

CLAIMS

WHAT IS CLAIMED IS:

1. A computer-implemented method for routing search query results in a networked computing environment, the method comprising:

receiving, at an application executing on a first computing device, a search string comprising at least one key term;

generating a plurality of queries based on the at least one key term, wherein each one of the plurality of queries include at least one query expression;

sending each one of the plurality of queries to at least one information source for evaluation of a corresponding at least one query expression against information accessible to the at least one information source;

receiving a plurality of query results, wherein each one of the plurality of query results include a result classification identifying a type of a corresponding query result; combining at least two of the plurality of query results into a combined query result based on the result classification of each of the at least two of the plurality of query results; and

sending at least the combined query result to a second computing device in the networked computing environment for rendering within an interface executing thereon.

2. The method of claim 1, wherein each of the plurality of query results further includes a priority classification identifying a significance of a corresponding query result.

3. The method of claim 2, further comprising combining the at least two of the plurality of query results into the combined query result based on the priority classification of each of the at least two of the plurality of query results.

4. The method of claim 3, wherein each of the plurality of query results comprises a results table including a collection of information arranged in a column and row format.

5. The method of claim 4, further comprising generating the combined query result by interleaving the results table of the at least two of the plurality of query results.

6. The method of claim 4, further comprising generating the combined query result by stacking the results table of the at least two of the plurality of query results.

7. A computing device, comprising:

a processing unit; and

a system memory connected to the processing unit, the system memory including instructions that, when executed by the processing unit, cause the processing unit to

implement a mapping module, a results module, and a web part module, each configured to route search query results in a networked computing environment, wherein:

the mapping module is configured to:

generate a plurality of queries based on at least one key term, wherein each one of the plurality of queries include at least one query expression; and

send each one of the plurality of queries to at least one information source for evaluation of a corresponding at least one query expression against information accessible to the at least one information source;

the results module is configured to:

receive a plurality of query results, wherein each one of the plurality of query results include a result classification identifying a type of a corresponding query result; and

the web part module is configured to:

combine at least two of the plurality of query results into a combined query result based on the result classification of each of the at least two of the plurality of query results; and

send at least the combined query result to a second computing device for rendering within an interface executing thereon.

8. The computing device of claim 7, wherein the mapping module is further configured to receive a search string comprising the at least one key term.

9. The computing device of claim 7, wherein the web part module includes a plurality of web parts that implement one or more controls that enable end users to view and modify content, appearance, and behavior of Web pages.

10. A computer-readable storage medium having computer-executable instructions that, when executed by a first computing device, cause the first computing device to perform steps comprising:

receive, at an application executing on the first computing device, a search string comprising at least one key term;

generate a plurality of queries based on the at least one key term, wherein each one of the plurality of queries include at least one query expression;

send each one of the plurality of queries to at least one information source for evaluation of a corresponding at least one query expression against information accessible to the at least one information source;

receive a plurality of query results each comprising a results table including a collection of information arranged in a column and row format defined by one of an ordered collection of results and an unordered collection of results, wherein each one of the plurality of query results include a result classification identifying a type of a corresponding query result and a priority classification identifying a significance of a corresponding query result;

combine at least two of the plurality of query results into a combined query result based on the result classification and priority classification of each of the at least two of the plurality of query results, wherein the at least two of the plurality of query results are one of interleaved and stacked to form the combined query result; and

send at least the combined query result to a second computing device for rendering within an interface executing thereon.

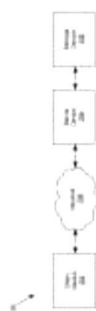


FIG. 1

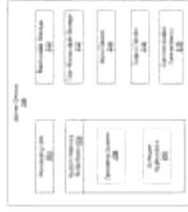


FIG. 2

FIG. 3

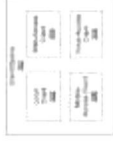


FIG. 3

FIG. 3

FIG. 3

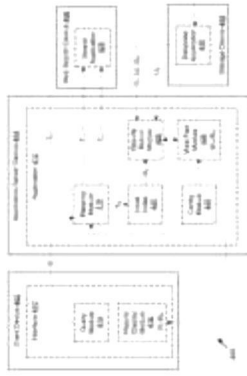


FIG. 4



FIG. 5

ROUTING QUERY RESULTS

BACKGROUND

[0001] A web part is a re-useable software component that implements one or more controls that enable end users to view and modify content, appearance, and behavior of Web pages directly from a browser. In general, web parts may have queries associated with them, and results of a given query may be rendered by particular web part. However, such queries are typically hard-coded, and therefore may preclude the use of conditional logic to drive an associated web part.

SUMMARY

[0002] In one aspect, a computer-implemented method for routing search query results in a networked computing environment is disclosed. The method includes: receiving, at an application executing on a first computing device, a search string comprising at least one key term; generating a plurality of queries based on the at least one key term, wherein each one of the plurality of queries include at least one query expression; sending each one of the plurality of queries to at least one information source for evaluation of a corresponding at least one query expression against information accessible to the at least one information source; receiving a plurality of query results, wherein each one of the plurality of query results include a result classification identifying a type of a corresponding query result; combining at least two of the plurality of query results into a combined query result based on the result classification of each of the at least two of the plurality of query results; and sending at least the combined query result to a second computing device in the networked computing environment for rendering within an interface executing thereon.

[0003] In another aspect, a computing device including a processing unit and a system memory connected to the processing unit is disclosed. The system memory includes instructions that, when executed by the processing unit, cause the processing unit to implement a mapping module, a results module, and a web part module, each configured to route search query results in a networked computing environment. The mapping module is configured to: generate a plurality of queries based on at least one key term, wherein each one of the plurality of queries include at least one query expression; and send each one of the plurality of queries to at least one information source for evaluation of a corresponding at least one query expression against information accessible to the at least one information source. The results module is configured to: receive a plurality of query results, wherein each one of the plurality of query results include a result classification identifying a type of a corresponding query result. The web part module is configured to: combine at least two of the plurality of query results into a combined query result based on the result classification of each of the at least two of the plurality of query results; and send at least the combined query result to a second computing device in for rendering within an interface executing thereon.

[0004] In yet another aspect, a computer-readable storage medium having computer-executable instructions is disclosed that, when executed by a first computing device, cause the first computing device to perform steps including: receive, at an application executing on the first computing device, a search string comprising at least one key term; generate a plurality of queries based on the at least one key term, wherein each one of the plurality of queries include at least one query expression; send each one of the plurality of queries to at least one information source for evaluation of a corresponding at least one query expression

against information accessible to the at least one information source; receive a plurality of query results each comprising a results table including a collection of information arranged in a column and row format defined by one of an ordered collection of results and an unordered collection of results, wherein each one of the plurality of query results include a result classification identifying a type of a corresponding query result and a priority classification identifying a significance of a corresponding query result; combine at least two of the plurality of query results into a combined query result based on the result classification and priority classification of each of the at least two of the plurality of query results, wherein the at least two of the plurality of query results are one of interleaved and stacked to form the combined query result; and send at least the combined query result to a second computing device for rendering within an interface executing thereon.

[0005] This Summary is provided to introduce a selection of concepts, in a simplified form, that are further described below in the Detailed Description. This Summary is not intended to be used in any way to limit the scope of the claimed subject matter. Rather, the claimed subject matter is defined by the language set forth in the Claims of the present disclosure.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] Figure 1 shows an example networked computing environment.

[0007] Figure 2 shows the example server device of Figure 1 in detail.

[0008] Figure 3 shows example logical modules of the client device of Figure 1.

[0009] Figure 4 shows example communications between example computing devices for routing and combining search query results.

[0010] Figure 5 shows a flowchart of an example method for routing search query results.

[0011] Figure 6 shows an example configuration window for configuring parameters for routing search query results.

DETAILED DESCRIPTION

[0012] The present disclosure is directed towards routing search query results in a networked computing environment. In one example embodiment, an initial search query is reformulated into at least one sub-query in accordance with one or more configurable rules. The sub-query is then sent to at least one information system or source, and any potential hits associated with the same are optionally combined and then rendered for viewing. Although not so limited, an appreciation of the various aspects of the present disclosure will be gained through a discussion of the examples provided below.

[0013] Referring now to Figure 1, an example networked computing environment 100 is shown in which aspects of the present disclosure may be implemented. The networked computing environment 100 includes a client device 102, a server device 104, a storage device 106, and a network 108. Other embodiments are possible. For example, the networked computing environment 100 may generally include more or fewer devices, networks, and/or other components as desired.

[0014] The client device 102 and the server device 104 are computing devices, as described in further detail below in connection with Figure 2. In example embodiments, the client device 102 is configured for accessing and interacting with business processes implemented by the server device 104. Example business processes include messaging and communications process, collaboration processes, data management processes, and others.

SHAREPOINT® collaboration server from Microsoft Corporation of Redmond, Washington is an example of a business server that implements business processes in support of collaboration processes, as well as file sharing and web publishing. Other embodiments are possible.

[0015] In some embodiments, the server device 104 includes of a plurality of interconnected, networked devices operating together in a "Farm" configuration. In such a scenario, the networked devices provide a "cloud" computing platform in which one or more applications and data are hosted for one or more clients connected to the cloud computing platform. Still other embodiments are possible.

[0016] The storage device 106 is an electronic data storage device, such as a relational database or any other type of persistent data storage device. The storage device 106 stores data in a predefined format such that the server device 104 can query, modify, and manage data stored thereon. Example data includes information related to directory services, authentication services, administration services, and other services such as managed by the ACTIVE DIRECTORY® directory service from Microsoft Corporation. Other embodiments are possible.

[0017] The network 108 is a bi-directional data communication path for data transfer between one or more devices. In the example shown, the network 108 establishes a communication path for data transfer between the client device 102 and the server device 104. The network 108 can be of any of a number of wireless or hardwired WAN, LAN, Internet, Intranet, or other packet-based communication networks such that data can be transferred among the elements of the example networked computing environment 100.

[0018] Referring now to Figure 2, the server device 104 of Figure 1 is shown in detail. As mentioned above, the server device 104 is a computing device. An example computing device includes a server computer, desktop computer, laptop computer, personal data assistant, smartphone, gaming console, and others.

[0019] The server device 104 includes at least one processing unit 202 and a system memory 204. The system memory 204 stores an operating system 206 for controlling the operation of the server device 104 or another computing device. One example operating system is the WINDOWS® operating system from Microsoft Corporation. Other embodiments are possible.

[0020] The system memory 204 includes one or more software applications 208 and may include program data. Software applications 208 may include many different types of single and multiple-functionality programs, such as a server program, an electronic mail program, a calendaring program, an Internet browsing program, a spreadsheet program, a program to track and report information, a word processing program, and many others. One example program is the Office suite of business applications from Microsoft Corporation. Another example program includes SHAREPOINT® collaboration server, also from Microsoft

Corporation of Redmond, Washington. Still other programs are possible.

[0021] The system memory 204 is computer-readable media. Examples of computer-readable media include computer-readable storage media and communication media. Computer-readable storage media is physical media that is distinguished from communication media.

[0022] The phrase "computer-readable" generally refers to information that can be interpreted and acted on by a computer or computing device. The phrase "storage media" or, equivalently, "storage medium" refers to the various types of physical or tangible material on which electronic data bits are written and stored. Since it is not possible to store information in a transient signal, "computer-readable storage media" as defined within the context of the present disclosure excludes transient signals.

[0023] Computer-readable storage media includes physical volatile and nonvolatile, removable and non-removable media implemented in any method or technology for storage of information, such as computer-readable instructions, data structures, program modules, or other data. Computer storage media also includes, but is not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, DVD or other optical storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store the desired information and which can be accessed by the server device 104. Any such computer storage media may be part of or external to the server device 104. Such storage is illustrated in Figure 2 by removable storage 210 and non-removable storage 212.

[0024] Communication media is typically embodied by computer-readable instructions, data structures, program modules, or other data, in a transient modulated data signal, such as a carrier wave or other transport mechanism, and includes any information delivery media. The term "modulated data signal" refers to a signal that has one or more of its characteristics set or changed in such a manner as to encode information in the signal. By way of example, communication media includes wired media such as a wired network or direct-wired connection, and wireless media such as acoustic, RF, infrared and other wireless media.

[0025] The server device 104 also includes any number and type of an input device 214 and an output device 216. An example input device 214 includes a keyboard, mouse, pen, voice input device, touch input device, motion input device, and others. For example, the input device 214 may be a camera operative to capture and record motions and/or gestures made by a user. The input device 214 may be further operative to capture words spoken by a user, such as by a microphone, and/or capture other inputs from user such as by a keyboard and/or mouse. Consistent with embodiments of the present disclosure, the input device 214 may comprise any motion detection device capable of detecting the movement of a user. For example, the input device 214 may comprise a KINECT® motion capture device, from Microsoft Corporation. Other embodiments are possible.

[0026] An example output device 216 includes a display, speakers, printer, and others. The server device 104 also includes a communication connection 218 configured to enable communications with other computing devices over a network (e.g., network 108 of Figure 1) in a distributed computing system environment.

[0027] The client device 102 of Figure 1 is also a computing device configured in a manner similar to that of the server device 104 described above. Referring now to Figure 3, the client

device 102 is also configured to include one or more different types of interfaces to the server device 104. In the example shown, the client device 102 includes a local client 302, a web-access client 304, a mobile-access client 306, and a voice-access client 308.

[0028] The local client 302 is configured as a dedicated messaging and collaboration client that serves as an interface to the server device 104, and is part of a suite of applications executing on the client device 102. In one embodiment, the local client 302 includes the OUTLOOK® messaging and collaboration client, which is an e-mail application that is part of the Office suite of business applications from Microsoft Corporation. A user can compose, interact with, send and receive e-mails with the OUTLOOK® messaging and collaboration client. Other embodiments of the local client 302 are possible. For example, in one embodiment, the local client 302 includes the Office Communicator client from Microsoft Corporation, an instant messaging client used with Office Communications Server. Still other embodiments are possible.

[0029] The web-access client 304 is configured to access the server device 104 remotely via a network connection, such as the Internet. In one embodiment, the web-access client 304 is the Outlook Web Access (OWA) webmail service of Exchange Server. In the example embodiment, the client device 102 uses a web browser to connect to Exchange Server via Outlook Web Access. This brings up a user interface similar to the interface in the OUTLOOK® messaging and collaboration client in which a user can compose, interact with, send and receive e-mails. Other embodiments of the web-access client 304 are possible. For example, the web-access client 304 may be configured to connect to SHAREPOINT® collaboration server to access corresponding collaboration, file sharing and web publishing services. Still other embodiments are possible.

[0030] The mobile-access client 306 is another type of client interface to the server device 104. In one embodiment, the mobile-access client 306 includes the Mobile Access with ACTIVESYNC® synchronization technology, or the Windows Mobile Device Center for WINDOWS VISTA® operating system or Windows 7 operating system, all

from Microsoft Corporation. Example mobile devices include a cellular telephone, smartphone, a personal digital assistant, and others. Other embodiments of the mobile-access client 306 are possible.

[0031] The voice-access client 308 is yet another type of client interface to the server device 104. In some embodiments, the voice-access client 308 includes Exchange Unified Messaging that is supported in Exchange Server. With Unified Messaging, users have one inbox for e-mail and voicemail. Voicemails are delivered directly into the OUTLOOK® messaging and collaboration client inbox. The message containing the voicemails may also include an attachment. Other embodiments of the voice-access client 308 are possible.

[0032] Referring now to Figure 4, an example block diagram 400 illustrates example communications between computing devices for routing and combining search query results in accordance with the present disclosure.

[0033] The block diagram 400 includes a client device 402, an application server device 404, a web search device 406, and a storage device 408. Other embodiments are possible. For example, the block diagram 400 may generally include more or fewer devices and/or other

components as desired.

[0034] The client device 402 is similar to the client device 102 described above, and includes an interface 410 configured for accessing and interacting with business processes implemented by an application 412 executing on the application server device 404. The application server device 404 is similar to the server device 104 described above.

[0035] The interface 410 includes a query module 414 and a results display module 416, and the application 412 includes a mapping module 418, a local index 420, a configuration module 422, a web part module 424, and a results return module 426. Functionality of the respective example modules of the interface 410 and application 412 are described in further detail below.

[0036] The web search device 406 is a computing device and includes a search application 428 generally configured to search, crawl, and/or index information on the World Wide Web. An example of such a search application includes the BING® web search engine from Microsoft Corporation. The storage device 408 is also a computing device and includes a database application 430 generally configured to store, access and/or modify a collection of electronic data. An example of such a database application includes the ACTIVE DIRECTORY® service from Microsoft Corporation. Other embodiments of the search application 428 and database application 430 are possible.

[0037] In one example communications sequence, the query module 414 of the interface 410 receives an example string $S = \text{"skiing vacation"}$ from a user, and sends the string S to the mapping module 418 of the application 412. In one embodiment, the user enters the string S in the query module 414 (e.g., within a text field), and then executes a search query against the string S by engaging a selectable control (e.g., a button) within the query module 414. Other embodiments are possible.

[0038] The mapping module 418 receives the string S from the query module 414, reformulates the string S into a plurality of search queries T_i - T_M , where M is an arbitrary integer value, and sends each of the plurality of search queries T_i - T_M to at least one information source. For example, search query $T_i = \text{"ski equipment"}$ may be generated by the mapping module 418 and then sent to the search application 428, search query $T_2 = \text{"ski resort reviews"}$ may be formed and then transferred to the search application 428 and the database application 430, and search query $T_M = \text{"ski travel promotions"}$ may be constructed and then directed to the local index 420, which contains data structured in accordance with a schema associated with the application 412. In general, the plurality of search queries T_i - T_M are formed in accordance with at least one mapping rule pre-defined via the configuration module 422 by a user (e.g., an administrator), discussed in further detail below in connection with Figure 5.

[0039] The results return module 426 of the application 412 receives a plurality of query results U_i - U_N , where N is an arbitrary integer value, from the search application 428, database application 430, and/or the local index 420 in view of the plurality of search queries T_i - T_M previously being sent to the same.

[0040] In the current example, query results U_i and U_2 may be more closely related to each other than query results U_3 , U_4 , and U_N in that these respective results may include at least one potential "hit" associated with the phrase "ski equipment" of the example search query

Ti. For example, query result U_i may include a results table of "ski jackets" which are of "equipment" type, and query result U_2 may include a results table of "ski goggles" which are also of "equipment" type; whereas query results U_3 and U_N may include a results table of "ski resorts" which are of "review" type, and query result U_4 may include a results table of "marketing information" which is of "promotion" type. Other embodiments are possible.

[0041] The web part module 424 of the application 412 accesses the plurality of query results U_1 - U_N within the results return module 426 and generally operates on the same. Specifically, the web part module 424 includes a plurality of web parts W_i - W_o , where O is an arbitrary integer value. A web part is a re-useable software component that implements one or more controls that enable end users to view and modify content, appearance, and behavior of Web pages directly from a browser.

[0042] In example embodiments, each of the plurality of web parts W_i - W_o accesses and evaluates each of the plurality of query results U_1 - U_N within the results return module 426. When a given web part is compatibly-configured to display at least one specific "type" of result associated with each of the respective plurality of query results U_1 - U_N , then that particular web part renders an associated result(s) within at least one of a plurality of result sections R_i - R_p , where P is an arbitrary integer value, within the results display module 416 of the interface 410.

[0043] For example, assuming that the web part W_i is configured to render a query result of "promotion" type, when this web part accesses and evaluates the plurality of query results U_1 - U_N within the results return module 426, the query result U_4 is picked-up by the web part W_i and rendered within at least one of a plurality of result sections R_i - R_p within the results display module 416. Similarly, when respective web parts W_2 - W_o access and evaluate each of the plurality of query results U_1 - U_N in the results return module 426, one or more of the plurality of query results U_1 - U_N are picked-up and rendered within at least one of a plurality of result sections R_i - R_p .

[0044] For example, assuming that the web part W_2 is configured to render a result of "review" type, when this web part accesses and evaluates the plurality of results U_1 - U_N in the results return module 426, the query results U_3 and U_N are picked-up and rendered within at least one of a plurality of result sections R_i - R_p . In this example, the web part W_2 may concatenate or otherwise combine the query results U_3 and U_N in accordance with at least one combining rule pre-defined via the configuration module 422, discussed in further detail below in connection with Figures 5 and 6.

[0045] Referring now to Figure 5, an example method 500 for routing and combining search query results is shown. In the example embodiment, the method 500 is implemented by the application 412 executing on the application server device 404 described above in connection with Figure 4. Other embodiments are possible. For example, one or more modules of the application 412 may be located wholly or in part on one or more different computing devices in a networked computing environment in a "cloud" computing implementation.

[0046] The method 500 begins at a first module 502. The first module 502 is configured to receive a search string comprising at least one key term for comparison

against content accessible to at least one information source (e.g., local index 420, search application 428, database application 430, etc.). In example embodiments, a "key term" can

be any one of a user entered query term, a web part configured query term, and site navigation context or user context. In this manner, a "key term" can be any type of input (i.e., entered by a user and/or by configuring a web part and/or by the context of the web part) that is taken into consideration when it comes to determining recall and/or sorting/ranking according to the principles of the present disclosure. Other embodiments are possible.

[0047] Operational flow then proceeds to a second module 504. The second module 504 is configured to map the search string received by the first module 502 into at least one search query (e.g., "Author: John Doe" and "Document: .ppt") in accordance with one or more query mapping rules. In one embodiment, a particular query mapping rule additionally assigns at least one "type" to one or more expressions of a search query. For example, a search query expression "John Doe" might be assigned as "user" type and "author" type and a search query expression ".ppt" might be assigned as "document" type. Other embodiments are possible.

[0048] Operational flow then proceeds to a third module 506. The third module 506 is configured to send the search query generated by or at the second module 504 to the at least one information source. In general, the information source is accessible over a network (e.g., network 108) within a networked computing environment (e.g., environment 100). Other embodiments are possible.

[0049] Operational flow then proceeds to a fourth module 508. The fourth module 508 is configured to receive a plurality of query results in view of the at least one search query sent by the third module 506. In one embodiment, each of the plurality of results is generically formatted as a results table collection, which is a set of results tables. In general, a results table may consist of a collection of results or information arranged in a row/column format in an ordered arrangement such as, for example, results listed according to a pre-defined ranking scheme (e.g., priority, significance, confidence, etc.), or an unordered arrangement (e.g., raw data). A results table may additionally include at least one "type" associated with the same. For example, a results table including a collection of ".ppt" results might be assigned as "document" type, and a results table including a collection of individual names might be assigned as "user" type and/or "author" type. Other embodiments are possible.

[0050] Operational flow then proceeds to a fifth module 510. The fifth module 510 is configured to optionally (indicated by intermittent lines in Figure 5) combine two or more of the plurality of query results received by the fourth module 508. In one embodiment, example methods for combining query results include interleaving, stacking, and others. In general, rules for combining query results are defined via the configuration module 422 of Figure 4, described in further detail below in connection with Figure 6. Other embodiments are possible.

[0051] Operational flow then proceeds to a sixth module 512. The sixth module 512 is configured to render one or more of the plurality of query results within at least one of the plurality of result sections Ri-Rp within the results display module 416 of the interface 410 of the client device 402 of Figure 4.

[0052] Example logic that can be used to implement query mapping rule(s) and the rendering of query results within a client interface is described in U.S. Patent Application Serial No. 13/287, 999 (Attorney Docket No. 14917.1952US01 / 334101.01) as stated above. Other embodiments are possible.

[0053] Referring now to Figure 6, an example configuration window 600 for configuring web parts and search results for routing and combining search query results is shown in accordance with the present disclosure. In the example embodiment, the configuration window 600 includes a web part configuration section 602 and a search results configuration section 604. In general, the configuration window 600 is implemented and/or accessed via the configuration module 422 of the application 412 of Figure 4. Other embodiments are possible.

[0054] The web part configuration section 602 includes a web part selection 606 and a results class selection 608. The web part selection 606 enables a user to select a particular web part (e.g., web part Wi) at least for the purpose of assigning at least one query result class to the particular web part indicating at least one "type" of query result the web part will be configured to render. The "type" of query result (e.g., "Promotion") that the particular web part will be configured to render is assigned via the results class selection 608. In the example embodiment, a plurality of other web parts and a plurality of other results classes (identified by the characters in Figure 6) is accessible by user manipulation of the web part selection 606 and results class selection 608, respectively.

[0055] The search results configuration section 604 includes a search source selection 610 and a ranking model selection 612. The search source selection 610 enables a user to identify at least information source (e.g., search application 428) a particular web part (e.g., web part Wi) can expect to potentially evaluate query results from. The ranking model selection 612 enables a user to define one or more rules or methods for combining query results such as, for example, interleaving, stacking, and others. In the example embodiment, a plurality of other search sources and a plurality of other ranking models (identified by the characters in Figure 6) is accessible by user manipulation of the search source selection 610 and ranking model selection 612, respectively.

[0056] Example rules include definitions that combine results tables when corresponding tables having a similar type such as, for example, "combine result tables having a "user" type and an "author" type," or a dissimilar type such as, for example, "combine result tables having a "document" type and an "author" type." Additionally, results tables may be preferably combined according to a pre-defined ranking scheme or model such as, for example, combining results table according to priority factor or classification. An example of a priority factor includes providing results returned according to a first mapping rule in a preferred position of a results table in comparison to results returned according to a second mapping rule that is defined, in time, following the first mapping rule. Still other embodiments are possible.

[0057] The example embodiments described herein can be implemented as logical operations in a computing device in a networked computing system environment. The logical operations can be implemented as: (i) a sequence of computer implemented instructions, steps, or program modules running on a computing device; and (ii) interconnected logic or hardware modules running within a computing device.

[0058] For example, embodiments of the present disclosure may be practiced via a system-on-a-chip (SOC) where each or many of the components illustrated in Figure 2 may be integrated onto a single integrated circuit. Such an SOC device may include one or more processing units, graphics units, communication units, system virtualization units and application functionality, all of which are integrated onto a chip substrate as a single

integrated circuit.

[0059] Additionally, the logical operations can be implemented as algorithms in software, firmware, analog/digital circuitry, and/or any combination thereof, without deviating from the scope of the present disclosure. The software, firmware, or similar sequence of computer instructions can be encoded and stored upon a computer readable storage medium and can also be encoded within a carrier-wave signal for transmission between computing devices.

[0060] Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described above. Rather, the specific features and acts described above are disclosed as example forms of implementing the claims.