

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
15 March 2007 (15.03.2007)

PCT

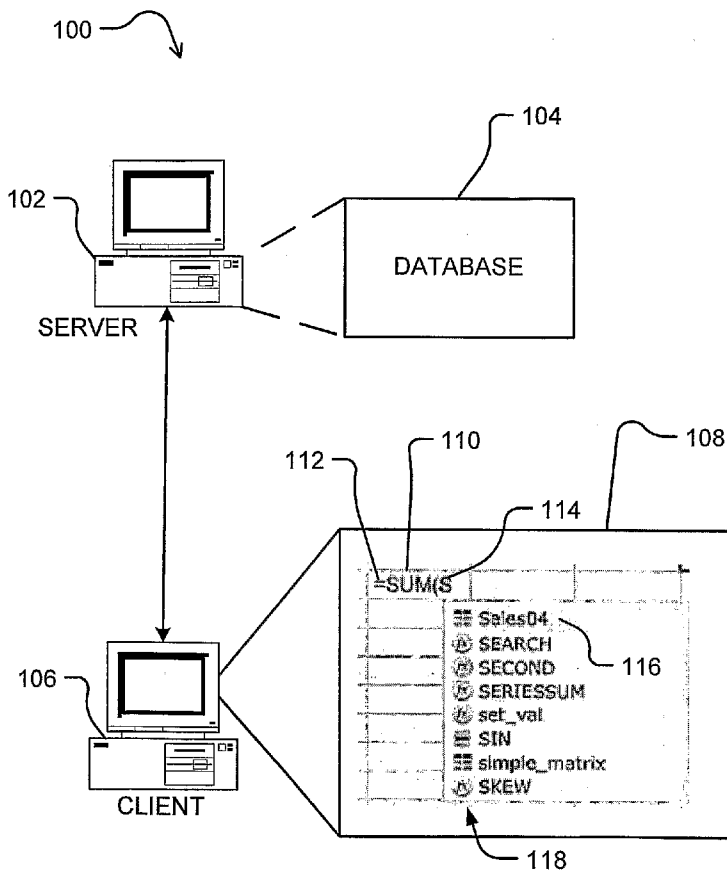
(10) International Publication Number
WO 2007/030692 A1

- (51) International Patent Classification:
G06F 17/30 (2006.01) *G06F 17/00* (2006.01)
- (21) International Application Number:
PCT/US2006/034987
- (22) International Filing Date:
7 September 2006 (07.09.2006)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
11/222,928 8 September 2005 (08.09.2005) US
- (71) Applicant (for all designated States except US): **MICROSOFT CORPORATION** [US/US]; One Microsoft Way, Redmond, Washington 98052-6399 (US).
- (72) Inventors: **MARTYNOV, Alexander**; One Microsoft Way, Redmond, Washington 98052-6399 (US). **GAINER, David, E.**; One Microsoft Way, Redmond, Washington

- 98052-6399 (US). **DICKERMAN, Howard, J.**; One Microsoft Way, Redmond, Washington 98052-6399 (US).
- (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, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, 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, 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, IS, IT, LT, LU, LV, 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).

[Continued on next page]

(54) Title: AUTOCOMPLETING WITH QUERIES TO A DATABASE



(57) Abstract: An autocomplete method and system provides suggested completions for partial data entries based upon predefined coordinates, operands and/or commands. In one embodiment, the partial data entry may be in an active cell of a spreadsheet and the predefined coordinates may be queried from a separate, preconfigured, multi-dimensional database linked to the active cell or the spreadsheet.

WO 2007/030692 A1



Declarations under Rule 4.17:

- *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*
- *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

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.

AUTOCOMPLETING WITH QUERIES TO A DATABASE

Background

Automatically completing spreadsheet entries is a useful tool, particularly for large data entry projects where a user is tasked with entering duplicative data. For example, a user may be required to repeatedly enter the same word or phrase within a column of a spreadsheet, and the "autocomplete" function found in many spreadsheets programs simplifies this task by providing suggestions for completed entries as a user starts to enter data within a spreadsheet cell.

While spreadsheets may be used to present data from many different sources, the autocomplete function has typically been limited to providing suggestions for data entries found within the current spreadsheet. Thus, when a user attempts to address data sources separate from the spreadsheet, such as linking to an entry within a separate database, the user is not prompted with any autocomplete suggestions for the various data locations within the database. This problem may be compounded by the fact that the user is not familiar with the layout or configuration of the database, as well as the fact that the format or syntax of the database may not be intuitive to the user. For example, the user may not know the predefined dimensional labels for a particular database, or the data sources may use complex and non-intuitive command or programming languages. Furthermore, even if a user is familiar with the underlying database structure and syntax, the user will still be required to manually input long strings (i.e., addresses within the database) when linking data from the database within the spreadsheet. Any typographical error or mistake in the syntax of a database label or command will produce an error in the resulting spreadsheet. It is with respect to these and other considerations that the subject of the present disclosure has been made.

Summary

Exemplary embodiments disclosed herein may include a method and system for alleviating the disadvantages noted above, as well as others. Exemplary embodiments disclosed herein may include a method and system for autocompleting entries in an active cell of a spreadsheet from a separate multidimensional database.

Other embodiments may include a method for autocompleting data entries, including accessing a predefined data source, activating the edit mode for an application, determining a

partial data entry in the application, and displaying a list of corresponding data labels based upon the partial entry in the application.

Methods and systems described herein may be implemented as a computer process, a computing system, or as an article of manufacture, such as a computer program product. The computer program product may be a computer storage medium readable by a computer system and encoding a computer program of instructions for executing a computer process. The computer program product may also be a propagated signal on a carrier readable by a computing system and encoding a computer program of instructions for executing a computer process.

Brief Description of the Drawings

FIG. 1 illustrates a system for autocompleting entries from a database according to an exemplary embodiment.

FIG. 2 illustrates an example of a suitable computing system environment on which exemplary embodiments may be implemented.

FIG. 3 illustrates a system for autocompleting entries in an active cell of a spreadsheet program from a preconfigured multidimensional database according to an exemplary embodiment.

FIG. 4 depicts operational characteristics involved in autocompleting entries in a spreadsheet utilizing a database according to an exemplary embodiment.

FIG. 5 depicts operational characteristics involved in autocompleting entries within an active cell of a spreadsheet utilizing a database according to an exemplary embodiment.

FIG. 6 is a flow diagram illustrating a method of autocompleting entries in a spreadsheet from a database according to an exemplary embodiment.

Detailed Description

A system 100 for automatically completing (“autocompleting”) entries in an active cell of a spreadsheet according to an exemplary embodiment is shown in FIG. 1. In this embodiment, system 100 includes a server 102, which in turn includes a data source or database 104. Server 102 is communicatively coupled to client 106. Client 106 includes a spreadsheet application 108. Spreadsheet application 108 includes an active cell 110, which is highlighted by the dark border around the particular active cell 110. Within active cell 110 is partial data entry 114. In this embodiment, partial data entry 114 includes operand 112. Based upon the partial data entry 114, elements 116 are shown.

Elements 116 include data labels/coordinates from database 104 which is coupled to server 106 and correspondingly coupled to the client 106 and the spreadsheet application 108. Furthermore, elements 116 may be functions or other commands that are appropriate for the particular type of database being utilized, or for the type of spreadsheet being created. Database 104 may be queried to suggest completed entries for the active cell 110, based at least in part upon partial data entry 114.

Database 104 is typically a multi-dimensional and/or on-line analytical processing or "OLAP" database. Database 104 may be a "cube" database, which may be a portion of a larger, related database. If database 104 is an OLAP type database, the coordinates of that database may be described as dimension, hierarchy, member, or coordinate of the multi-dimensional database. Database 104 is separate from spreadsheet 108. Indicators 118 may also be included with the suggested elements 116. Indicators 118 may indicate what type of element the suggested elements are, namely, a function, a dimension, hierarchy, member, etc. This may assist the user in determining which suggested element to choose. With this system, a user may not need to know the data labels, functions, commands and/or dimensions of a database that is linked to the current spreadsheet to enter data labels.

Communication between the spreadsheet application and the data source may be accomplished with multidimensional expression language, or MDX. It will be appreciated that other data sources, such as other types of databases, tables, pivot tables, etc. may be utilized without straying from the concepts disclosed herein. Additionally, other command languages or protocols, such as sequential query language (SQL) or other database query languages, may be utilized within the scope of the present invention. Furthermore, although the application is described herein as a spreadsheet, it will be appreciated that other applications, such word processing applications, as well as other applications, may utilize this autocomplete methodology and/or system.

Elements 116 may include all possible functions and coordinates where partial data entry 114 is merely an operand and/or no character entry. Elements 116 may include all available data labels in database 104, as well as other commands and/or operands. In an embodiment, once a character is entered into partial data entry 114 within active cell 110, data labels 116 will include labels that start with the particular character that has been entered or commands that may be appropriate based upon the character(s) entered in the active cell 110. It will also be appreciated that once another character is typed into the active cell 110, the displayed labels 116 may change and/or may be reduced in number.

As the user enters each keystroke, special queries may be created and sent to an external server to fetch one or more elements from the database that correspond to the partial data entry. In the case of a multi-dimensional database, the queries that are sent as well as the elements that are returned may be MDX expressions. The MDX expressions that are returned from the database can be used to build more complex MDX expressions that may eventually be used as part of a larger argument for the spreadsheet function, such as a Microsoft® Excel® Cube function, pivot table, other table, etc. A displayed list may be an autocomplete drop-down list, which permits the user to select an item from the list or continue entering characters. Of course, any time new characters are entered, a new query may be executed.

With system 100, users need not know the particular data labels or elements within a database since they may search for the appropriate data label. This will save time and frustration for the user since the particular data labels may be auto-completed from a list of pre-defined data labels, thereby reducing mistakes and the time required for spreadsheet configuration.

FIG. 2 illustrates an example of a suitable computing system environment 200 on which embodiments of the invention may be implemented. In its most basic configuration, system 200 typically includes at least one processing unit 202 and memory 204. Depending on the exact configuration and type of computing device, memory 204 may be volatile (such as RAM), non-volatile (such as ROM, flash memory, etc.) or some combination of the two. This most basic configuration is illustrated in FIG. 2 by dashed line 206. Additionally, system 200 may also have additional features/functionality. For example, system 200 may also include additional storage (removable and/or non-removable) including, but not limited to, magnetic or optical disks or tape. Such additional storage is illustrated in FIG. 2 by removable storage 208 and non-removable storage 210. Computer storage media includes 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. Memory 204, removable storage 208 and non-removable storage 210 are all examples of computer storage media. Computer storage media includes, but is not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital versatile disks (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 system 200. Any such computer storage media may be part of system 200.

System 200 may also contain communications connection(s) 212 that allow the system to communicate with other devices. Communications connection(s) 212 is an example of communication media. Communication media typically embodies computer readable instructions, data structures, program modules or other data in a modulated data signal such as a carrier wave or other transport mechanism and includes any information delivery media. The term "modulated data signal" means 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, and not limitation, 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.

System 200 also includes input devices 214 such as a keyboard, mouse, pen, voice input device, touch input device, etc. Output device(s) 216 such as a display, speakers, printer, etc. may also be included. All these devices are well known in the art and need not be discussed at length here.

A computing device, such as system 200, typically includes at least some form of computer-readable media. Computer readable media can be any available media that can be accessed by the system 200. By way of example, and not limitation, computer-readable media might comprise computer storage media and communication media, as described above.

FIG. 3 illustrates a system 300 for auto-completing entries in an active cell of a spreadsheet application, which is communicationally coupled to a database according to an exemplary embodiment. System 300 in this embodiment includes a server 302 which correspondingly has one or more databases stored thereupon 304. System 300 may optionally include a network 306 such as a LAN, WAN, the Internet or other network which server 302 may be coupled to.

System 300 in this embodiment also includes a client 308. Client 308 includes a communication module 310 communicationally coupled to a spreadsheet application 312. Furthermore, communication module 310 is coupled to the network 306. It should be appreciated that communication module 310 may also be directly coupled to server 302 and/or directly to database(s) 304.

When a user sets up a new spreadsheet in spreadsheet application 312, they may chose from a list of databases 304 that may be linked to. Alternatively, the user may type in the location of a database to be linked to. This link may then be given a label name, such that

this name is used by a query module 311 to construct a query for the named database. Query module 311 is configured to receive the partial data entry and the name of the linked database, and then construct a query that will be passed on to server 302 (via communication module 310) to be interpreted. The database 304 is then queried and, in response to the query, returns suggested completed elements to the communication module 310. Communication module 310 then passes those suggested completed elements on to the spreadsheet application 312, which in turn passes the suggestions to user interface 314 for selection by the user.

Databases 304 may be multi-dimensional and/or OLAP-type databases, or other types of external data sources that can be linked to spreadsheet 312. Communication module 310 may be located on client 308, however it may also be included on server 302 or may be included in database(s) 304, among other locations. Communication module 310 is typically provided by database 304 such that the client 308 and spreadsheet application 312 may communicate with the database(s) 304. In one embodiment, communication module 310 may comprise a dynamic-link library (DLL) that is provided (and configured) by the particular linked database, or database type.

Query module 311 is shown in one embodiment as being separate from spreadsheet application 312. However, query module 311 may be included within the spreadsheet application 312. The location of query module 311 may also be other than in the client 308, such as within the server 302, or a remote location. The query module 311 allows suggested completed elements to be created and presented dynamically, rather than statically.

As databases 304 may be of different types and located in different locations, as well as have multiple dimensions, it may be difficult for a user to have thorough knowledge of the data labels, expressions, and/or dimensions, labels, coordinates of the various databases, data, metadata, and/or portions or combinations thereof. Therefore, autocompleting from a defined list of elements/data labels/dimension labels/dimensions/coordinates may be extremely helpful in that the user would not have to know beforehand all of the particular data labels or elements of all of the databases 304 needed to configure the spreadsheet application 312.

FIG. 4 shows a portion of a spreadsheet 400, which includes a highlighted or activated cell 402, as delineated by the dark border 403 around the active cell 402. Within active cell 402, there may be an operand 404 as well as a partial data entry 406. Operand 404 may be a portion of the partial data entry 406. Corresponding to partial data entry 406 is a displayed

list of elements 408 which have been returned via a query from a linked database. Elements 408 correspond to partial data entry 406, which are used along with the name of the link to the database to create the query for the linked database. In this example, all the suggested elements 408 are functions that may be used based upon the partial data entry 406, the type of spreadsheet, and/or the type and particular database linked to. In this manner, elements 408 are suggested based in part upon partial data entry 406. As the user enters more characters, new queries are created, and different suggested completed elements are returned. Consequently, the displayed suggested data list may change and/or may be reduced in size depending on the characters entered.

FIG. 5 shows a portion of a spreadsheet 500, which in this embodiment includes an active cell 502. Within active cell 502, may be a partial data entry 506. In this embodiment, partial data entry 506 includes the letter "s". And correspondingly elements from the linked database 508 corresponding to the letter "s", from within a database linked for this particular active cell 502. If user typed in another letter such as "e", that displayed list of elements 508 would change to only elements that begin with "se" such as "search", "second", "seriessum" and "set_val", as shown, etc. Furthermore, characters other than letters may be entered. For instance entering a "(" may indicate that the user may want to enter a command and/or a data label. In this instance commands, elements, and/or data labels may be displayed for the user, based upon the particular attributes of the linked database.

Along with suggested elements 508, indicators 510, 512, 514 may be included. Indicators 510, 512, 514 may indicate what type of element the suggested element 508 is. Indicator 510 indicates that the corresponding element directly to the right of it is a function that may be utilized based at least in part upon the partial data entry 506. Indicator 512 may indicate that the corresponding element is a dimension from the linked database. Indicator 514 may indicate that the corresponding element is a hierarchy or member from the linked database. Indicators 510, 512, 514 may enable a user to better, and more easily enter the appropriate element and more easily construct a spreadsheet.

FIG. 6 displays an operational flow 600 incorporating aspects of the present invention. In an exemplary embodiment, flow 600 begins with define operation 602, which creates and defines a database. A database is defined or created when the particular coordinates are defined and saved, and the data fields populated. This step may be optional in that a user may link to databases that have already been created and defined.

Create operation 604 creates a spreadsheet. Create option 604 is also optional in that a spreadsheet may have already been created and/or opened such that a user may activate the edit mode of a cell to utilize the auto-complete function.

Once a database and spreadsheet have been created, the spreadsheet may link to and query the database. It should be appreciated that other types of application programs and/or databases may utilize this method to auto-complete the data labels, dimension labels, elements, and/or dimensions of any database for a user.

Next, operation 606 couples the spreadsheet to the database. When the spreadsheet is created and particular cells are defined within the spreadsheet, different databases may be linked to the information in that cell and/or spreadsheet, such that the spreadsheet autocomplete function would have a reference database to provide suggestions and/or display completed elements, etc. to the user. Once a link is created, each link may be identified or labeled. This link label may be utilized, along with the partial data entry in the active cell, to create a query. The query is to the linked database, and returns suggested completed elements in the form of a list that the user may select from. If the user selects or accepts a suggested completed element, it will be added to the existing string, if any, in the active cell. If the user then begins to add another data item, a new query will be created. Additionally the user may enter another type of command that may be acted upon by the system, such as a function, or if the user continues typing without accepting any of the suggested completions

Once a spreadsheet is opened and is linked to a database, the user may employ an activation operation 608 to activate a cell of the spreadsheet. This may be accomplished in various ways including clicking on the cell, double clicking on the cell, and other methods such as using a keyboard to enter the edit mode for a cell in a spreadsheet or other application. In an embodiment, a dark border around the particular cell indicates the particular cell is activated. It will be appreciated that other methods for activating a cell and indicating an active cell may be utilized.

Following activation of a cell, identify operation 610 identifies a partial data entry within the active cell. This may include no entry, i.e. no characters entered into the active cell as the partial data entry. The partial data entry also may be letters, parentheses, a comma, bracket, and/or a "{", among other types of characters. The spreadsheet application then utilizes a query module to construct a query for the linked database. In one embodiment, the query module sends the query to a communication module. The communication module may

be a data logic library (DLL) configured by the particular linked database or database type, as described above. The query is then sent to the database or database server that interprets the query and queries the database. The query then returns the suggested completed elements to the communication module. The suggested completed elements are then sent to the spreadsheet for selection/acceptance by the user. As the partial data entry changes, a new query will be created and executed to retrieve an updated list of suggested completed elements from the linked data source.

It will be appreciated that as the partial data entry changes, so will the elements that correspond to that particular partial data entry. The list of potential elements may be the defined data labels, dimensions, dimension labels, function, other elements, and/or coordinates that are based at least in part upon the partial data entry in the active cell. Furthermore, it should be appreciated that the list of potential elements may also include operands and/or other database descriptors or commands that provide possible matches for the partial data entry.

Following identification of the partial data item, display operation 612 displays the corresponding elements for the partial data entry on a user interface. As noted above, the corresponding elements are typically coordinates, functions, dimensions and/or data labels from the database linked to that particular cell and/or spreadsheet.

An optional decision 614 determines if the partial data entry has changed. If the user has changed the partial data entry, the "YES" leg is followed back to identify the new partial data entry at 610. A new query will be created and accomplished, and display operation 612 will update the display of the corresponding elements for selection by the user. If the partial data entry has not changed at 614, then the "NO" leg is followed to end the flow at 616.

FIG. 7 illustrates a user interface 700 for editing and defining a link to a data source. Interface 700 may be one way to define the link between the application program and the data source. In this embodiment, interface 700 includes a connection name portion 702. Connection name portion 702 is configured to display and allow editing of the name/description of the connection or link. This name is used by the query module to identify and query the linked database.

Interface 700 also includes a description portion that allows editing and defining a description of the link. This description may include the data in the linked database, the linked database name, etc. This description may be user by the user to better describe the contents of the database, such that a spreadsheet is easier to understand and construct.

Interface 700 may also include a connection file portion 706. Connection file portion 706 allows easy identification (and editing) of the location of the connection file.

Interface 700 also includes a connection string portion 708. Connection string portion 708 displays the connection string for purposes of editing the string. The connection string may include a type of database linked to the spreadsheet (i.e. OLAP, SQL, etc.). It may also include a type of communication protocol, security features, etc. Many other types of data and definitions may be displayed and edited utilizing this interface.

The logical operations of the various embodiments of the present invention may be implemented (1) as a sequence of computer implemented acts or program modules running on a computing system and/or (2) as interconnected machine logic circuits or circuit modules within the computing system. The implementation is a matter of choice dependent on the performance requirements of the computing system implementing the invention. Accordingly, the logical operations making up the embodiments of the exemplary embodiments described herein are referred to variously as operations, structural devices, acts or modules. It will be recognized by one skilled in the art that these operations, structural devices, acts and modules may be implemented in software, in firmware, in special purpose digital logic, and/or any combination thereof without deviating from the spirit and scope of the present disclosure as recited within the claims attached hereto.

Although the exemplary embodiments have been described in language specific to computer structural features, methodological acts and by computer readable media, it is to be understood that the exemplary embodiments defined in the appended claims are not necessarily limited to the specific structures, acts or media described. As an example, different formats other than MDX may be used to configure and/or access a database or other data source. Therefore, the specific structural features, acts and mediums are disclosed as exemplary embodiments implementing the claimed invention.

The various embodiments described above are provided by way of illustration only and should not be construed to limit this disclosure. Those skilled in the art will readily recognize various modifications and changes that may be made to the present disclosure without following the exemplary embodiments and applications illustrated and described herein, and without departing from the true spirit and scope of the present disclosure, which is set forth in the following claims.

Claims

WHAT IS CLAIMED IS:

1. A computer readable medium encoding a computer program of instructions for executing a computer process for suggesting completion of a partial data entry for an active cell in a spreadsheet from a database, wherein the database is separate from the spreadsheet and connected to the spreadsheet by a communications link, said computer process comprising:
 - defining a partial data entry within the active cell;
 - querying the database to identify an element from the database corresponding to said partial data entry; and
 - displaying the corresponding element as a suggested completion for the partial data entry.
2. The computer readable medium of claim 1, further comprising:
 - receiving an acceptance command in association with said suggested completion.
3. The computer readable medium of claim 2, further comprising:
 - responsive to said acceptance command, storing said matching completed elements within the active cell.
4. The computer readable medium of claim 2, further comprising:
 - receiving a command pertinent to said suggested completion;
 - operating on said suggested completion in accordance with said command; and
 - if said command contains a modified partial data entry, then displaying a modified suggested element that corresponds to said modified partial data entry.
5. The computer readable medium of claim 1, further comprising:
 - displaying multiple suggested completions in a list adjacent the active cell to allow a user to select one of the suggested completion from the list.
6. The computer readable medium of claim 1, further comprising:

modifying the partial data entry in the active cell in response to a user input; and adjusting the list of suggested completions to correspond to the modified partial data entry.

7. The computer readable medium of claim 1, wherein the suggested completion comprises a dimension, hierarchy, member, or coordinate of the database.
8. The computer readable medium of claim 1, wherein the database is an OLAP database.
9. A system for autocompleting entries within an active cell of a spreadsheet, comprising:
 - a database contained on a server device;
 - a client device comprising a spreadsheet application having an active cell with a partial data entry; and
 - a communication module configured to communicate between the client device and the server device to retrieve suggested completed elements from the database based upon the partial data entry within the active cell, wherein the completed elements from the database are displayed for selection by a user of the spreadsheet application.
10. The system of claim 9, further comprising a query module configured to receive the partial data entry and a predefined link to a database, wherein the query module creates a query for the linked database based on the partial data entry.
11. The system of claim 9, wherein the communication module is a dynamic-link library provided by the database.
12. The system of claim 9, wherein the communication module is configured by the database.
13. The system of claim 9, wherein at least one suggested completed element is a coordinate of the database.

14. The system of claim 9, wherein at least one suggested completed element is a function capable of being utilized in the spreadsheet application.

15. A computer readable medium encoding a computer program of instructions for executing a computer process for autocompleting an entry in an active cell of a spreadsheet utilizing a preconfigured database, said computer process comprising:
 - linking the spreadsheet to the preconfigured database;
 - defining a partial data entry within the active cell;
 - querying the database to find elements corresponding to the partial data entry;
 - displaying the corresponding elements;
 - receiving an acceptance command related to one of the displayed elements; and
 - storing the accepted element within the active cell.

16. The computer readable medium of claim 15, further comprising:
 - modifying the partial data entry; and
 - querying the database to identify elements corresponding to the modified partial data entry.

17. The computer readable medium of claim 15, wherein the database is an OLAP database.

18. The computer readable medium of claim 17, wherein the displayed elements comprise coordinates of the database.

19. The computer readable medium of claim 15, wherein at least one suggested completed element is a function capable of being utilized in the spreadsheet application.

20. The computer readable medium of claim 15, wherein the partial data entry comprises no character and the suggested elements comprise all possible functions, and coordinates from the database.

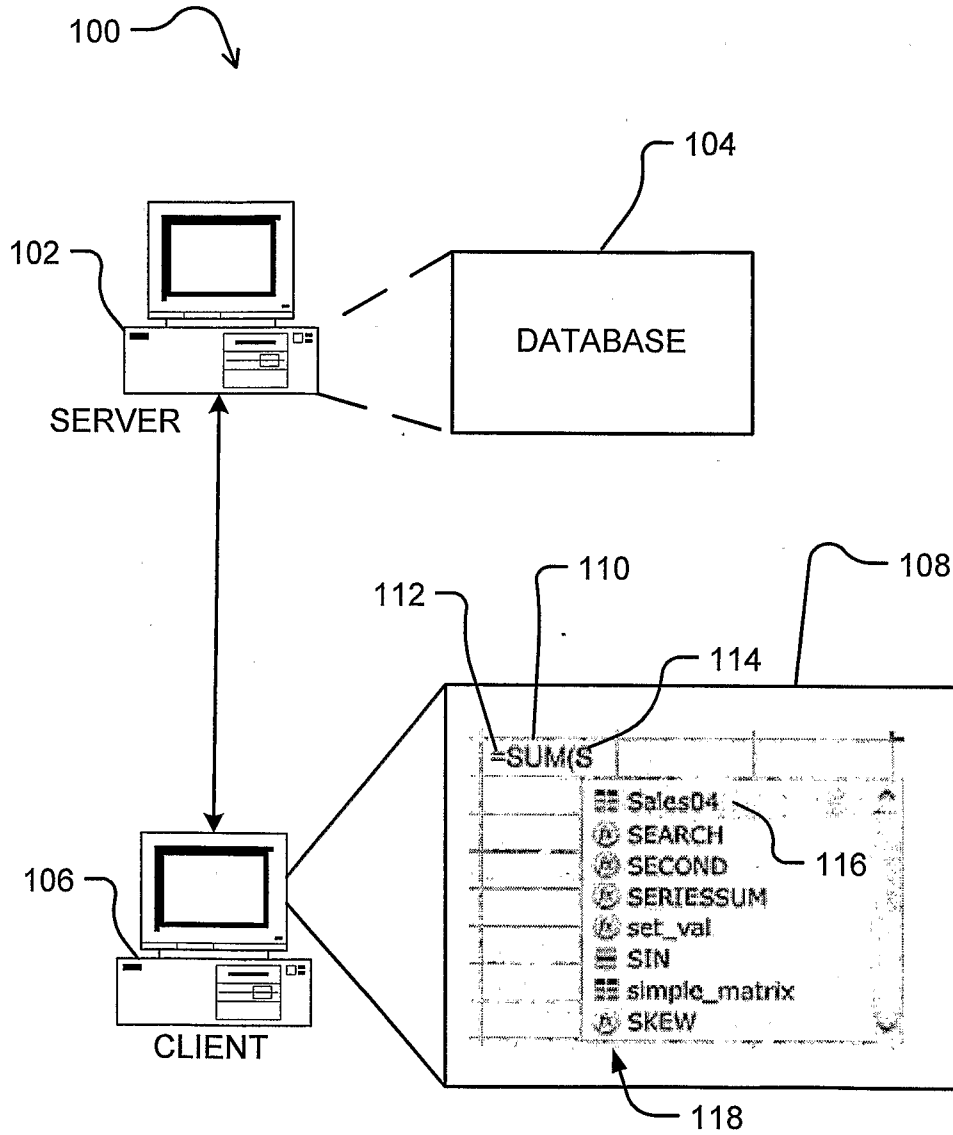


FIG. 1

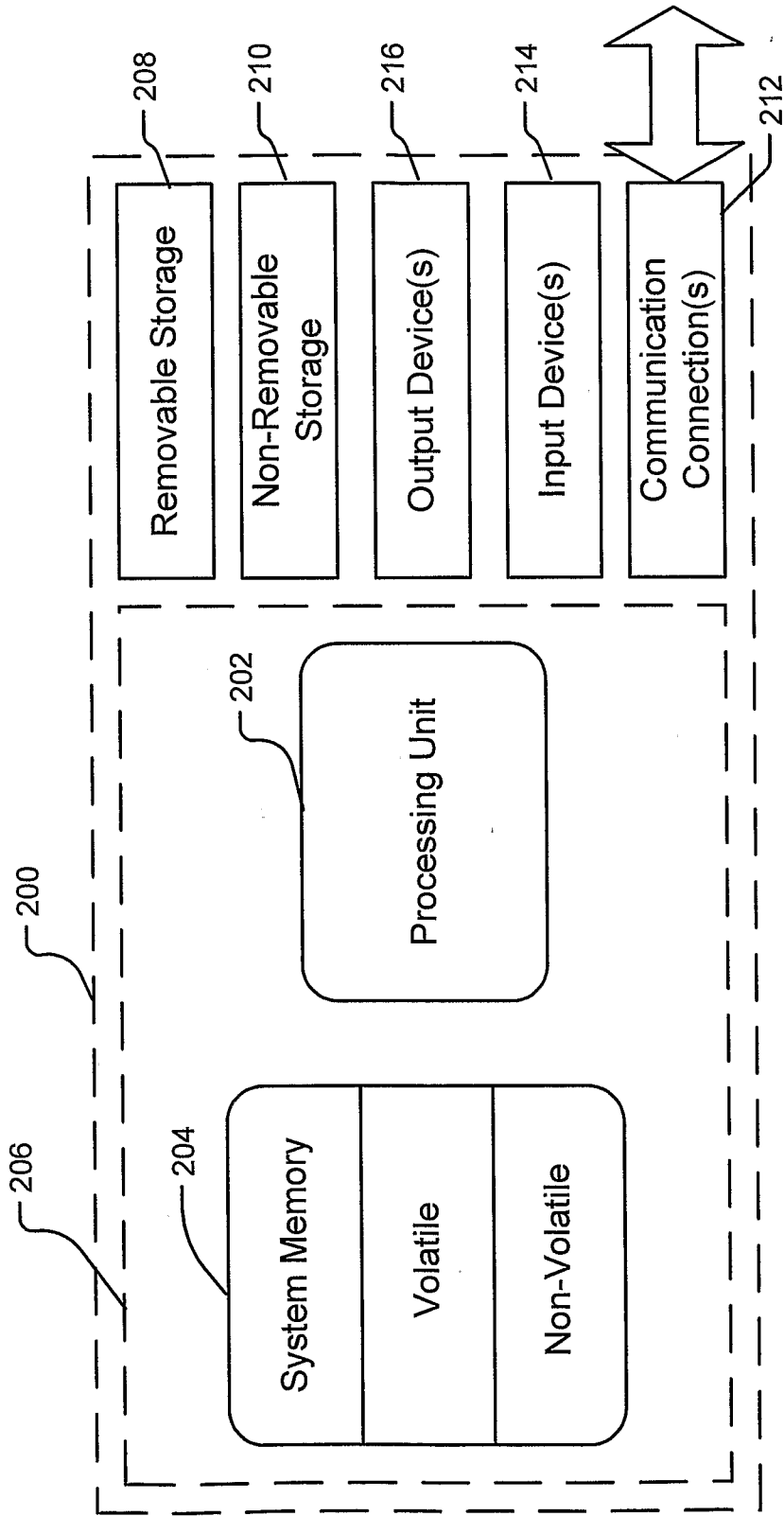


FIG. 2

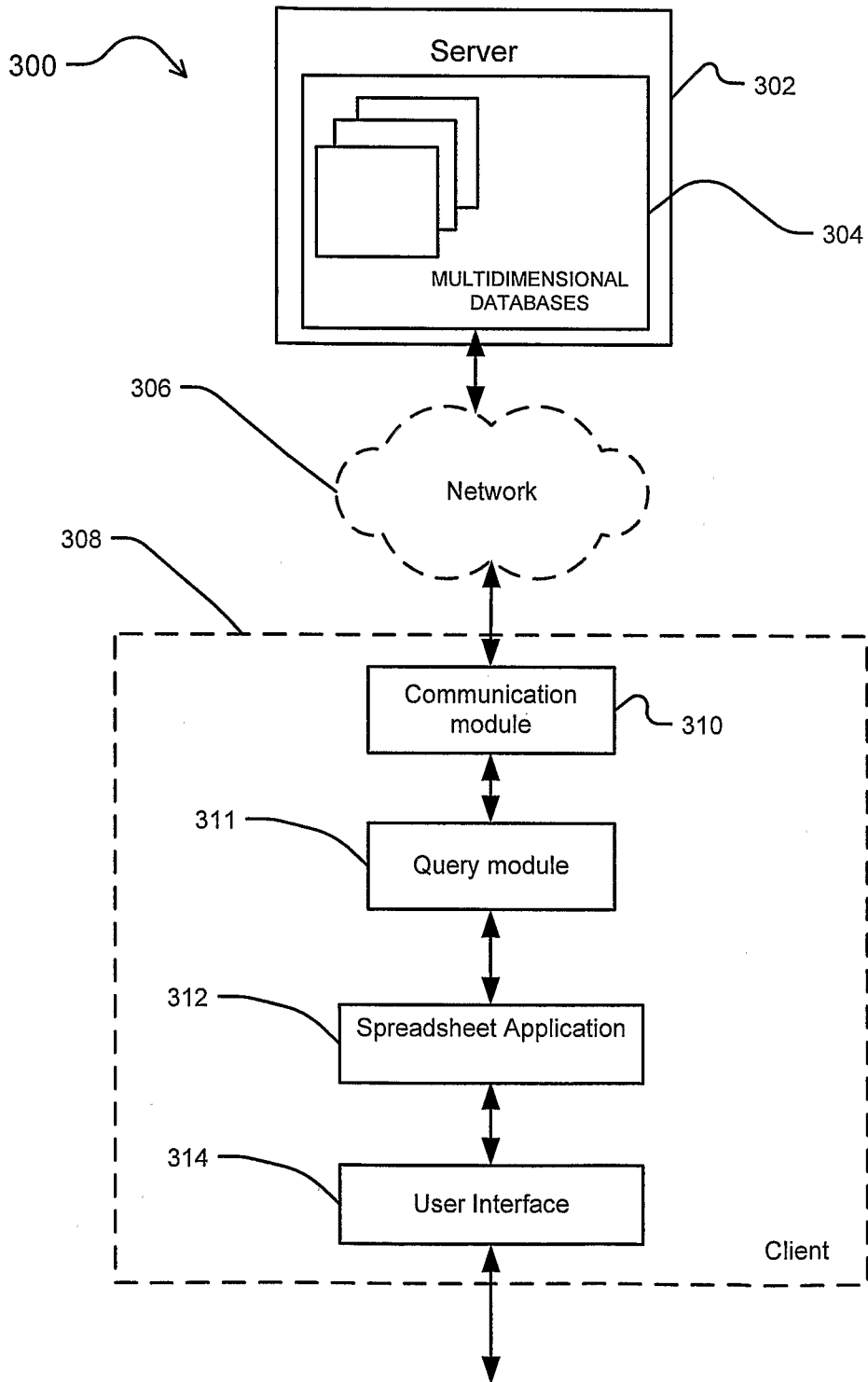


FIG. 3

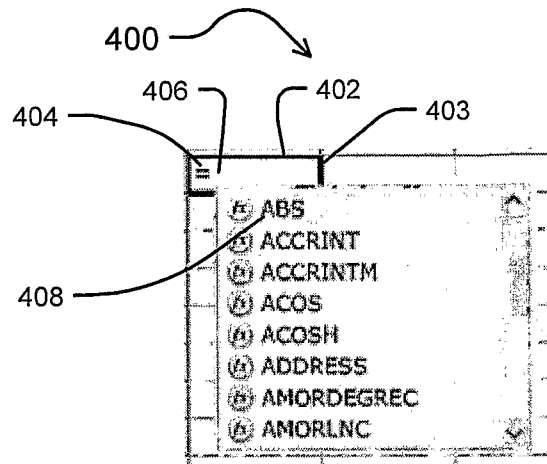


FIG. 4

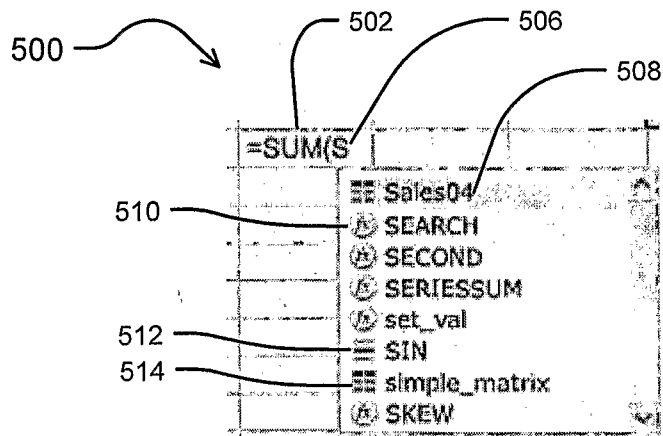


FIG. 5

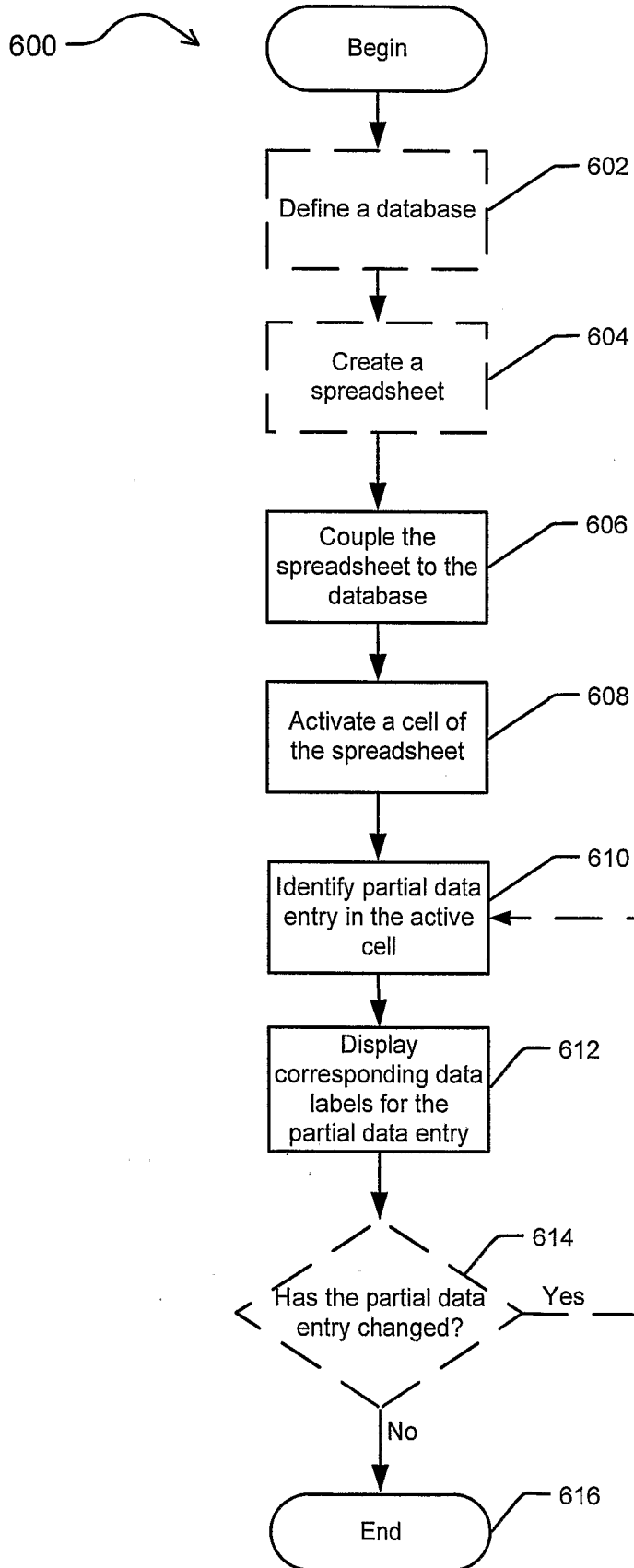


FIG. 6

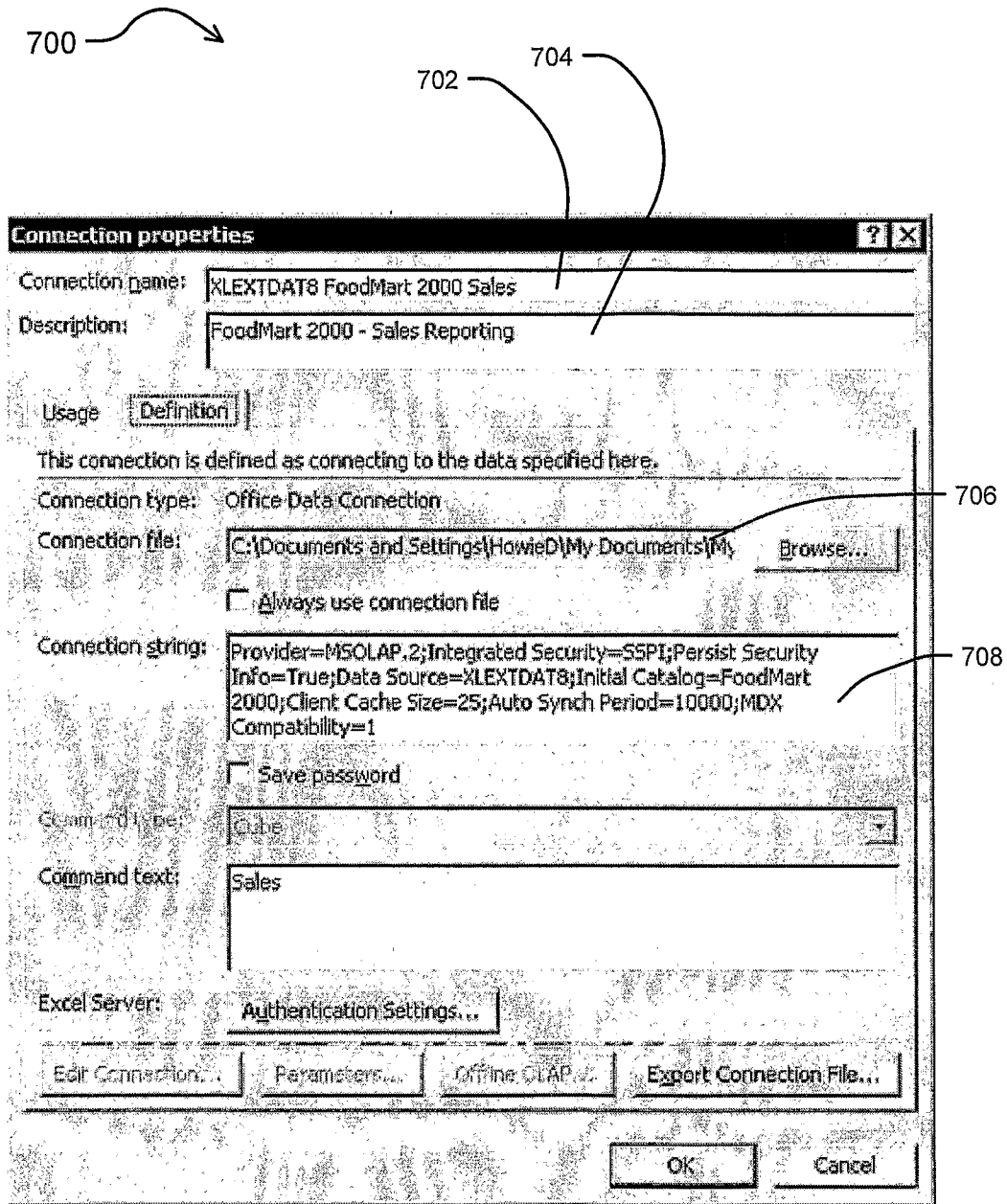




FIG. 7

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2006/034987

A. CLASSIFICATION OF SUBJECT MATTER		
<i>G06F 17/30(2006.01)i, G06F 17/00(2006.01)i</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) IPC 8 G06F 17/27, G06F 17/30, G06F 17/00		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched KR, JP : IPC as above		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) eKIPASS(KIPO Internal) "keyword : autocomplet*, spreadsheet, query and similar terms"		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,845,300 A (Comer et al.) 01 December 1998 see figures 4-5, 7-8; abstract; claims 1, 26.	1-20
Y	US 6,564,213 B1 (Ortega et al.) 13 May 2003 see abstract; figures 1-4.	1-20
A	US 6,377,965 B1 (Hachamovitch et al.) 23 April 2002 see column 1 line 10 - column 2 line 39.	1-20
A	KR 10-0289065 B1 (ELECTRONICS AND TELECOMMUNICATIONS RESEARCH INSTITUTE) 06 May 1999 see abstract; claims 1, 2.	1-20
P Y	US 2006/0129929 A1 (Weber et al.) 15 June 2006 see abstract; figures 5-8; claim 1.	1-20
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> 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 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 23 FEBRUARY 2007 (23.02.2007)		Date of mailing of the international search report 26 FEBRUARY 2007 (26.02.2007)
Name and mailing address of the ISA/KR  Korean Intellectual Property Office 920 Dunsan-dong, Seo-gu, Daejeon 302-701, Republic of Korea Facsimile No. 82-42-472-7140		Authorized officer SON, Young Tae Telephone No. 82-42-481-5748 

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2006/034987

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US05845300	01. 12. 1998	US39326E 1 US5845300A	03. 10. 2006 01. 12. 1998
US6564213BA	13. 05. 2003	AU2001249800A8 AU200149800A 1 AU200149800A5 EP01389322A2 EP1389322A2 US6564213B 1 W00180079A2 W0200180079A2 W0200180079A3	30. 10. 2001 30. 10. 2001 30. 10. 2001 18. 02. 2004 18. 02. 2004 13. 05. 2003 25. 10. 2001 25. 10. 2001 04. 12. 2003
US06377965	23. 04. 2002	US6377965BA	23. 04. 2002
KR100289065B1	14. 02. 2001	None	
US2006/0129929A1	15. 06. 2006	AU2005225040AA BR200505072A CA2525986AA CN1790259A EP01672527A2 EP1672527A2 JP2006172445A2 KR2006067813A MXPA05012292A US2006129929AA	29. 06. 2006 12. 09. 2006 15. 06. 2006 21. 06. 2006 21. 06. 2006 21. 06. 2006 29. 06. 2006 20. 06. 2006 19. 06. 2006 15. 06. 2006.