CUSTOMIZABLE PARAMETER USER INTERFACE

BACKGROUND

[0001] A database is an organized collection of data. A database management system (DBMS) is a computer program (or more typically, a suite of them) designed to manage a database and run operations on the data requested by numerous clients. A DBMS allows queries to be written and executed to retrieve the data. A DBMS may also allow creation of forms or reports. More refined queries may also be executed from the DBMS. These customized queries/reports/forms may require that certain parameters be provided before a query/report/form is executed against the database. Such customized queries/reports/forms typically involve several complicated steps and the result is typically difficult to understand or modify.

SUMMARY

[0002] 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 identify key features or essential features of the claimed subject matter, nor is it intended to be used as an aid in determining the scope of the claimed subject matter.

[0003] A customizable user interface is provided for entering requested parameters related to a database query. The customized parameter user interface provides parameter entry dialogs/windows in correlation to a data view (e.g., form or report) that is produced according to a database query. The parameters entered may provide for modification of the data view. Also, the manager of the database may structure data views of a database to automatically include prompts for parameters before results are returned by the database. These prompts may be customized by the manager and may be provided according to dialogs such as pop-ups, pull-down menus, fly-outs, or a variety of other user interface components. Accordingly, an intuitive, efficient and modifiable data view is provided.

DRAWINGS

[0004] Non-limiting and non-exhaustive embodiments of the present invention are described with reference to the following figures, wherein like reference numerals refer to like parts throughout the various views unless otherwise specified.

[0005] FIGURE 1 illustrates an exemplary computing architecture for a computer;

[0006] FIGURE 2 illustrates a functional block diagram of an exemplary system for providing a customized parameter user interface within a database management system (DBMS);

[0007] FIGURE 3 is an operational flow diagram of a process for providing a customized parameter entry form in association with a database query;

[0008] FIGURE 4 illustrates an operational flow diagram for adding a parameter to a customized parameter form;

[0009] FIGURE 5 illustrates an exemplary report before being modified by entry of an additional parameter to the query that generated the form;

[0010] FIGURE 6 illustrates the exemplary report of FIGURE 5 with a selection being made to filter the query results corresponding to the report;

[0011] FIGURE 7 illustrates an exemplary parameter generation form;

[0012] FIGURE 8 illustrates an exemplary parameter entry form that has not been customized;

[0013] FIGURE 9 illustrates an exemplary customized parameter entry form; and

[0014] FIGURE 10 illustrates the report of FIGURES 5 and 6 with the results modified according the entered parameters.

DETAILED DESCRIPTION

[0015] Embodiments are herein described more fully below with reference to the accompanying drawings, which form a part hereof, and which show specific examples for practicing the embodiments. However, embodiments may be implemented in many different forms and should not be construed as limited to the embodiments set forth herein; rather, these embodiments are provided so that this disclosure will be thorough and complete, and will fully convey the scope of the subject matter to those skilled in the art. Embodiments disclosed may be practiced as methods, systems or devices. Accordingly, embodiments disclosed may take the form of an entirely hardware implementation, an entirely software implementation or an implementation combining software and hardware aspects. The following detailed description is, therefore, not to be taken in a limiting sense.

[0016] When reading the discussion of the routines presented herein, it should be appreciated that the logical operations of various embodiments are 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 illustrated and making up the embodiments described herein are referred to variously as operations, structural devices, acts or modules. These operations, structural devices, acts and modules may be implemented in software, in firmware, in special purpose digital logic, and any combination thereof.

[0017] Referring now to the drawings, in which like numerals represent like elements. FIGURE 1 and the corresponding discussion are intended to provide a brief, general description of a suitable computing environment in which embodiments of the invention may be implemented.

[0018] Generally, program modules include routines, programs, components, data structures, and other types of structures that perform particular tasks or implement particular abstract data types. Other computer system configurations may also be used, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, minicomputers, mainframe computers, and the like. Distributed computing environments may also be used where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

[0019] Referring now to FIGURE 1, an exemplary computer architecture for a computing device 100 utilized in various embodiments will be described. For example, the computer may be configured as a personal computer, a mobile computer and the like. As shown, computing device 100 includes a central processing unit 102 ("CPU"), a system memory 104, including a random access memory 106 ("RAM") and a read-only memory ("ROM") 108, and a system bus 116 that couples the memory to the CPU 102. A basic input/output system containing the basic routines that help to transfer information between elements within the computer, such as during startup, is stored in the ROM 108. The computing device 100 further includes a mass storage device 120 for storing an operating system 122, application programs, and other program modules, which will be described in greater detail below.

[0020] The mass storage device 120 and its associated computer-readable media provide volatile and non-volatile storage for the computing device 100. The computer readable media may include any type of removable and/or non-removable media.

[0021] The computing device 100 operates in a networked environment using logical connections to remote computers through a network 112, such as the Internet. The computing device 100 may connect to the network 112 through a network interface unit 110 connected to the bus 116.

[0022] The computing device 100 may also include an input/output controller 114 for receiving and processing input from a number of devices, such as: a keyboard, mouse, electronic stylus and the like. Similarly, the input/output controller 114 may provide output to a display screen, a printer, or some other type of device (not shown).

[0023] As mentioned briefly above, a number of program modules and data files may be stored in the mass storage device 120 and RAM 106 of the computing device 100, including an operating system 122 suitable for controlling the operation of a networked computer. The mass storage device 120 and RAM 106 may also store one or more program modules. In particular, the mass storage device 120 and the RAM 106 may store a local component of a Database Management System (DBMS) 124.

[0024] The local component of the DBMS 124 includes functionality for presenting a customized parameter user interface for entering parameters in relation to a database search query. The customized parameter user interface provides parameter entry dialogs/windows in correlation to a data view (e.g., form or report) that is produced according to a database query. The parameters entered may allow for modification of the data view. The modification may occur without altering the original query statement. The modification may also occur by directly modifying the query statement. The persistence of the modification may be associated with the record source, a query object, and/or a parameter property. Also, the manager of the database may structure data views of a database to automatically include prompts for parameters before results are returned by the database. These prompts may be customized by the manager and may be provided according to dialogs such as pop-ups, pull-down menus, fly-outs, or a variety of other user interface components. Accordingly, an intuitive, efficient and modifiable data view is provided.

[0025] As used herein, "Control" generally refers to a child display frame an application uses in conjunction with another display frame to perform simple input and

output (I/O) tasks. Controls are most often used within dialog boxes, but may also be used in other display frames. Controls within dialog boxes provide the user with the means to type text, choose options, and direct a dialog box to complete its action. Controls in other display frames provide a variety of services, such as letting the user choose commands, view status, and view and edit text. One example of a control is a right-click selection menu that presents a child display frame of an application's user interface.

[0026] "Data view" generally refers to a form, report and/or table that represents a collection of data from a database. The collection of data from the database may be provided in the form or report according to a database search query. The data view may be modified by the entry of any parameters according to the customized parameter user interface.

[0027] As used herein below, "Parameter" generally refers to any data value or set of data values associated with the data included in a data view. Entered parameters may modify a particular data view when entered according to the customized parameter user interface.

[0028] "Parameter entry prompt" generally refers to a user interface dialog that is provided for entry of a parameter in association with a data view. The user interface dialog may take the form of a display frame, pop-up, pull-down menu, fly-out, or a variety of other user interface components. Entry of the parameter in the parameter entry prompt may be provided as a text box entry, a pull-down selection box, or other types of data entry user interface components.

[0029] "Parameter filter property" (PFilter) generally refers to a property that corresponds to an entered parameter associated with a data view. The parameter filter property is added to the query statement according to a Boolean AND operation to modify the results included in the data view.

[0030] "Parameter form property" (PForm) generally refers to a property that when activated indicates that a customized parameter form is associated with a particular data view.

[0031] "Record source" generally refers to a stored property corresponding to the query statement that represents the data view. When a parameter is entered in accordance with the present disclosure, the parameter may be added according to a Boolean AND operation to the record source.

[0032] "Query statement" corresponds to the statement used for a query to the database that produces the data view. In a structured query language (SQL) database, the query statement corresponds to the SQL statement passed to the database engine for retrieving the database results corresponding to the data view.

[0033] FIGURE 2 illustrates a functional block diagram of an exemplary system for providing a customized parameter user interface within a database management system (DBMS), in accordance with the present disclosure. DBMS interface 124 includes data view browser 202, parameter prompt module 204, customized parameter UI module 206, and property manager 208. Data view browser 202 further includes statement generator 210 and is coupled to database engine 230. Database engine 230 is coupled to database 240. Property manager 208 is coupled to storage 212. Storage 212 includes entries for a record source 214, a parameter filter property 216, and a parameter form property 220.

[0034] In operation, data view browser 202 communicates between database engine 230 and the other functional blocks of DBMS 124 to provide a data view (e.g., a form or report, see FIGURE 5) of data contained within database 240. Statement generator 210, included in data view browser 202, generates query statement 232 that is passed to database engine 230. If query statement 232 is executable in its present form, database engine 230 returns query results 236 to data view browser 202. If instead, query statement 232 is not executable because it requires identification of an additional parameter not included in query statement 232, then database engine 230 returns any necessary parameter requests to data view browser 202.

[0035] Data view browser 202 communicates any parameter requests 234 to parameter prompt module 204. Parameter prompt module 204 consults with property manager 208 to determine if a customized parameter form is associated with current data view. Property manager 208 accesses a list of properties (e.g., 214-220) in storage 212 to make the determination. Storage 212 may be a temporary or permanent storage and may correspond to system memory 104 or mass storage device 120 shown in FIGURE 1.

[0036] If the properties (e.g., 214-220) in mass storage device indicate that a customized parameter form is associated with the data view, functionality provided by customized parameter UI module 206 is used to display the customized parameter form in association with the data view. If however, the properties (e.g., 214-220) indicate that no customized parameter form is associated with the data view, a general parameter form is

provided by parameter prompt module 204 to solicit entry of the requested parameter from the user.

[0037] The functional blocks described in FIGURE 2 are not an exhaustive representation of the functional blocks possibly associated with a DBMS. Additional or fewer functional blocks may be included and the DBMS still operates for its intended purpose. For example, in an alternative embodiment, it may be that customized parameter UI module 206 communicates directly with data view browser 202 rather than communicating through parameter prompt module 204. In another alternative embodiment, a decision may be made by a database manager that every form or report to be produced for a particular database requires a customized parameter form. In this embodiment, parameter prompt module 204 and certain of the properties (e.g., 214-220) of storage 212 may be unnecessary since each data view will have an associated customized parameter form.

[0038] FIGURE 3 is an operational flow diagram of a process for providing customized parameter entry form in association with a database query, in accordance with the present disclosure. Processing starts when a query has been generated for a database that does not include all the parameters necessary for execution of the query such that parameters are required to be entered. Processing continues with operation 302.

[0039] At operation 302, an object/view is being opened (e.g., a form or report) that returns the response to the query in a readily understandable format. While the report is being opened, processing continues with decision operation 304.

[0040] At decision operation 304, a determination is made whether a PFilter property is specified for the query. The PFilter property indicates a WHERE clause in the query, such that an additional filtering of the query is being requested. If no PFilter property is being specified, processing advances to operation 308. However, if a PFilter property is specified, processing continues to operation 306.

[0041] At operation 306, a PFilter clause corresponding to the specificed PFilter property is added to the query statement. For example, if the PFilter property is one that indicates an alphabet limitation of results beginning with the letter "A", and the original query was for current customers, then a clause is added such that the resulting query is for customers that start with the letter "A". Processing continues at operation 308.

[0042] At operation 308, the query statement is executed. While the query statement is executed, processing continues with decision operation 310.

[0043] At decision operation 310, a determination is made whether execution of the query statement requires a parameter for completion. If no parameter is required, then processing advances to operation 322. However, if a parameter is required to be entered, then processing moves to decision operation 312.

[0044] At decision operation 312, a determination is made whether a form has been specified already for use for requesting the entry of the parameter. In one embodiment, a parameter form is specified when indicated by a PForm property. If no form is specified, processing moves to operation 314.

[0045] At operation 314, a prompt is provided for entering the parameter that does not include the customizations available in a parameter form generated according to the present disclosure. Once the prompt is displayed, processing continues with decision operation 316.

[0046] If however, a parameter form is specified, processing continues with operation 318. At operation 318, the specified form is displayed for entry of the parameter. Once the specified form is displayed, processing continues with decision operation 316.

[0047] At decision operation 316, a determination is made whether the user has selected to cancel the query. The query can be canceled at any time at the user's selection. If the user has canceled the query, processing moves to operation 318. At operation 318, the opening of the object/view of the data is canceled, and processing ends.

[0048] If however, the query has not been canceled, processing moves to operation 320. At operation 320, the execution of the statement, modified with the entries of the parameters, is executed. The modification may occur without altering the original query statement. The modification may also occur by directly modifying the query statement. The persistence of the modification may be associated with the record source, a query object, and/or a parameter property. Processing continues at operation 322.

[0049] At operation 322, the results of the query are bound to the object/view and the results are shown as modified by the entered parameters. Once the object/view is displayed, process 300 ends, and processing move on to other tasks.

[0050] FIGURE 4 illustrates an operational flow diagram for adding a parameter to a customized parameter form, in accordance with the present disclosure. Processing starts where a developer desires to add a parameter to a customized parameter form. Processing continues with operation 402.

[0051] At operation 402, a command is received to add a parameter to a customized parameter form. The command may come from a pull down menu selection or other edit selection by the developer of the customized form. Processing continues with decision operation 404.

[0052] At decision operation 404, a determination is made whether the PFilter property already contains the parameter being added. If the PFilter property does not contain the parameter, processing moves to operation 406 where the parameter is added to PFilter property. If the PFilter property was already populated with the parameter or once the PFilter property is populated with the parameter, processing moves to decision operation 408.

[0053] At decision operation 408, a determination is made whether a PForm property indicates that a parameter form has been already designated for inclusion of the property. If no parameter form has already been generated, processing moves to operation 410 where a parameter form is created and the PForm property is set.

[0054] If the parameter form for inclusion of the parameter was already created, or once the parameter form is created, processing continues with operation decision operation 412.

[0055] At decision operation 412, a determination is made whether a control has been generated on the form for entry of the parameter. If no control is present, processing moves to operation 414 where a process is commenced for generating a control on the parameter form for entry of the parameter.

[0056] If a control was already generated, or once the control is generated, processing continues to operation 416. At operation 416, the object/view of the data with the added parameter may be displayed after execution of the query statement. Once the object/view is displayed, process 400 ends, and processing moves to other tasks.

[0057] FIGURE 5 illustrates an exemplary report before being modified by entry of an additional parameter to the query that generated the form, in accordance with the present disclosure. Report 500 includes a listing of sales by year for a product, the product numbers, when the products shipped and the sales amounts.

[0058] FIGURE 6 illustrates the exemplary report of FIGURE 5 with a selection being made to filter the query results corresponding to the report, in accordance with the present disclosure. The selection being made in report 600 is to filter the results according

to date filters that indicate to limit the result to those occurring between two dates specified.

[0059] FIGURE 7 illustrates an exemplary parameter generation form, in accordance with the present disclosure. Parameter generation form 700 allows the developer of the parameter form to select to customize the parameter prompts for entry of the parameters. Selecting to create the parameters places the prompt in an edit mode for designing the prompts and control for the parameters.

[0060] FIGURE 8 illustrates an exemplary parameter entry form that has not been customized according to the present disclosure. Parameter entry form 800, has no specified information for the report to which the prompt is associated. In contrast, FIGURE 9 illustrates an exemplary customized parameter entry form, in accordance with the present disclosure. Customized parameter entry form 900 shows marked difference from the parameter entry form of FIGURE 8. Customized parameter entry form 900 includes information that illustrates its related to the form for which it has been generated. For example, the prompt for the form, instead of merely describing a start and end date, illustrates an "oldest ship date" and "newest ship date". The title of the form has also changed. The customized parameter entry form 900 may include additional logic for populating a drop down display frame with distinct values. As an example, depending on authoring needs, default values may be obtained from a database setting or calculated for a particular year. The customizations of the form, allow for more seamless user experience when entering parameters for search queries of a database.

[0061] FIGURE 10 illustrates the report of FIGURES 5 and 6 with the results modified according the entered parameters, in accordance with the present disclosure. Report 1000 illustrates a display frame having ship dates for the query results as modified by ship dates entered into a customized parameter form.

[0062] The above specification, examples and data provide a complete description of the manufacture and use of the composition of the invention. Since many embodiments of the invention can be made without departing from the spirit and scope of the invention, the invention resides in the claims hereinafter appended.

WHAT IS CLAIMED IS:

1. A computer-implemented method for providing a user interface for entering parameters associated with a database query statement, the method comprising:

generating a database query statement (210) for retrieving data from a database that is displayed within a data view; wherein the data view comprises a view of the data retrieved from the database;

sending the query statement (232) to a database engine for execution;

the database engine determining that the query statement is not executable as received, and in response, returning a parameter request that requests entry of at least one additional parameter;

accessing a property list (212) to determine when a customized parameter form exists to receive entry of the at least one additional parameter, wherein the customized parameter form is a custom designed form that is created for entry of the at least one additional parameter; in response to determining that the customized parameter form does not exist, displaying a general parameter form and a control generated on the form for entry of the at least one additional parameter and displaying the general parameter form (206) to receive entry of the at least one additional parameter when the property list indicates that a customized parameter form does not exist to receive the at least one additional parameter and is not associated with the data view. in response to determining that the customized parameter form exists, displaying the customized parameter form to receive entry of the at least one additional parameter when the property list indicates that the customized parameter form exists for entry of the at least one additional parameter; wherein the customized parameter form includes values obtained from the database relating to the data view, wherein the values obtained from the database are used as default values to receive the at least one additional parameter and wherein prompts of the customized parameter form are modified based on the values; and

executing the query statement with a modified value of the parameter.

2. The computer-implemented method of claim 1, wherein the data view (302) includes at least one of: a form, a report, and a table.

3. The computer-implemented method of claim 1, wherein the property list (212) includes a record source property (214) that represents the query statement that indicates the data view.

4. The computer-implemented method of claim 1, wherein the property list (212) includes a parameter filter property (216) for adding a parameter to the query statement according to a Boolean operation to modify results associated with the data view.

5. The computer-implemented method of claim 1, further comprising modifying the data view (302) by altering the query statement.

6. The computer-implemented method of claim 1, wherein the property list (212) includes a parameter form property (220) that indicates that a customized parameter form is associated with the data view.

7. The computer-implemented method of claim 1, wherein the values obtained from the database relating to the data view included on the customized parameter form are static values.

8. The computer-implemented method of claim 1, wherein the customized parameter form (206) includes text blocks (900) specific to the data view.

9. The computer-implemented method of claim 1, wherein the customized parameter form (206) is configured to provide at least one of: an additional property, an additional filter property, and a control.

10. The computer-implemented method of claim 1, further comprising modifying the data view (302) in association with a parameter of the parameter form, wherein the data view (302) is modified without altering the query statement.

11. A computer-readable medium having computer executable instructions for entering parameters associated with a database query statement comprising:

determining whether execution of a query statement requires a parameter for completion before the query statement is executable by a database engine; wherein the query statement requests a data view (310) to be displayed that comprises a collection of data from a database; wherein the parameter modifies results generated by executing the query statement such that the data view to be displayed is modified;

accessing a property list (212) when the query statement requires a parameter to complete the query statement;

determining whether a customized property formexists for receiving the parameter from a parameter form property (220) of the property list (212); wherein the customized parameter form is already specified and uses values obtained from the database to customize prompts and entry areas for receiving the parameter;

displaying the customized property form (206) to receive entry of the parameter when it is determined that the customized parameter form exists; and

when it is determined that the customized parameter form does not exist; displaying a general parameter form to receive entry of the parameter, wherein the general parameter form comprises parameter prompts for entry of the at least one additional parameter that are not customized.

12. The computer-readable medium of claim 11, wherein the property list (212) includes a record source property (214) that represents the query statement that indicates the data view.

13. The computer-readable medium of claim 11, wherein the property list (212) includes a parameter filter property (216) for adding a parameter to the query statement according to a Boolean operation to modify results associated with the data view.

14. The computer-readable medium of claim 11, further comprising modifying the data view (302) by altering the query statement.

15. The computer-readable medium of claim 11, wherein the customized parameter form (206) is configured to provide at least one of: an additional property, an additional filter property, and a control.

16. The computer-readable medium of claim 11, further comprising modifying the data view (322) in association with a parameter of the parameter form, wherein the data view is modified without altering the query statement.

17. A computer-implemented method for correlating a customized property form to a data view, the method comprising:

receiving a parameter request (324) in response to a query statement; wherein the parameter request is received when the query statement requires a parameter before the query statement is executable; wherein the query statement requests a data view that comprises a collection of data from a database; wherein the parameter modifies results generated by executing the query statement and causes the data view to be changed;

accessing a property manager (208) to determine whether a property list (212) indicates that a customized property form (206) exists to receive entry of the parameter;

displaying the customized property form (206) to receive entry of the parameter when the customized property form exists; wherein the customized parameter form is created and uses values obtained from the database to customize prompts and entry areas for receiving the parameter;

when the customized property form does not exist, displaying a standard property form to receive entry of the parameter, wherein the standard property form is not pre-defined and customized for entry of the parameter;

associating a parameter with at least one of: the customized property form (206) and the standard property form (204); and

modifying the data view (322) in association with the parameter

18. The computer-implemented method of claim 17, wherein the property list (212) includes a parameter filter property (216) for adding a parameter to the

query statement according to a Boolean operation to modify results associated with the data view.

19. The computer-implemented method of claim 17, further comprising modifying the data view (302) without altering the query statement.

20. The computer-implemented method of claim 17, wherein the property list (212) includes a parameter form property (220) that indicates that a customized parameter form is associated with the data view.