(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)
(19) World Intellectual Property Organization
(43) International Publication Date
11 April 2013 (11.04.2013)
(51) International Patent Classification:
G06F 17/21 (2006.01) G06F 3/14 (2006.01)
(21) International Application Number:
PCT/US20 12/058667
(22) International Filing Date:
4 October 2012 (04.10.2012)
(25) Filing Language:
English
(26) Publication Language:
English
(30) Priority Data:
13/252,450 4 October 2011 (04.10.2011) US
(54) Title: AUTOMATIC RELATIONSHIP DETECTION FOR REPORTING ON SPREADSHEET DATA
(57) Abstract: A system for creating a report in a spreadsheet includes: a central processing unit; and a memory encoding instructions that, when executed by the central processing unit, cause the central processing unit to create: a data source module programmed to identify tables associated with the spreadsheet; a relationship module programmed to analyze the tables and identify relationships between the items; and a display module programmed to display the relationships between the tables.
Declarations under Rule 4.17:

— as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(H))

— as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(Hb))

Published:

— without international search report and to be republished upon receipt of that report (Rule 48.2(g))

— as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(Hb))
AUTOMATIC RELATIONSHIP DETECTION FOR REPORTING ON SPREADSHEET DATA

BACKGROUND

[0001] Data summary tables in spreadsheets can be used to analyze large amounts of data. A data summary table provides an efficient way to display and summarize data that is supplied by a database program or that is in a data listing of a spreadsheet. A user can select fields of the data to include within filter regions, row, column, or data regions of the data summary table and can choose aggregation functions such as the sum, variance, count, and standard deviation to be displayed for selected data fields. Data in a database that can be queried from within a spreadsheet program, or spreadsheet data including lists, can be analyzed in a data summary table.

[0002] With spreadsheets that include a significant amount of data, it can be difficult for the user to select the desired data upon which to be included in such reports as data summary tables. For example, the volume of data provided across one or more largely unstructured spreadsheets can overwhelm the user, making it difficult to find the pieces of data relevant for a specific reporting scenario.

SUMMARY

[0003] In one aspect, a system for creating a report in a spreadsheet includes: a central processing unit; and a memory encoding instructions that, when executed by the central processing unit, cause the central processing unit to create: a data source module programmed to identify tables associated with the spreadsheet; a relationship module programmed to analyze the tables and identify relationships between the items; and a display module programmed to display the relationships between the tables.

[0004] In another aspect, a method for identifying relationships between data items in a spreadsheet, the method comprising: identifying, by a computing device, tables associated with the spreadsheet; allowing a user to add one of the tables to a data summary table in the spreadsheet; analyzing the added table to determine a relationship between the added table and other tables in the data summary table; and when the added table is unrelated to the other tables in the data summary table, automatically identifying a relationship between the added table and the other tables in the data summary table.

[0005] In yet another aspect, a method for identifying relationships between dimensions in a data summary table of a spreadsheet, the method comprising: allowing a user to add a table to the data summary table in the spreadsheet, the table including a data
region defined in the spreadsheet; analyzing columns associated with the table to determine a relationship between the dimensions and other tables in the data summary table; when the added table is unrelated to other tables in the data summary table: notifying a user of a need to define the relationship when the added table is added to the data summary table; and using heuristics to automatically identify the relationship between the table and the other tables in the data summary table; displaying the relationship between the added table and the other tables in the data summary table.

[0006] 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 or essential features of the claimed subject matter, nor is it intended to be used in any way to limit the scope of the claimed subject matter.

**DESCRIPTION OF THE DRAWINGS**

[0007] Figure 1 shows an example system for generating a report in a spreadsheet program.

[0008] Figure 2 shows an example pivot table in a spreadsheet.

[0009] Figure 3 shows example data regions used to generate a pivot table in a spreadsheet.

[0010] Figure 4 shows example modules of a client computing device of the system of Figure 1.

[0011] Figure 5 shows an example method for identifying a relationship between two or more tables in a report.

[0012] Figure 6 shows an example wizard to manually identify a relationship between two or more tables in a report.

[0013] Figure 7 shows an interface to identify a relationship between data from an external data store and one or more tables in a report.

[0014] Figure 8 shows example components of computing devices shown in Figure 1.

**DETAILED DESCRIPTION**

[0015] The present application is directed to systems and methods for automatically detecting and suggesting relationships between data regions in a spreadsheet application. In examples described herein, relationships between data regions are identified and presented to the user to allow the user to more easily combine data into a single usable report.

[0016] Figure 1 shows an example system 100 that supports the generation of one or more reports based upon data provided in a spreadsheet application. The system 100
includes a client computer 102 and a server computer 106. The client computer 102 is connected to the server computer 106 across a network, typically a corporate Intranet or the Internet. More or fewer client computers, server computers or other computing devices may be used.

As described further below, the client computer 102 can connect to the server computer 106 to obtain data, such as a spreadsheet file. The client computer 102 can execute a native a spreadsheet application on the client computer 102 to manipulate the spreadsheet file. In another example, the client computer 102 can utilize one or more applications hosted on the server 106 to access and manipulate the spreadsheet file. Other configurations are possible.

Referring now to Figure 2, an example program 200 is shown. In this example, the program 200 is Microsoft's EXCEL spreadsheet software program running on a computer system, such as the computer system 100 described above. The program 200 includes a spreadsheet with an example list of data.

A user can create one or more reports based on the data within the spreadsheet. In some examples described herein, the report can be a data summary table. However, in other examples, the report can simply be another "regular" table. For example, the systems and methods described herein can be used to suggest certain columns for a table created by the user based on associations identified with other tables in the spreadsheet.

For example, an example user interface 302 of program 200 is shown. The user interface 302 includes an initial data summary table 320 (The data summary table 320 is blank in Figure 3). The data summary table 320 can be created from data from various sources. In one example as shown in Figure 3, the data summary table 320 can be created from data provided within the spreadsheet file or from one or more external data stores.

The user interface 302 of the program 200 also includes an example task pane 450 that can be used to create and modify the data summary table 320. For example, the task pane 450 includes a list of the data fields from data associated with the spreadsheet file. The user can select and deselect fields from the task pane 450 to create the data summary table 320.

The task pane 450 generally includes a field pane 452 and a layout pane 454. The field pane 452 includes a list of the data fields available in the spreadsheet. In these examples, the data fields include tables, with each table having one or more columns that can be added to the report. When a data field in the list is added to the layout pane 454 as described below, the checkbox associated with the data field is checked.
The layout pane 454 includes a plurality of zones that represent aspects of the data summary table 320 that is created using the task pane 450. For example, the layout pane 454 includes a row zone, a column zone, a value zone, and a filter zone. The row zone defines the row labels for the resulting data summary table 320. The column zone defines the column labels for data summary table 320. The value zone identifies the data that is summarized (e.g., aggregation, variation, etc.) on data summary table 320. The Filter zone allows for the selection of filtering that is applied to all other fields in the other zones.

One or more of the data fields from field pane 452 are added to one or more of the zones of layout pane 454 to create and modify the data summary table 320. In the example shown, the user can click, drag, and drop a data field from the list in the field pane 452 to one of the zones of the layout pane 454 to add a field to the data summary table 320.

Referring now to Figure 3, the data used to create a report, such as the data summary table 320, can come from multiple sources. The sources of data can be internal to a spreadsheet, as well as external.

For example, a spreadsheet 500 shown in Figure 3 includes two internal data regions 502 and 504. Data regions are defined sets of data having at least two dimensions, such as rows and columns. For example, the data region 502 includes columns EmployeeID, Name, and rows 1-3. Similarly, the data region 504 includes columns EmployeeID, Sales Amount, and rows 1-7.

In these examples, a plurality of data regions can be included at multiple locations within a spreadsheet, such as on same or different sheets within the spreadsheet. As noted below, data can also be incorporated from external sources.

In this example, the program 200 is programmed to analyze the data regions 502 and 504 and to automatically determine relationships between the regions. For example, the program determines that the EmployeeID data in both regions are compatible using various heuristics described further below.

In these examples, a relationship is a link created between two data regions (sometimes referred to as "tables") based on matching data values. These relationships can include a one-to-one relationship and/or a one-to-many relationship. For example, a single value in a table can correspond to multiple values in a look-up or related table. Relationships allow the program to look-up and summarize data in both tables simultaneously.
Upon this detection, the program 200 creates a relationship between the two data regions 502, 504 and allows the user to easily create a report by simply selecting the fields in a user interface. For example, the data fields associated with the data regions 502, 504 can be clustered in the user interface so that the user can easily identify the relationships between the data regions. Additional details about example user interfaces for this purpose are found in U.S. Patent Application Serial No. 13/252,412, Attorney Docket No. 333789.02 (14917.1922US01), titled "Automatic Scoping Of Data Entities," filed on even date herewith, the entirety of which is hereby incorporated by reference.

One example of a report that can be generated after the relationship between the data regions 502, 504 is detected is a report 510 on the spreadsheet 500. In this example, the report 510 is a data summary table, such as a pivot table, generated using the data from the data regions 502, 504. To create the report 510, the user selects desired data fields from the data regions 502, 504 and the external data store 506 in a manner similar to that described in Figure 2.

In addition to the data regions 502, 504 provided in the spreadsheet 500, data can also be incorporated from external data stores into the spreadsheet 500. For example, an external data store 506 is shown in Figure 3. Examples of such external data stores include OLAP and non-OLAP data sources. Data from this external data store 506 is incorporated into the spreadsheet 500. Based on information about this external data store 506, relationships between the data in the external data store 506 and data regions 502, 504 within the spreadsheet 500 can be identified, as described further below.

In some examples, the number of available data fields provided in the data regions and external data stores from which the user can select becomes voluminous. In such scenarios, it can be difficult for the user to review the available data fields and select the desired fields for inclusion in a report.

Referring now to Figure 4, logical modules of the client computer 102 are shown. These modules include a data source module 522, a relationship module 524, and an interface module 526. In these examples, the modules 522, 524, 526 are executed in conjunction with the spreadsheet application that allows the user to manipulate data within spreadsheet files. The modules 522, 524, 526 of the client computer 102 function to identify relationships between the data in the spreadsheet so that the user can more easily identify relevant fields that the user may want to select when creating a report.

The data source module 522 identifies all data sources associated with a particular file, such as a spreadsheet file. This can include internal data regions defined on
one or more sheets of the spreadsheet file. This can also include data from external data stores. Such external data stores can include data that is referenced from within the spreadsheet, such as external data that has been incorporated into the spreadsheet file. The data source module 522 provides all of the data sources to the relationship module 524 for analysis.

[0036] The relationship module 524 determines which, if any, of the data sources within a spreadsheet are related. For example, if the user is creating a report, such as a pivot table, the relationship module 524 determines uses various heuristics to analyze all of the data regions and external data stores that have been incorporated into the spreadsheet to determine any relationships therebetween.

[0037] For example, the program uses heuristics to examine the data, either data in the spreadsheet grid or data from any external data source which the user has imported into the spreadsheet application, and based on that suggest related data entities in different data regions. The spreadsheet application then stores information on these relationships and other spreadsheet features will now automatically leverage these relationships, allowing the user to create a report which includes data from multiple different data regions.

[0038] Some examples of the heuristics that can be used to identify associations include one or more of the following:

- column name matching (e.g., columns with similar captions);
- value matching, such as one column with unique values in the primary key table and a column with repeated values in the foreign key table but where the distinct set of values exist in the primary key column;
- data type matching (e.g., date or currency fields);
- fuzzy value matching (e.g., "CA" matching "California"); and
- using mapping tables to produce needed identifiers in one table to create a relationship to another table which already has a column with those identifiers.

[0039] For example, referring to Figure 5, in one scenario, the relationship between two or more tables is created when the user adds data fields from unrelated tables to a report, such as a data summary table.

[0040] In this example, a method 530 begins when the user adds a field to the report at operation 532. Next, at operation 534, a determination is made regarding whether or not a relationship is needed because, for example, the field is unrelated to the fields already
included in the table. If a relationship is not needed (i.e., the field is already related in some manner), the field is added and the method ends.

[0041] However, if a relationship is needed, control is instead passed to operation 536, and the user is notified that a relationship may be missing. One example method for such a notification is providing an info-bar within the user interface to notify the user that the relationship is missing (e.g., "Relationships between tables may need to be created.").

[0042] Next, at operation 538, a determination is made regarding whether or not the user wants to create a relationship for the newly-added field. If not, the method ends, and user may be prompted again for the need to provide a relationship the next time the user modifies the report. If so, control is instead passed to operation 539. At operation 539, the relationship between the newly-added field and the fields already included in the report is determined. In one example, the relationship can be automatically determined by using one or more of the heuristics described above. In another example, the relationship can be manually defined.

[0043] For example, referring now to Figure 6, an example relationship wizard 540 is illustrated. In this wizard 540, the user can manually define the relationship between fields in different tables. In this example, the field Employee ID is related for tables SalesTable and EmployeeTable.

[0044] The relationships defined by the relationship module 524, whether automatically created by the program or defined manually by the user, can be persisted throughout the creation of the report. In some examples, the relationships can also be persisted through creation of other reports within the spreadsheet so that the program and/or user do not have to redefine the relationships each time the data fields in the spreadsheet are used.

[0045] Referring again to Figure 4, the interface module 526 presents the available data fields from the data regions and external data stores in the spreadsheet to the user for inclusion in the report. In this example, the interface module 526 indicates relationship between the data sources to the user in the interface. These relationships can be manifest in several manners. In one example, when the user selects a table, the interface can present only that table and any other tables that are related to the selected table. In another example, the tables can be clustered to identify the relationships.

[0046] As noted above, in addition to defining relationships among data fields from tables within the spreadsheet, the program can define relationships with data from external sources, such as external data stores. In this scenario, the relationship module 524 is
programmed to analyze the data from an external data store and provide any relationships between that data and any other data contained in the spreadsheet.

[0047] For example referring to Figure 7, an example data connection wizard 550 is shown. In this example, the data connection wizard 550 allows the user to select tables which are directly related to the currently selected table(s) in the wizard. In this example, the external data store "Adventure WorksDW2008R2" is selected in the wizard 550, and the user can select specific tables within the spreadsheet to relate to the database (e.g., DimEmployee is selected). This selection can be done manually, by the user in the wizard 550, or it can be done automatically by the program using the heuristics described above.

In other examples, if the external data source has relationships already defined between its tables, those can be imported along with the data from the external data source and used in the spreadsheet application along with any other relationships created there.

[0048] For example, the program can access schema information about the external data store that allows the program to ascertain relationships between tables. This schema (e.g., in the form of an XML file) can relate fields within the external data store that is imported, as well as other data in other external data stores. The program leverages this information to relate data fields automatically for the user.

[0049] With reference to Figure 8, example components of the client computer 102 and the server computer 106 are shown. In example embodiments, the client computer 102 and the server computer 106 are computing devices. The client computer 102 and server computer 106 can include input/output devices, a central processing unit ("CPU"), a data storage device, and a network device. The client computer 102 can also be a mobile computing device, such as a laptop, tablet, convertible, or other handheld device like a smartphone or cellular telephone. The server computer 106 can also be incorporated as one or more server computers in a network of servers. In such a scenario, the network can provide a cloud computing platform in which one or more applications and data are hosted for clients connecting to the cloud computing platform. The discussion below regarding the server computer 106 also applies to the components of the client computer 102.

[0050] In a basic configuration, the server computer 106 typically includes at least one processing unit 602 and system memory 604. Depending on the exact configuration and type of computing device, the system memory 604 may be volatile (such as RAM), non-volatile (such as ROM, flash memory, etc.) or some combination of the two. System memory 604 typically includes an operating system 606 suitable for controlling the operation of a server, such as the Microsoft SharePoint® Server 2010 collaboration server,
from Microsoft Corporation of Redmond, Washington. The system memory 604 may also include one or more software applications 608 and may include program data.

[0051] The server computer 106 may have additional features or functionality. For example, server computer 106 may also include computer readable media. Computer readable media can include both computer readable storage media and communication media.

[0052] Computer readable storage media is physical media, such as data storage devices (removable and/or non-removable) including magnetic disks, optical disks, or tape. Such additional storage is illustrated in Figure 6 by removable storage 610 and non-removable storage 612. Computer readable storage media may include 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 readable storage media can include, 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 server computer 106. Any such computer readable storage media may be part of server computer 106. Server computer 106 may also have input device(s) 614 such as keyboard, mouse, pen, voice input device, touch input device, etc. Output device(s) 616 such as a display, speakers, printer, etc. may also be included.

[0053] Consistent with embodiments of the present disclosure, the input device(s) 614 may comprise any motion detection device capable of detecting the movement or gesture of a user. For example, the input device(s) 614 may comprise a Kinect® motion capture device, from Microsoft Corporation, comprising a plurality of cameras and a plurality of microphones.

[0054] The server computer 106 may also contain communication connections 618 that allow the device to communicate with other computing devices 620, such as over a network in a distributed computing environment, for example, an intranet or the Internet. Communication connections 618 are one example of communication media.

Communication media may typically be embodied by 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.

[0055] 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 6 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 various application functionality all of which are integrated (or "burned") onto the chip substrate as a single integrated circuit. When operating via an SOC, the functionality, described above, with respect to the present disclosure may be operated via application-specific logic integrated with other components of the computing devices 102 and 106 on the single integrated circuit (chip).

[0056] The various embodiments described above are provided by way of illustration only and should not be construed to limiting. Various modifications and changes that may be made to the embodiments described above without departing from the true spirit and scope of the disclosure.
CLAIMS

What is claimed is:

1. A system for creating a report in a spreadsheet, the system comprising:
   a central processing unit; and
   a memory encoding instructions that, when executed by the central processing unit
   (602), cause the central processing unit to create:
      a data source module programmed to identify tables associated with the
      spreadsheet;
      a relationship module programmed to analyze the tables and identify
      relationships between the items; and
      a display module programmed to display the relationships between the
      tables.

2. The system of claim 1, wherein the relationship module analyzes and identifies the
   relationships when a table is added to the report.

3. The system of claim 1, wherein the report is a data summary table, and the tables
   each include one or more dimensions to be added to the data summary table.

4. The system of claim 1, wherein one of the tables is a data region defined in the
   spreadsheet.

5. A method for identifying relationships between data items in a spreadsheet, the
   method comprising:
      identifying, by a computing device, tables associated with the spreadsheet;
      allowing a user to add one of the tables to a data summary table in the spreadsheet;
      analyzing the added table to determine a relationship between the added table and
      other tables in the data summary table; and
      when the added table is unrelated to the other tables in the data summary table,
      automatically identifying a relationship between the added table and the other tables in the
      data summary table.

6. The method of claim 5, further comprising displaying the relationship between the
   added table and the other tables in the data summary table.
7. The method of claim 5, further comprising notifying a user of a need to define the relationship when the added table is added to the data summary table.

8. The method of claim 5, further comprising using heuristics to automatically identify the relationship.

9. The method of claim 5, further comprising persisting the relationship throughout use of the tables in the spreadsheet.

10. A method for identifying relationships between dimensions in a data summary table of a spreadsheet, the method comprising:

    allowing a user to add a table to the data summary table in the spreadsheet, the table including a data region defined in the spreadsheet;

    analyzing columns associated with the table to determine a relationship between the dimensions and other tables in the data summary table;

    when the added table is unrelated to other tables in the data summary table:

        notifying a user of a need to define the relationship when the added table is added to the data summary table; and

        using heuristics to automatically identify the relationship between the table and the other tables in the data summary table; and

    displaying the relationship between the added table and the other tables in the data summary table.
Figure 2
Figure 4
530
Field is added to report

534
Relationship needed?

YES
Notify user of needed relationship

NO
End

536

538
Create relationship?

YES

539
Create relationship

NO

Create Relationship

Select the tables and columns to use for the relationship

Table: SalesTable

Column: Employee ID

Related Table: EmployeeTable

Column: Employee ID

Manage Relationships...
Figure 7