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

(19) **World Intellectual Property Organization**
International Bureau

(43) **International Publication Date**
21 October 2010 (21.10.2010)

PCT

(10) **International Publication Number**
WO 2010/119794 A1

(51) **International Patent Classification**: *G06F 17/30* (2006.01)

(21) **International Application Number**: PCT/JP2010/056277

(22) **International Filing Date**: 31 March 2010 (31.03.2010)

(25) **Filing Language**: English

(26) **Publication Language**: English

(30) **Priority Data**:
2009-097389 13 April 2009 (13.04.2009) JP

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

(72) **Inventor; and**
(75) **Inventor/Applicant** *(for US only)*: **TAMIYA, Keisuke** [JP/JP]; c/o CANON KABUSHIKI KAISHA, 30-2, Shimomaruko 3-chome, Ohta-ku, Tokyo, 1468501 (JP).

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

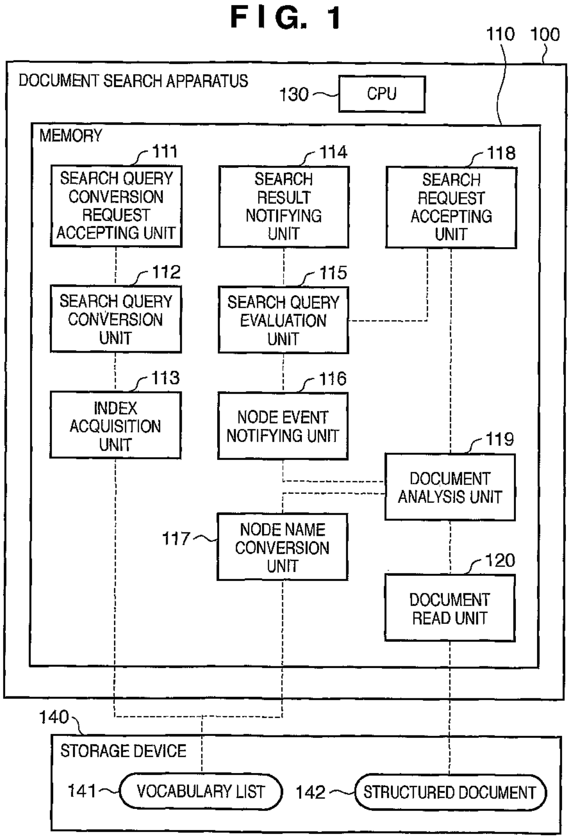
(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, KE, KG,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

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, 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, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM,

[Continued on next page]

(54) **Title**: INFORMATION PROCESSING APPARATUS AND INFORMATION PROCESSING METHOD

FIG. 1


110
100
DOCUMENT SEARCH APPARATUS
130
CPU
MEMORY
111
SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
114
SEARCH RESULT NOTIFYING UNIT
118
SEARCH REQUEST ACCEPTING UNIT
112
SEARCH QUERY CONVERSION UNIT
115
SEARCH QUERY EVALUATION UNIT
113
INDEX ACQUISITION UNIT
116
NODE EVENT NOTIFYING UNIT
119
DOCUMENT ANALYSIS UNIT
117
NODE NAME CONVERSION UNIT
120
DOCUMENT READ UNIT
140
STORAGE DEVICE
141
VOCABULARY LIST
142
STRUCTURED DOCUMENT

(57) **Abstract**: This invention is directed at providing a technique for implementing higher-speed search processing for a binary structured document. A search query conversion means converts a search query for a structured document by converting each node building the search query into a corresponding index by using a vocabulary list. A document analysis means specifies an index corresponding to each node building the structured document by using the vocabulary list. A search query evaluation means searches for part of the structured document that corresponds to the converted search query, by using each index described in the converted search query and the index corresponding to each node that is specified by the document analysis means.

TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

DESCRIPTION

TITLE OF INVENTION

INFORMATION PROCESSING APPARATUS AND INFORMATION PROCESSING METHOD

TECHNICAL FIELD

[0001] The present invention relates to a search technique for a structured document described in a binary format.

BACKGROUND ART

[0002] An XML language, specifications of which are formulated by the W3C standards body, is a language which describes a structured document. The XML language can describe a structured document using components (nodes) such as elements, attributes, and namespaces.

[0003] Although a document described in the XML language has a text format, there is a so-called binary XML technique which expresses the same document in a binary format. Typical formats are the Fast Infoset (ITU-T X.891) format standardized by the ITU-T (ITU-T Rec. X.891 | ISO/IEC 24824-1 (Fast Infoset)), and the Efficient XML Interchange format whose specifications are under development by the W3C. According to these binary XML techniques, a text document described in the

XML language can be expressed in a smaller size using a vocabulary table and node data information.

[0004] On the other hand, an XML Path Language (XPath) whose specifications are formulated by the W3C is proposed as a technique of designating, searching for, and extracting a specific part of an XML document (XML Path Language (XPath) Version 1.0 W3C Recommendation 16 November 1999). According to the XPath specifications, an XML document is regarded as a tree structure made up of nodes such as elements, attributes, and texts. A search query is described as a character string called a location step.

[0005] The location step is formed from an axis and node test which designate a node, and a predicate which designates a narrow-down condition using a node value or the like. The predicate can designate a character string comparison condition such as "character string data of a text node matches a specific character string." A technique of quickly comparing character strings in the predicate description has already been proposed (Japanese Patent Laid-Open No. 2007-249773).

[0006] A program using part of a binary XML structured document can extract the part by designating a search query described in XPath in a program such as an XML parser which analyzes an XML document, similar to a text XML structured document. In the search query described in XPath, the names of nodes such as elements and attributes are described in a text format. The program which analyzes an XML document checks if a condition for the binary XML format as well as the text XML format is met by comparing the name of a node obtained as a result of analysis with that of a node in the search query.

[0007] Processing of searching for a binary XML structured document using a search query described in XPath requires many character string comparison processes, increasing the calculation cost. In general, one purpose of the program using the binary XML format is to quickly perform analysis processing.

SUMMARY OF INVENTION

[0008] The present invention has been made to solve the above problems, and provides a technique for implementing higher-speed search processing for a binary structured document.

[0009] According to the first aspect of the present invention, an information processing apparatus characterized by comprising:

means for holding a table in which each node usable in a structured document and an index unique to the node are registered;

means for acquiring a search target structured document described in a binary format;

acquisition means for acquiring a search query for the search target structured document;

conversion means for converting the search query by converting each node building the search query into a corresponding index by using the table;

specifying means for specifying an index corresponding to each node building the search target structured document by using the table;

search means for searching for part of the search target structured document that corresponds to the search query converted by said conversion means, by using each index described in the search query converted by said conversion means and the index corresponding to each node in the search target structured document that is specified by said specifying means; and

means for outputting a result of the search by said search means.

[0010] According to the second aspect of the present invention, an information processing method characterized by comprising:

a step of acquiring a search target structured document described in a binary format;

an acquisition step of acquiring a search query for the search target structured document;

a conversion step of converting the search query by converting each node building the search query into a corresponding index by using a table in which each node usable in a structured document and an index unique to the node are registered;

a specifying step of specifying an index corresponding to each node building the search target structured document by using the table;

a search step of searching for part of the search target structured document that corresponds to the search query converted in the conversion step, by using each index described in the search query converted in the conversion step and the index corresponding to each node in the search target structured document that is specified in the specifying step; and

a step of outputting a result of the search in the search step.

[0011] The arrangement of the present invention can implement higher-speed search processing for a binary structured document.

[0012] Further features of the present invention will become apparent from the following description of exemplary embodiments with reference to the attached drawings.

BRIEF DESCRIPTION OF DRAWINGS

[0013] Fig. 1 is a block diagram exemplifying the hardware configuration of a document search apparatus serving as an information processing apparatus according to the first embodiment of the present invention;

[0014] Fig. 2 is a view exemplifying the structure of a structured document which describes a binary XML structured document 142 in a text XML format;

[0015] Fig. 3 is a table exemplifying the structure of a vocabulary list 141;

[0016] Fig. 4 is a view exemplifying the structure of the structured document 142 obtained by converting the text XML structured document shown in Fig. 2 into the Fast Infoset format serving as an example of the binary XML format using the vocabulary list 141;

[0017] Fig. 5 is a view exemplifying the structure of the structured document 142 obtained by converting the text XML structured document shown in Fig. 2 into the Fast Infoset format serving as an example of the binary XML format using the vocabulary list 141;

[0018] Figs. 6A to 6D are views showing search queries described in the W3C XPath language, and results of converting the search queries using indices;

[0019] Fig. 7 is a flowchart of search processing for the structured document 142 by a document search apparatus 100;

[0020] Fig. 8A and 8B are flowcharts each showing details of processing in step S707;

[0021] Fig. 9 is a block diagram exemplifying the hardware configuration of a document search apparatus 900 serving as an information processing apparatus according to the second embodiment of the present invention; and

[0022] Fig. 10 is a flowchart of search processing for the structured document 142 by the document search apparatus 900.

DESCRIPTION OF EMBODIMENTS

[0023] Embodiments of the present invention will now be described with reference to the accompanying drawings. It should be noted that the following embodiments are merely examples of specifically practicing the present invention, and are concrete examples of the arrangement defined by the scope of the appended claims.

[0024] [First Embodiment]

Fig. 1 is a block diagram exemplifying the hardware configuration of a document search apparatus serving as an information processing apparatus according to the first embodiment. Fig. 1 shows the main arrangement in the following description, and the arrangement of an apparatus capable of implementing a technique to be described in the embodiment is not limited to that shown in Fig. 1.

[0025] As shown in Fig. 1, a document search apparatus 100 includes a CPU 130 and memory 110. The document search apparatus 100 is connected to a storage device 140 via a cable. The document search apparatus 100 can read out and write data from and in the storage device 140 via the cable.

[0026] The storage device 140 is a large-capacity information storage device typified by a hard disk drive. The storage device 140 stores a binary structured document 142 to be searched (search target structured document), and a vocabulary list 141 which holds the name and index of each node appearing in the structured document 142 (search target structured document).

[0027] More specifically, the structured document 142 is a structured document in the binary XML format defined in the ISO Fast Infoset and W3C Efficient XML Interchange specifications. Nodes are document units such as elements and attributes which form the structured document 142. A node name registrable in the vocabulary list 141 is the name of a node used in the structured document 142. In addition, the name and index of a node generally usable in a structured document may be registered.

[0028] Fig. 3 is a table exemplifying the structure of the vocabulary list 141. The name of each node appearing in the structured document 142 is registered in a column 302. An index unique to each node (unique in the structured document 142) is registered in a column 301. More specifically, a set (entry) of the name of a node and an index unique to the node is registered in the vocabulary list 141 for each node.

[0029] Fig. 2 is a view exemplifying the structure of a structured document which describes the binary XML structured document 142 in a text XML format. Figs. 4 and 5 are views exemplifying the structure of the structured document 142 obtained by converting the text XML structured document shown in Fig. 2 into the Fast Infoset format serving as an example of the binary XML format using the vocabulary list 141.

[0030] According to the Fast Infoset format, a structured document is represented by binary symbols indicating the start and end of each node, and a binary string indicating the value of each node. In Figs. 4 and 5, these binary representations are described as

[node start symbol (parameter)] node value
[node end symbol]

[0031] In the Fast Infoset, the name of a node can be replaced with an index using the vocabulary list 141. Instead of the index, the node name can also be directly described. Fig. 4 exemplifies the structure of a structured document in which node names are completely replaced with indices. Fig. 5 exemplifies the structure of a structured document in which some node names remain unreplaced.

[0032] The structured document 142 and vocabulary list 141 stored in the storage device 140 are loaded into the memory 110 under the control of the CPU 130, as needed, and processed by the CPU 130.

[0033] The memory 110 is a readable/writable memory typified by the RAM, and stores units to be described below in the form of computer programs. The units, which are stored in the memory 110 in the following description, may be stored in the storage device 140. Even in this case, these units are loaded into the memory 110 in operation under the control of the CPU 130.

[0034] A search query conversion request accepting unit 111 acquires a search query for the structured document 142 via an application program or the like. As a consequence, the search query conversion request accepting unit 111 acquires a request (conversion request) to convert the search query.

[0035] An index acquisition unit 113 acquires an index registered in the vocabulary list 141 and supplies it to a search query conversion unit 112. When the search query conversion request accepting unit 111 acquires a search query, the search query conversion unit 112 converts it using the index supplied from the index acquisition unit 113.

[0036] A search request accepting unit 118 acquires a search query for the structured document 142 via an application program or the like, thereby acquiring a search request. The search query is one converted by the search query conversion unit 112.

[0037] A document read unit 120 reads out the structured document 142. A document analysis unit 119 analyzes the structured document 142 read out by the document read unit 120, and specifies each node described in the structured document 142.

[0038] When the document analysis unit 119 detects a node whose name has not been replaced with an index in the structured document 142 as a result of analyzing the structured document 142, a node name conversion unit 117 converts the name into a corresponding index by referring to the vocabulary list 141.

[0039] A node event notifying unit 116 notifies a search query evaluation unit 115 of the result of analysis by the document analysis unit 119 as an event. The search query evaluation unit 115 evaluates the search query acquired by the search request accepting unit 118, based on the event received from the node event notifying unit 116. A search result notifying unit 114 outputs (notifies) the result of evaluation by the search query evaluation unit 115.

[0040] In addition to these units, information to be described is registered as known information in the memory 110. Also, the memory 110 has a work memory used when the CPU 130 executes various processes. That is, the memory 110 can properly provide a variety of areas.

[0041] Search processing for the structured document 142 by the document search apparatus 100 will be explained with reference to Fig. 7 which is a flowchart of this processing. For the descriptive convenience, the foregoing units stored in the memory 110 serve as main processors. However, these units are stored in the memory 110 in the form of computer programs, as described above, and the CPU 130 executes these computer programs. In practice, therefore, the CPU 130 is a main processor.

[0042] In step S701, the search query conversion request accepting unit 111 acquires a search request by acquiring a search query and the name of a vocabulary list (the file name of the vocabulary list 141 in the embodiment) from an application program or the like. The acquisition form of the search query and the file name of the vocabulary list 141 is not particularly limited. In step S702, the search query conversion request accepting unit 111 sends the acquired file name of the vocabulary list 141 and the acquired search query to the subsequent search query conversion unit 112.

[0043] In step S703, the search query conversion unit 112 extracts the name of each node described in the search query received from the search query conversion request accepting unit 111 in step S702. The search query conversion unit 112 sends the extracted node name to the subsequent index acquisition unit 113 together with the file name of the vocabulary list 141 that has also been received from the search query conversion request accepting unit 111 in step S702.

[0044] In step S704, the index acquisition unit 113 specifies the vocabulary list 141 in the storage device 140 using the name of the vocabulary list 141 that has been received from the search query conversion unit 112. By referring to the specified vocabulary list 141, the index acquisition unit 113 acquires, from the vocabulary list 141, an index corresponding to each node name received from the search query conversion unit 112. The index acquisition unit 113 sends back the acquired "index corresponding to each node name" to the search query conversion unit 112.

[0045] In step S705, the search query conversion unit 112 converts the search query received from the search query conversion request accepting unit 111 by using each index received from the index acquisition unit 113. The conversion of the search query using the index will be explained.

[0046] Figs. 6A to 6D are views showing search queries described in the W3C XPath language, and results of converting the search queries using indices.

Fig. 6A shows a search query "/booklist/book/title".

[0047] When the search query conversion request accepting unit 111 acquires this search query and sends it to the subsequent search query conversion unit 112, the search query conversion unit 112 first segments the search query described in the W3C XPath language into search units called location steps. In Fig. 6A, the search query is segmented into three location steps "booklist", "book", and "title". The location step is formed from an axis indicating the search direction of a node in a structured document, a node test designating the type of node, and a predicate serving as a selection condition for narrowing down.

[0048] The search query conversion unit 112 operates as follows when it refers to the vocabulary list 141 exemplified in Fig. 3. More specifically, the search query conversion unit 112 acquires, from the vocabulary list 141 for the respective location steps, indices (EII) corresponding to character strings (booklist, book, title) which are node test values. Then, the search query conversion unit 112 generates information in the form of a table exemplified in Fig. 6B as a converted search query using the acquired indices for the respective location steps.

[0049] In Fig. 6B, a number (location step number) unique to each location step is registered in a column 601. The location step number indicates the search order. The axis of each location step is registered in a column 602. The node test value of each location step is registered in a column 603. The predicate of each location step is registered in a column 604.

[0050] Fig. 6C shows a search query "//book/price[number()>2000]". When the search query conversion request accepting unit 111 acquires this search query and sends it to the subsequent search query conversion unit 112, the search query conversion unit 112 first segments the search query described in the W3C XPath language into search units called location steps. In Fig. 6C, the search query is segmented into two location steps "book" and "price".

[0051] The search query conversion unit 112 operates as follows when it refers to the vocabulary list 141 exemplified in Fig. 3. More specifically, the search query conversion unit 112 acquires, from the vocabulary list 141 for the respective location steps, indices (EII) corresponding to character strings (book, price) which are node test values. Then, the search query conversion unit 112 generates information in the form of a table exemplified in Fig. 6D as a converted search query using the acquired indices for the respective location steps.

[0052] In Fig. 6D, the location step number of each location step is registered in a column 611. The axis of each location step is registered in a column

612. The node test value of each location step is registered in a column 613. The predicate of each location step is registered in a column 614.

[0053] In Figs. 6A to 6D, only the element name of an element node is targeted as a character string to be converted. However, the Fast Infoset format allows managing even character strings such as an attribute name, namespace URI, and namespace prefix in the vocabulary list. The same conversion can be executed even when a location step in a search query has a description regarding an attribute node or namespace node other than an element node. The search query conversion unit 112 sends the converted search query to the search query conversion request accepting unit 111.

[0054] Referring back to Fig. 7, in step S706, the search query conversion request accepting unit 111 outputs the converted search query received from the search query conversion unit 112. Although the output destination is not particularly limited, the user inputs the search query into the apparatus for search. Thus, the search query can be held in the storage device 140 or memory 110 so that the user can handle it.

[0055] In step S707, processing to search for a target part of the structured document 142 using the converted search query is performed. Fig. 8A and 8B are flowcharts each showing details of the processing in step S707.

[0056] First, the user of the apparatus inputs, with a keyboard and mouse (neither is shown) to the apparatus, a search query, the file name of a structured document to be searched using the search query, and the file name of a vocabulary list.

[0057] Then, in step S801, the search request accepting unit 118 acquires the input pieces of information. In the embodiment, the input search query is a search query converted in the processes of steps S701 to S706. The input file name of the structured document is assumed to be that of the structured document 142. The input file name of the vocabulary list is assumed to be that of the vocabulary list 141.

[0058] In step S802, the search request accepting unit 118 sends the input search query to the search query evaluation unit 115. In step S803, the search request accepting unit 118 sends the input file names of the vocabulary list 141 and structured document 142 to the document analysis unit 119. Processes in steps S804 to S817 are performed for each building part of the structured document 142.

[0059] In step S805, the document analysis unit 119 sends, to the document read unit 120, the file name of the structured document 142 that has been received from the search request accepting unit 118. The document read unit 120 reads out the next part of the structured document 142 specified by the file name.

When the processing in this step is executed for the first time, the document read unit 120 reads out the first part of the structured document 142. The "next part" means an unread part of the structured document that can be stored in a document read buffer area by the document read unit 120.

[0060] If there is no part to be read out in this step, the process ends via step S806. If the next part has been read out successfully, the process advances to step S807 via step S806.

[0061] In step S807, the document analysis unit 119 analyzes the part read out by the document read unit 120 and extracts the next node. In step S808, the document analysis unit 119 refers to the extracted node and determines whether the node has been converted into an index. When the node has been converted into an index, the index is described in an element start symbol (EII) in Figs. 4 and 5 in the Fast Infoset format. Thus, it suffices to determine in step S808 whether an index is described in EII.

[0062] If the document analysis unit 119 determines that the node has been converted into an index, the process advances to step S809; if NO, to step S813.

[0063] In step S813, the document analysis unit 119 sends, to the node name conversion unit 117, the file name of the vocabulary list 141 that has been received from the search request accepting unit 118 and the node name extracted in step S807.

[0064] In step S814, the node name conversion unit 117 specifies an index corresponding to the node name received from the document analysis unit 119 by referring to the vocabulary list 141 specified by the file name similarly received from the document analysis unit 119. The node name conversion unit 117 sends the specified index to the document analysis unit 119.

[0065] In step S809, the document analysis unit 119 sends node information of the node extracted in step S807 and the index of the node to the node event notifying unit 116. The node information includes the namespace definition of an element, the contents of character string data defined as element contents, a parent element, and an attribute value. The node event notifying unit 116 sends the information received from the document analysis unit 119 as an event to the search query evaluation unit 115.

[0066] In step S810, the search query evaluation unit 115 performs search processing by comparing the search query received from the search request accepting unit 118 in step S802 with the index received from the document analysis unit 119 via the node event notifying unit 116. For example, the search query evaluation unit 115 receives the search query shown in Fig. 6A in step S802, and receives indices "1", "2", and "3" in this order in step S809. In this case, the search query evaluation unit 115 determines that a node corresponding to this index is hit as a search target (satisfies a condition described in the search query).

[0067] If the search query evaluation unit 115 determines as a result of the comparison in step S810 that the condition described in the search query is satisfied, the process advances to step S815 via step S811. If the search query evaluation unit 115 determines that the condition described in the search query is not satisfied, the process advances to step S817 via step S811, and the subsequent processing is done for the next part.

[0068] In step S815, the search query evaluation unit 115 sends node information of the node hit in the search to the search result notifying unit 114. In step S816, the search result notifying unit 114 generates a search result notification event from the node information received from the search query evaluation unit 115, and outputs the generated search result notification event. The output destination is not particularly limited. For example, the search result notification event may be sent to an application program which displays the node information on the display device (not shown) of the document search apparatus 100.

[0069] When the search query is described in XPath, as shown in Figs. 6A and 6C, the search result takes one data type among a node set, true/false (Boolean) value, numerical value, and character string. The form of the search result notification event complies with a preliminary agreement between the user of the apparatus and the search result notifying unit 114. For example, for a program described in the C language, the search query evaluation unit 115 invokes a function defined by the user of the apparatus and transfers it as the data type return value of the search result.

[0070] [Second Embodiment]

In the first embodiment, the vocabulary list 141 is generated in advance and held in the storage device 140. However, according to the Fast Infoset format and the like, the structured document 142 can be analyzed while dynamically generating a vocabulary list without referring to a vocabulary list generated in advance from a schema definition or the like.

[0071] In the second embodiment, an arrangement for generating a vocabulary list 141 is added to the document search apparatus 100 according to the first embodiment. Fig. 9 is a block diagram exemplifying the hardware configuration of a document search apparatus 900 serving as an information processing apparatus according to the second embodiment. As shown in Fig. 9, the document search apparatus 900 includes a vocabulary list generation unit 914 for generating the vocabulary list 141, in addition to the arrangement shown in Fig. 1. In Fig. 9, the reference numerals as those in Fig. 1 denote the same parts, and a description thereof will not be repeated.

[0072] Fig. 10 is a flowchart of search processing for a structured document 142 by the document search apparatus 900. In step S1001, a search query conversion request accepting unit 111 acquires a search request by acquiring a search query and the file name of the structured document 142 from an application program or the like. The acquisition form of the search query and the file name of the structured document 142 is not particularly limited. In step S1002, the search query conversion request accepting unit 111 sends the acquired file name of the structured document 142 to the subsequent vocabulary list generation unit 914.

[0073] In step S1003, the vocabulary list generation unit 914 sends the file name received from the search query conversion request accepting unit 111 to a document read unit 120. The document read unit 120 reads out the structured document 142 specified by the file name. The document read unit 120 sends the readout structured document 142 to the vocabulary list generation unit 914.

[0074] In step S1004, the vocabulary list generation unit 914 analyzes the structured document

142, acquiring the node definitions of an element node, attribute node, namespace node, and the like. In step S1005, the vocabulary list 141 registers, in the vocabulary list 141, the node names of the element node and attribute node, and the namespace URI and namespace prefix of the namespace node.

[0075] In step S1006, the vocabulary list generation unit 914 issues the file name of the vocabulary list 141 generated in step S1005, and sends the issued file name to the search query conversion request accepting unit 111. Step S1007 and subsequent steps are the same as step S702 and subsequent steps in Fig. 7, and a description thereof will not be repeated.

[0076] According to the above-described embodiments, the number of character string comparison processes can be decreased when a specific part of a structured document compressed by a binary XML technique or the like is searched for using a search query. The specific part of the compressed structured document can therefore be searched for and extracted more quickly. This effect is significant especially when many node names such as an element name and attribute name are described in a search query and when the size of a search target document is large.

[0077]Other Embodiments

Aspects of the present invention can also be realized by a computer of a system or apparatus (or devices such as a CPU or MPU) that reads out and executes a program recorded on a memory device to perform the functions of the above-described embodiment(s), and by a method, the steps of which are performed by a computer of a system or apparatus by, for example, reading out and executing a program recorded on a memory device to perform the functions of the above-described embodiment(s). For this purpose, the program is provided to the computer for example via a network or from a recording medium of various types serving as the memory device (e.g., computer-readable medium).

[0078] While the present invention has been described with reference to exemplary embodiments, it is to be understood that the invention is not limited to the disclosed exemplary embodiments. The scope of the following claims is to be accorded the broadest interpretation so as to encompass all such modifications and equivalent structures and functions.

[0079] This application claims the benefit of Japanese Patent Application No. 2009-097389, filed April 13, 2009, which is hereby incorporated by reference herein in its entirety.

CLAIMS

1. An information processing apparatus characterized by comprising:

means for holding a table in which each node usable in a structured document and an index unique to the node are registered;

means for acquiring a search target structured document described in a binary format;

acquisition means for acquiring a search query for the search target structured document;

conversion means for converting the search query by converting each node building the search query into a corresponding index by using the table;

specifying means for specifying an index corresponding to each node building the search target structured document by using the table;

search means for searching for part of the search target structured document that corresponds to the search query converted by said conversion means, by using each index described in the search query converted by said conversion means and the index corresponding to each node in the search target structured document that is specified by said specifying means; and

means for outputting a result of the search by said search means.

2. The apparatus according to claim 1, characterized in that the search target structured document is a structured document in a binary XML format defined by ISO Fast Infoset and W3C Efficient XML Interchange specifications.

3. The apparatus according to claim 1 or 2, characterized in that

the search query is described in a W3C XPath language, and

said conversion means segments the search query acquired by said acquisition means into location steps, acquires indices corresponding to the respective location steps from the table, and obtains, as the converted search query, a table in which a set of each location step and its corresponding index is registered.

4. The apparatus according to any one of claims 1 to 3, characterized by further comprising generation means for generating the table after acquiring the search target structured document.

5. An information processing method characterized by comprising:

a step of acquiring a search target structured document described in a binary format;

an acquisition step of acquiring a search query for the search target structured document;

a conversion step of converting the search query by converting each node building the search query into a corresponding index by using a table in which each node usable in a structured document and an index unique to the node are registered;

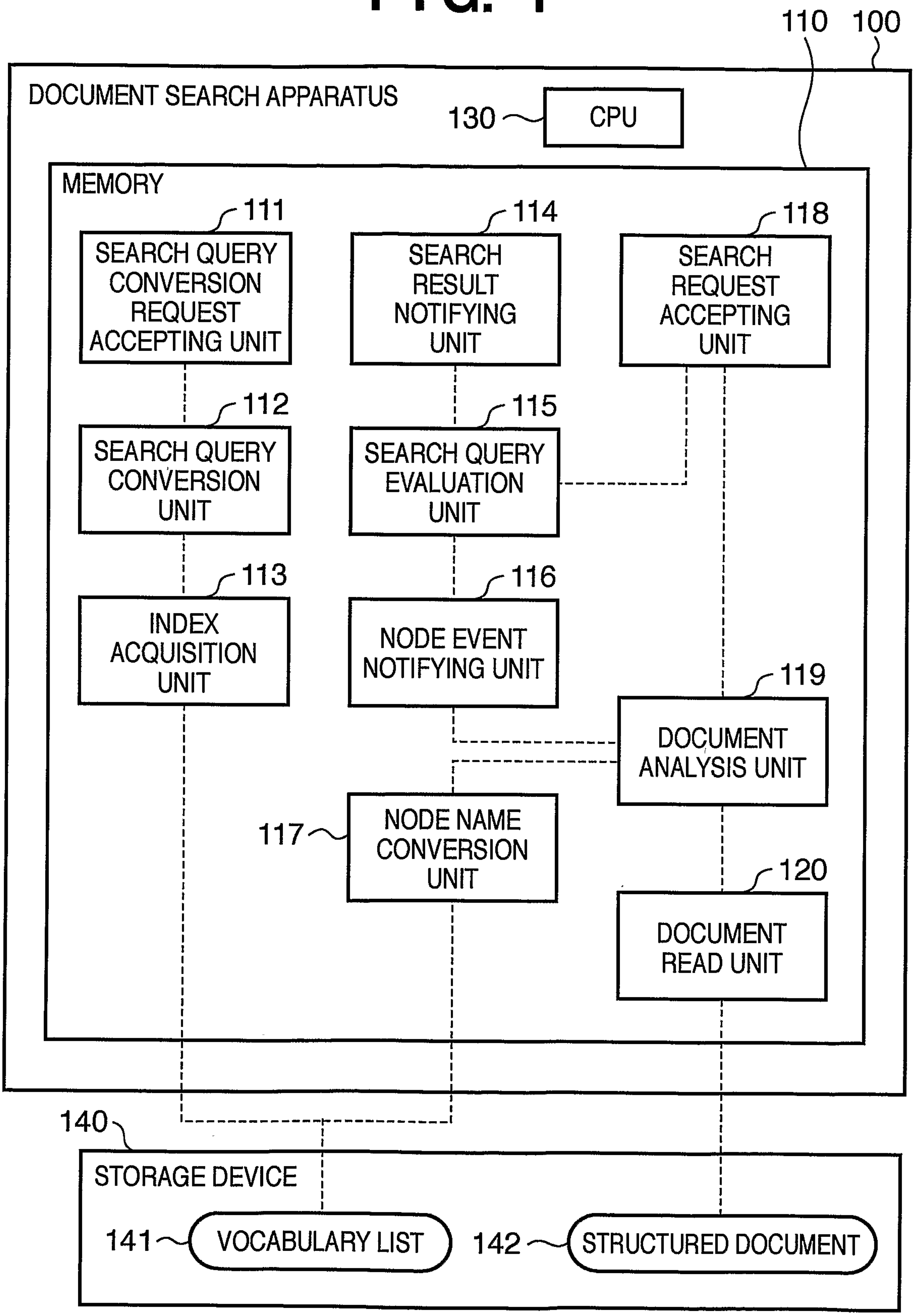
a specifying step of specifying an index corresponding to each node building the search target structured document by using the table;

a search step of searching for part of the search target structured document that corresponds to the search query converted in the conversion step, by using each index described in the search query converted in the conversion step and the index corresponding to each node in the search target structured document that is specified in the specifying step; and

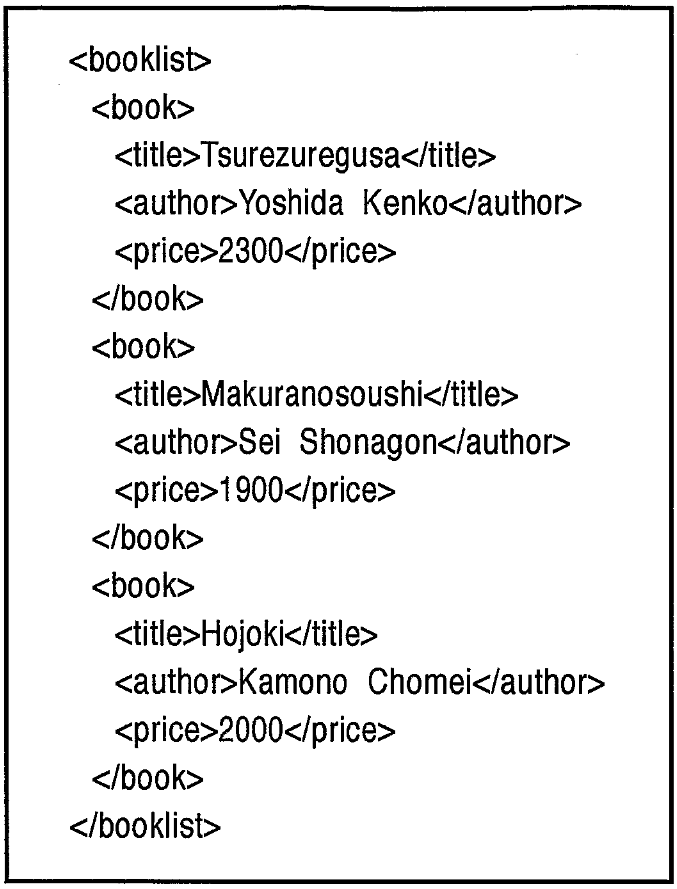
a step of outputting a result of the search in the search step.

6. A computer-readable storage medium storing a computer program for causing a computer to function as each means of an information processing apparatus defined in any one of claims 1 to 4.

FIG. 1

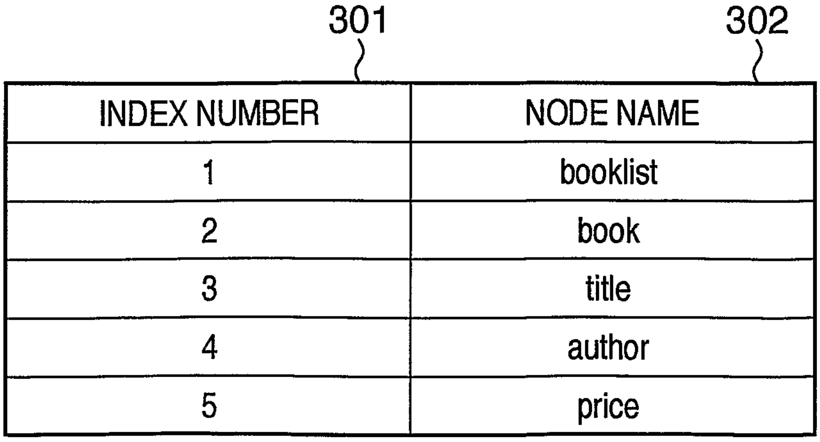

110
100
DOCUMENT SEARCH APPARATUS
130
CPU
MEMORY
111
SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
114
SEARCH RESULT NOTIFYING UNIT
118
SEARCH REQUEST ACCEPTING UNIT
112
SEARCH QUERY CONVERSION UNIT
115
SEARCH QUERY EVALUATION UNIT
113
INDEX ACQUISITION UNIT
116
NODE EVENT NOTIFYING UNIT
119
DOCUMENT ANALYSIS UNIT
117
NODE NAME CONVERSION UNIT
120
DOCUMENT READ UNIT
140
STORAGE DEVICE
141
VOCABULARY LIST
142
STRUCTURED DOCUMENT

F I G. 2



```
<booklist>
  <book>
    <title>Tsurezuregusa</title>
    <author>Yoshida Kenko</author>
    <price>2300</price>
  </book>
  <book>
    <title>Makuranosoushi</title>
    <author>Sei Shonagon</author>
    <price>1900</price>
  </book>
  <book>
    <title>Hojoki</title>
    <author>Kamono Chomei</author>
    <price>2000</price>
  </book>
</booklist>
```

F I G. 3



INDEX NUMBER	NODE NAME
1	booklist
2	book
3	title
4	author
5	price

FIG. 4

```
[EII(INDEX NUMBER=1)]
[EII(INDEX NUMBER=2)]
[EII(INDEX NUMBER=3)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=13]Tsurezuregusa
[TE]
[EII(INDEX NUMBER=4)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=13)]Yoshida Kenko
[TE]
[EII(INDEX NUMBER=5)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=4]2300
[TE]
[TE]
[EII(INDEX NUMBER=2)]
[EII(INDEX NUMBER=3)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=14]Makuranosoushi
[TE]
[EII(INDEX NUMBER=4)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=12)]Sei Shonagon
[TE]
[EII(INDEX NUMBER=5)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=4]1900
[TE]
[TE]
[EII(INDEX NUMBER=2)]
[EII(INDEX NUMBER=3)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=6)]Hojoki
[TE]
[EII(INDEX NUMBER=4)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=13)]Kamono Chomei
[TE]
[EII(INDEX NUMBER=5)]
[CII(CHARACTER ENCODING TYPE=UTF-8, CHARACTER STRING LENGTH=4]2000
[TE]
[TE]
[TE]
```

[EII] : ELEMENT START SYMBOL
[CII] : CHARACTER STRING DATA START SYMBOL
[TE] : ELEMENT END SYMBOL

F I G. 5

```
[EII(NO INDEX)]booklist
[EII(NO INDEX)]book
[EII(NO INDEX)]title
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH=13]Tsurezuregusa
[TE]
[EII(NO INDEX)]author
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH=13)]Yoshida Kenko
[TE]
[EII(NO INDEX)]price
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH=4]2300
[TE]
[TE]
[EII(INDEX NUMBER=2)]
[EII(INDEX NUMBER=3)]
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH =14]Makuranosoushi
[TE]
[EII(INDEX NUMBER=4)]
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH=12)]Sei Shonagon
[TE]
[EII(INDEX NUMBER=5)]
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH=4]1900
[TE]
[TE]
[EII(INDEX NUMBER=2)]
[EII(INDEX NUMBER=3)]
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH=6)]Hojoki
[TE]
[EII(INDEX NUMBER=4)]
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH=13)]Kamono Chomei
[TE]
[EII(INDEX NUMBER=5)]
[CII(CHARACTER ENCODING TYPE =UTF-8, CHARACTER STRING LENGTH=4]2000
[TE]
[TE]
[TE]
```

[EII] : ELEMENT START SYMBOL
[CII] : CHARACTER STRING DATA START SYMBOL
[TE] : ELEMENT END SYMBOL

FIG. 6A

/booklist/book/title

FIG. 6B

601 602 603 604

LOCATION STEP NUMBER	AXIS	NODE TEST	PREDICATE
1	child	EII(INDEX NUMBER =1)	NONE
2	child	EII(INDEX NUMBER =2)	NONE
3	child	EII(INDEX NUMBER =3)	NONE

FIG. 6C

//book/price[number() > 2000]

FIG. 6D

611 612 613 614

LOCATION STEP NUMBER	AXIS	NODE TEST	PREDICATE
1	descendant-or-self	ARBITRARY NODE	NONE
2	child	EII(INDEX NUMBER =2)	NONE
3	child	EII(INDEX NUMBER =5)	number() > 2000

FIG. 7

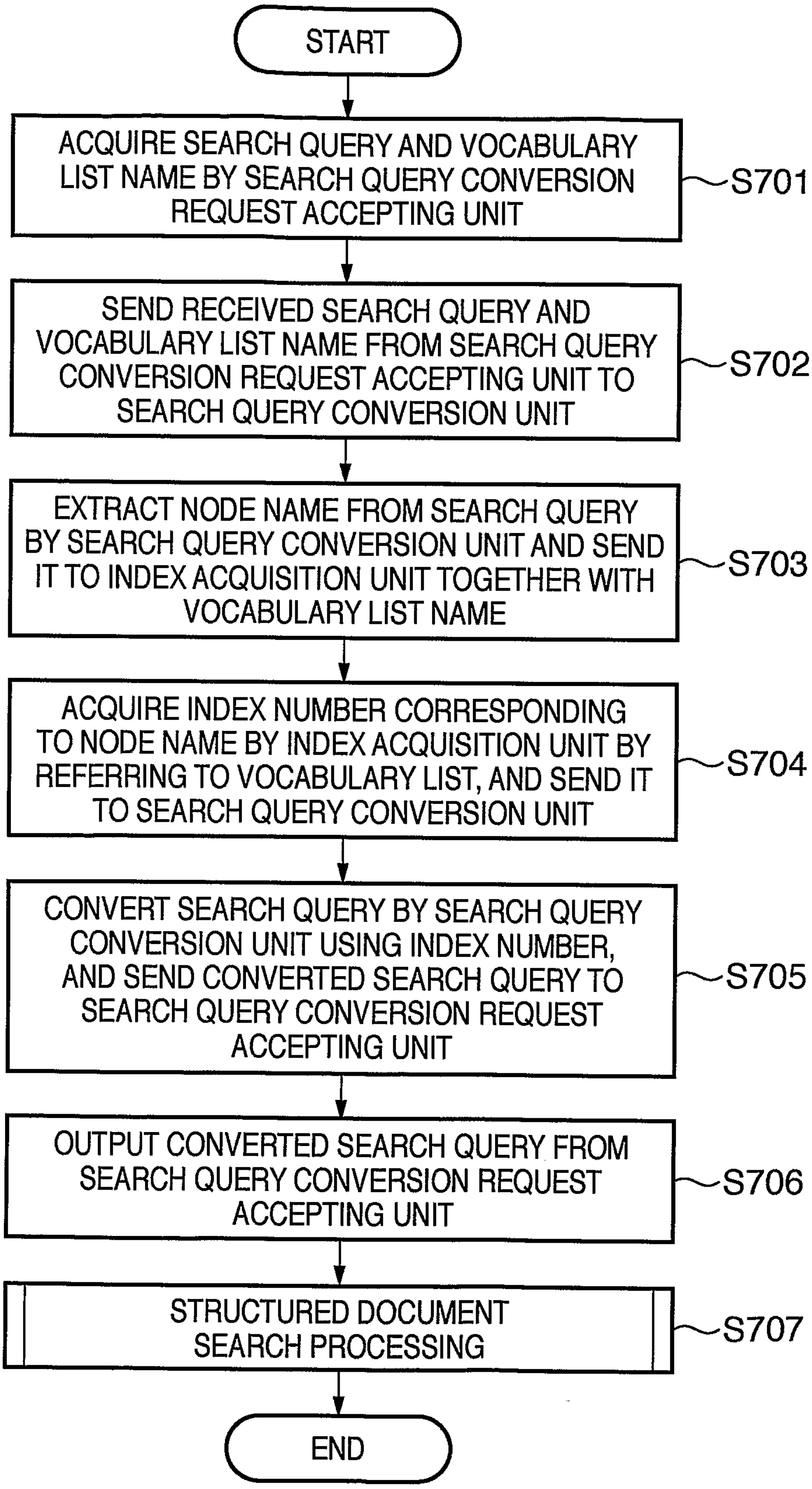

START
ACQUIRE SEARCH QUERY AND VOCABULARY LIST NAME BY SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
S701
SEND RECEIVED SEARCH QUERY AND VOCABULARY LIST NAME FROM SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT TO SEARCH QUERY CONVERSION UNIT
S702
EXTRACT NODE NAME FROM SEARCH QUERY BY SEARCH QUERY CONVERSION UNIT AND SEND IT TO INDEX ACQUISITION UNIT TOGETHER WITH VOCABULARY LIST NAME
S703
ACQUIRE INDEX NUMBER CORRESPONDING TO NODE NAME BY INDEX ACQUISITION UNIT BY REFERRING TO VOCABULARY LIST, AND SEND IT TO SEARCH QUERY CONVERSION UNIT
S704
CONVERT SEARCH QUERY BY SEARCH QUERY CONVERSION UNIT USING INDEX NUMBER, AND SEND CONVERTED SEARCH QUERY TO SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
S705
OUTPUT CONVERTED SEARCH QUERY FROM SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
S706
STRUCTURED DOCUMENT SEARCH PROCESSING
S707
END

FIG. 8A

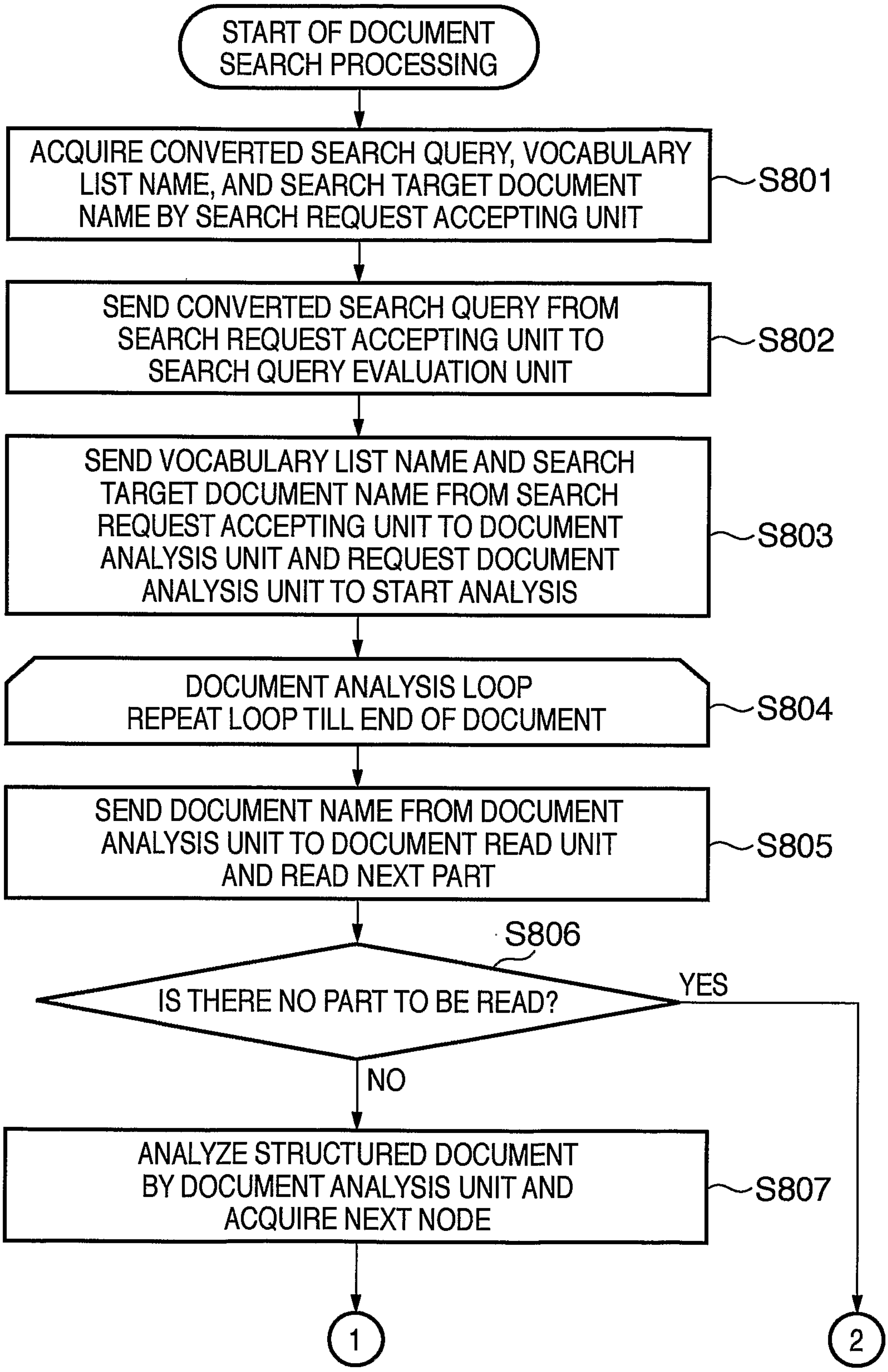
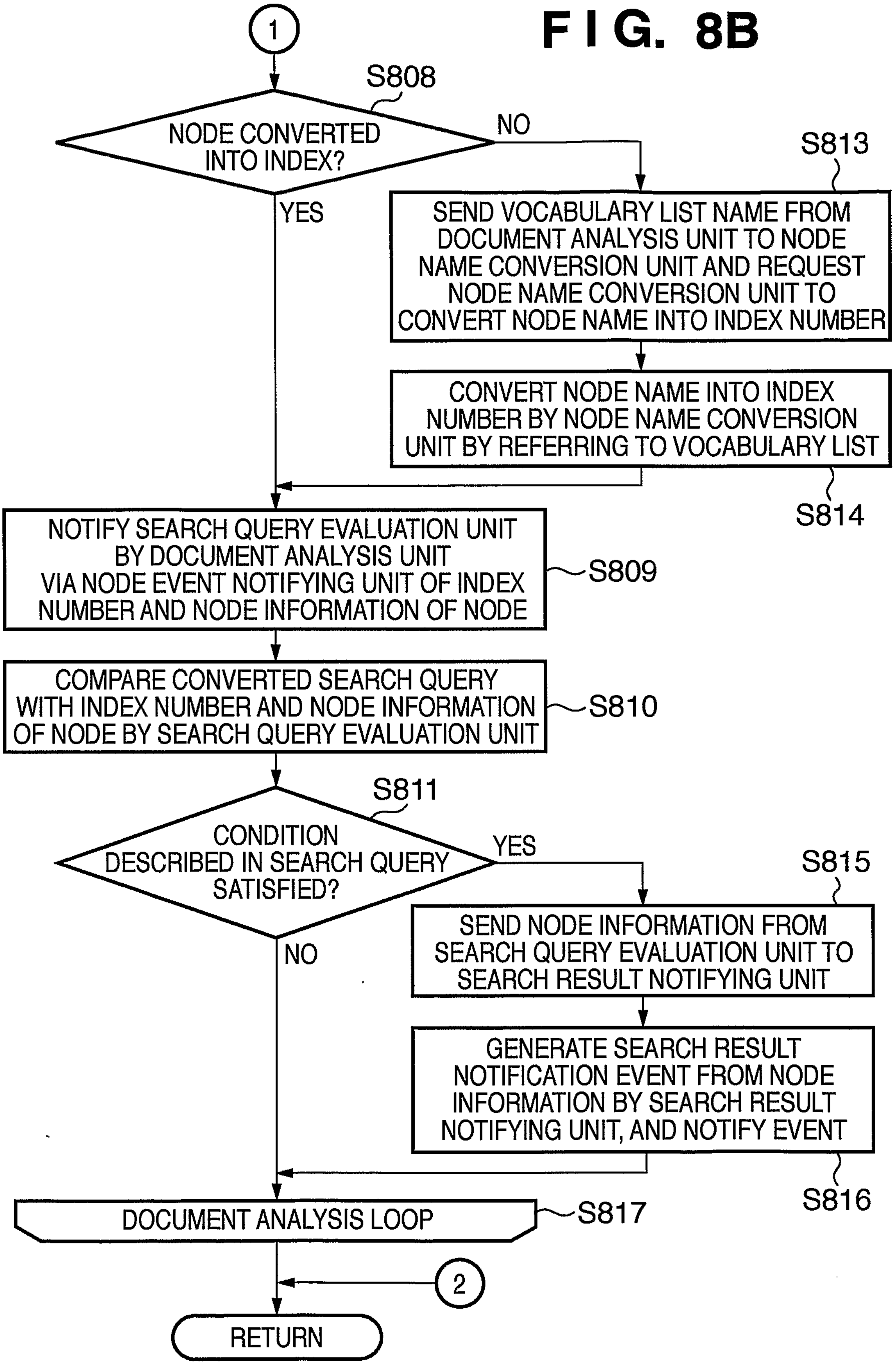

START OF DOCUMENT SEARCH PROCESSING
ACQUIRE CONVERTED SEARCH QUERY, VOCABULARY LIST NAME, AND SEARCH TARGET DOCUMENT NAME BY SEARCH REQUEST ACCEPTING UNIT
S801
SEND CONVERTED SEARCH QUERY FROM SEARCH REQUEST ACCEPTING UNIT TO SEARCH QUERY EVALUATION UNIT
S802
SEND VOCABULARY LIST NAME AND SEARCH TARGET DOCUMENT NAME FROM SEARCH REQUEST ACCEPTING UNIT TO DOCUMENT ANALYSIS UNIT AND REQUEST DOCUMENT ANALYSIS UNIT TO START ANALYSIS
S803
DOCUMENT ANALYSIS LOOP REPEAT LOOP TILL END OF DOCUMENT
S804
SEND DOCUMENT NAME FROM DOCUMENT ANALYSIS UNIT TO DOCUMENT READ UNIT AND READ NEXT PART
S805
S806
IS THERE NO PART TO BE READ?
YES
NO
ANALYZE STRUCTURED DOCUMENT BY DOCUMENT ANALYSIS UNIT AND ACQUIRE NEXT NODE
S807
1
2

FIG. 8B



1
S808
NODE CONVERTED INTO INDEX?
NO
YES
S813
SEND VOCABULARY LIST NAME FROM DOCUMENT ANALYSIS UNIT TO NODE NAME CONVERSION UNIT AND REQUEST NODE NAME CONVERSION UNIT TO CONVERT NODE NAME INTO INDEX NUMBER
CONVERT NODE NAME INTO INDEX NUMBER BY NODE NAME CONVERSION UNIT BY REFERRING TO VOCABULARY LIST
S814
NOTIFY SEARCH QUERY EVALUATION UNIT BY DOCUMENT ANALYSIS UNIT VIA NODE EVENT NOTIFYING UNIT OF INDEX NUMBER AND NODE INFORMATION OF NODE
S809
COMPARE CONVERTED SEARCH QUERY WITH INDEX NUMBER AND NODE INFORMATION OF NODE BY SEARCH QUERY EVALUATION UNIT
S810
S811
CONDITION DESCRIBED IN SEARCH QUERY SATISFIED?
YES
NO
S815
SEND NODE INFORMATION FROM SEARCH QUERY EVALUATION UNIT TO SEARCH RESULT NOTIFYING UNIT
GENERATE SEARCH RESULT NOTIFICATION EVENT FROM NODE INFORMATION BY SEARCH RESULT NOTIFYING UNIT, AND NOTIFY EVENT
S816
DOCUMENT ANALYSIS LOOP
S817
2
RETURN

FIG. 9

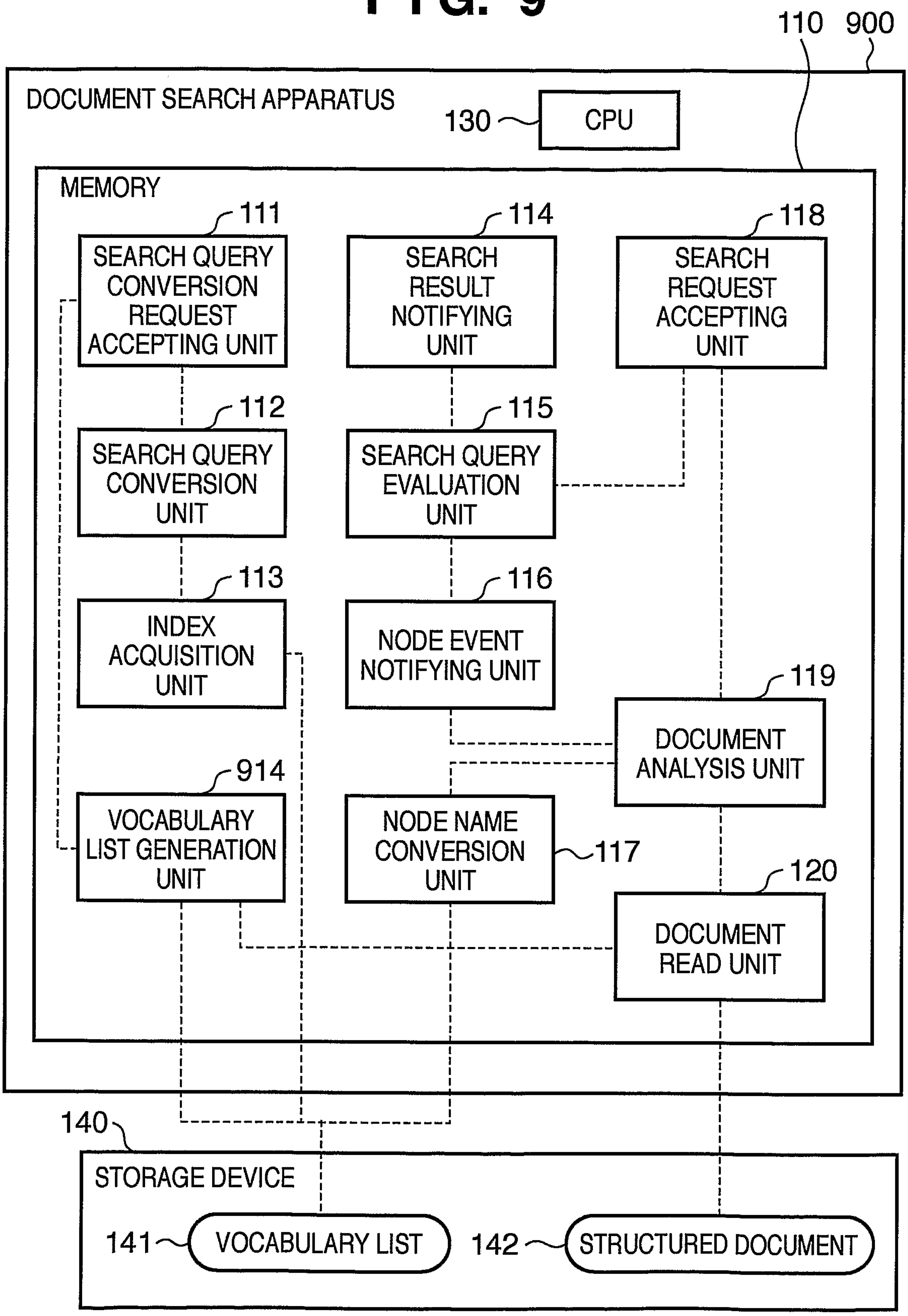
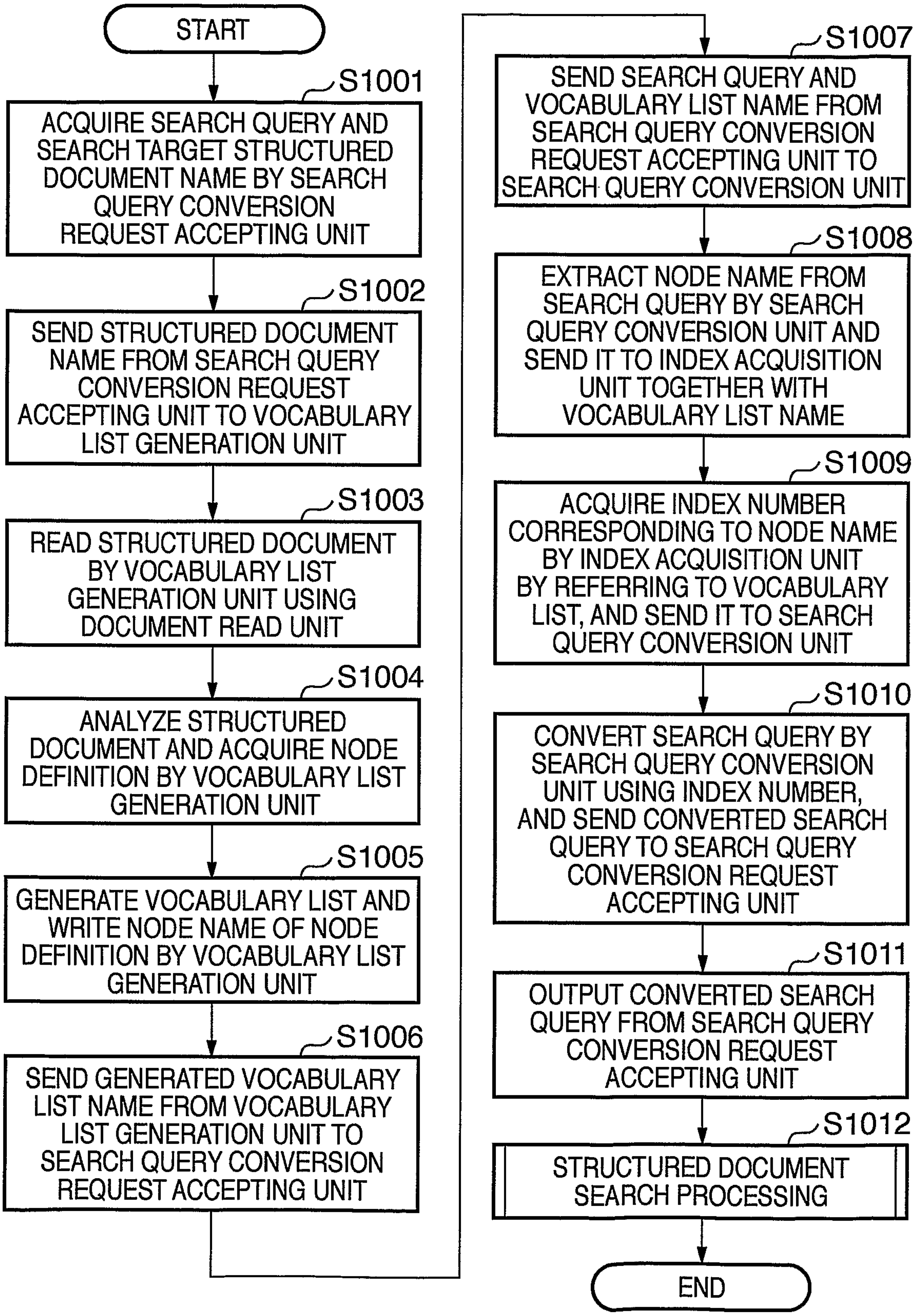

110
900
DOCUMENT SEARCH APPARATUS
130
CPU
MEMORY
111
SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
114
SEARCH RESULT NOTIFYING UNIT
118
SEARCH REQUEST ACCEPTING UNIT
112
SEARCH QUERY CONVERSION UNIT
115
SEARCH QUERY EVALUATION UNIT
113
INDEX ACQUISITION UNIT
116
NODE EVENT NOTIFYING UNIT
119
DOCUMENT ANALYSIS UNIT
914
VOCABULARY LIST GENERATION UNIT
NODE NAME CONVERSION UNIT
117
120
DOCUMENT READ UNIT
140
STORAGE DEVICE
141
VOCABULARY LIST
142
STRUCTURED DOCUMENT

FIG. 10


START
S1001
ACQUIRE SEARCH QUERY AND SEARCH TARGET STRUCTURED DOCUMENT NAME BY SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
S1002
SEND STRUCTURED DOCUMENT NAME FROM SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT TO VOCABULARY LIST GENERATION UNIT
S1003
READ STRUCTURED DOCUMENT BY VOCABULARY LIST GENERATION UNIT USING DOCUMENT READ UNIT
S1004
ANALYZE STRUCTURED DOCUMENT AND ACQUIRE NODE DEFINITION BY VOCABULARY LIST GENERATION UNIT
S1005
GENERATE VOCABULARY LIST AND WRITE NODE NAME OF NODE DEFINITION BY VOCABULARY LIST GENERATION UNIT
S1006
SEND GENERATED VOCABULARY LIST NAME FROM VOCABULARY LIST GENERATION UNIT TO SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
S1007
SEND SEARCH QUERY AND VOCABULARY LIST NAME FROM SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT TO SEARCH QUERY CONVERSION UNIT
S1008
EXTRACT NODE NAME FROM SEARCH QUERY BY SEARCH QUERY CONVERSION UNIT AND SEND IT TO INDEX ACQUISITION UNIT TOGETHER WITH VOCABULARY LIST NAME
S1009
ACQUIRE INDEX NUMBER CORRESPONDING TO NODE NAME BY INDEX ACQUISITION UNIT BY REFERRING TO VOCABULARY LIST, AND SEND IT TO SEARCH QUERY CONVERSION UNIT
S1010
CONVERT SEARCH QUERY BY SEARCH QUERY CONVERSION UNIT USING INDEX NUMBER, AND SEND CONVERTED SEARCH QUERY TO SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
S1011
OUTPUT CONVERTED SEARCH QUERY FROM SEARCH QUERY CONVERSION REQUEST ACCEPTING UNIT
S1012
STRUCTURED DOCUMENT SEARCH PROCESSING
END

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP2010/056277

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl. G06F17/30(2006.01)i

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.Cl. G06F17/30

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Published examined utility model applications of Japan 1922-1996
Published unexamined utility model applications of Japan 1971-2010
Registered utility model specifications of Japan 1996-2010
Published registered utility model applications of Japan 1994-2010

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

JSTPlus(JDreamII)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 2001-34619 A (FUJITSU LIMITED) 2001.02.09, Whole Document, Fig.1-9 (No Family)	1-6
Y	David Geer, Will Binary XML Speed Network Traffic?, Computer, 2005.04, Pages 16-18	1-6
Y	Eduardo Pelegri-Llopart, Saishin Doukou from U.S. GlassFish Tettei Kaibou, Java Expert, 2007.04.25, No.1, Page 15	1-6
Y	JP 2005-135199 A (Nippon Telegraph and Telephone Corporation) 2005.05.26, Whole Document, Fig.1-12 (No Family)	3

□ 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 06.05.2010	Date of mailing of the international search report 18.05.2010
Name and mailing address of the ISA/JP **Japan Patent Office** 3-4-3, Kasumigaseki, Chiyoda-ku, Tokyo 100-8915, Japan	Authorized officer 5M 3364 Hiroki Uejima Telephone No. +81-3-3581-1101 Ext. 3599