>
    <item> here <tag>$=[ "@unknown : area(90)", "@unknown : country(10)" ]</tag> </item>
    <item> tokyo <tag>$="TOKYO : area" </tag> </item>
    <item> kanagawa <tag>$="KANAGAWA : area" </tag> </item>
    <item> saitama <tag>$="SAITAMA : area" </tag> </item>
    <item> chiba <tag>$="CHIBA : area" </tag> </item>
    ...
  </one-of>
</rule>
</grammar>

```

28/28

FIG. 27



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP 2004/007905

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl⁷ G06F3/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl⁷ G06F3/00,3/16, G10L11/00-11/06,15/00-15/28

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
 Japanese Utility Model Gazette 1922-1996, Japanese Publication of Unexamined Utility Model Applications 1971-2004, Japanese Registered Utility Model Gazette 1994-2004, Japanese Gazette Containing the Utility Model 1996-2004

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 9-114634 A (ATR ONSEI HONYAKU TSUSHIN KENKYUSHO) 1997.05.02, Paragraphs [0047] - [0049] (Family:none)	1-16,19,20
A	US 5781179 A (NIPPON TELEGRAPH AND TELEPHONE CORP.) 1998.07.14, line 1, column 7 to line 9, column 9 & JP 9-81364 A	1-16,19,20
A	Yasuharu NAMBA, Semantic Analysis Using Fusionic Property of Multimodal Data, Transactions of Information Processing Society of Japan, July 1997, Vol.38, No.7, pages 1441 to 1453, ISSN 0387-5806.	1-16,19,20



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

31.08.2004

Date of mailing of the international search report

28.9.2004

Name and mailing address of the ISA/JP

Japan Patent Office

3-4-3, Kasumigaseki, Chiyoda-ku, Tokyo 100-8915, Japan

Authorized officer

MASAHARU KUBOTA

5E

4230

Telephone No. +81-3-3581-1101 Ext. 3520

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2004/007905

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Hideaki KIKUCHI, A Construction of Multimodal interface using Speech and Pen as Input Modalities, IPSJ SIG Notes, July 1995, Vol.95, No.73, pages 113 to 117, ISSN 0919-6072.	1-16,19,20
A	Kouichi KATSURADA, Comparison between VoiceXML and XISL, IPSJ SIG Notes, October 2001, Vol.2001, No.100, pages 49 to 54, ISSN 0919-6072.	1-16,19,20

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP 2004/007905

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☒ Claims Nos.: 17, 18
because they relate to subject matter not required to be searched by this Authority, namely:

The subject matter of claim 17, 18 relates to schemes, rules or methods of performing purely mental acts, which does not require an intentional search by the International Searching Authority in accordance with PCT Article 17(2)(a)(i) and Rule 39.1(iii).
2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:
4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest.
☐ No protest accompanied the payment of additional search fees.