METHODS AND APPARATUS FOR PROCESSING A DATABASE QUERY

Inventors: Eric Lawrence Barsness, Pine Island, MN (US); John Matthew Santosuosso, Rochester, MN (US)

Correspondence Address:
IBM CORPORATION, INTELLECTUAL PROPERTY LAW
DEPT 917
3605 HIGHWAY 52 NORTH
ROCHESTER, MN 55901-7829 (US)

Assignee: INTERNATIONAL BUSINESS MACHINES CORPORATION, ARMONK, NY

Appl. No.: 10/965,184

Filed: Oct. 14, 2004

Publication Classification

Int. Cl. G06F 17/30 (2006.01)

U.S. Cl. .......................................................... 707/3

ABSTRACT

In a first aspect, a method is provided that includes the steps of (1) pre-computing a query result for each of a plurality of whole segments of data included in a database; (2) receiving a query specifying a defined range of data in the database; (3) determining if any of the whole segments are within the defined range; (4) performing the query on any partial segments of data within the defined range; and (5) determining the result of the query based on the pre-computed query results for any whole segments determined to be within the defined range and the result of the query on any partial segments within the defined range. Numerous other aspects are provided.
FIG. 1
<table>
<thead>
<tr>
<th></th>
<th>Date</th>
<th>Revenue</th>
<th>Annual Total</th>
<th>Annual Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Q1 2000</td>
<td>1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Q2 2000</td>
<td>1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Q3 2000</td>
<td>1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Q4 2000</td>
<td>1000</td>
<td>4000</td>
<td>1000</td>
</tr>
<tr>
<td>6</td>
<td>Q1 2001</td>
<td>2000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Q2 2001</td>
<td>2000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Q3 2001</td>
<td>2000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Q4 2001</td>
<td>2000</td>
<td>8000</td>
<td>2000</td>
</tr>
<tr>
<td>10</td>
<td>Q1 2002</td>
<td>3000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Q2 2002</td>
<td>3000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Q3 2002</td>
<td>3000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Q4 2002</td>
<td>3000</td>
<td>12000</td>
<td>3000</td>
</tr>
<tr>
<td>14</td>
<td>Q1 2003</td>
<td>4000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Q2 2003</td>
<td>4000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Q3 2003</td>
<td>4000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Q4 2003</td>
<td>4000</td>
<td>16000</td>
<td>4000</td>
</tr>
<tr>
<td>18</td>
<td>Q1 2004</td>
<td>5000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Q2 2004</td>
<td>5000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>Q3 2004</td>
<td>5000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 2**
Pre-Compute a Query Result for Each of a Plurality of Whole Segments of Data Included in a Database

Receive a Query Specifying a Defined Range of Data in the Database

Determine if Any Whole Segments Are Within the Defined Range

Perform the Query on Any Partial Segments of Data Within the Defined Range

Determine the Result of the Query Based on the Pre-Computed Query Results for Any Whole Segments Within the Defined Range and the Result of the Query on Any Partial Segments Within the Defined Range

FIG. 3
Start

Receive a Request to Perform a Query on a First Portion of Data in a Database

Has a Result of the Query been Pre-Computed for at Least One Second Portion of Data in the Database, Wherein the Second Portion is Included in the First Portion?

Yes

Determine the Result of the Query on the First Portion of the Database is Based on the Pre-Computed Result

Retrieve the Pre-Computed Result

Perform the Query on a Third Portion of Data in the Database, Wherein the Third Portion is a Portion of the First Portion Other Than the Second Portion

Compute the Result of the Query on the First Portion Based on the Pre-Computed Result and the Result of the Query on the Third Portion

End

FIG. 4
METHODS AND APPARATUS FOR PROCESSING A DATABASE QUERY

FIELD OF THE INVENTION

[0001] The present invention relates generally to computer systems, and more particularly to methods and apparatus for processing a database query.

BACKGROUND

[0002] A database frequently receives a query for information, such as a sum, average, or the like, from a range of data in a database. Conventionally, in response to such query, the database accesses every record, each of which includes one or more data entries, in the range of data to process the query. However, such a method for processing a database query results in large processing overhead.

[0003] Accordingly, methods and apparatus for improved processing of a database query are desired.

SUMMARY OF THE INVENTION

[0004] In a first aspect of the invention, a first method is provided. The first method includes the steps of (1) precomputing a query result for each of a plurality of whole segments of data included in a database; (2) receiving a query specifying a defined range of data in the database; (3) determining if any of the whole segments are within the defined range; (4) performing the query on any partial segments of data within the defined range; and (5) determining the result of the query based on the pre-computed query results for any whole segments determined to be within the defined range and the result of the query on any partial segments within the defined range.

[0005] In a second aspect of the invention, a method of processing a database query for information is provided. The method includes the steps of (1) receiving a request to perform a query on a first portion of data in a database; (2) determining the result of the query on the first portion of the database is based on at least one pre-computed result, wherein the pre-computed result is a result of the query on a second portion of data in the database that is included in the first portion; (3) performing the query on a third portion of data in the database, wherein the third portion is a portion of the first portion other than the second portion; and (4) computing the result of the query on the first portion based on the pre-computed result and the result of the query on the third portion. Numerous other aspects are provided, as are methods, systems, apparatus and computer program products in accordance with these other aspects of the invention. Each computer program product described herein may be carried by a medium readable by a computer (e.g., a carrier wave signal, a floppy disc, a compact disc, a DVD, a hard drive, a random access memory, etc.).

[0006] Other features and aspects of the present invention will become more fully apparent from the following detailed description, the appended claims and the accompanying drawings.

BRIEF DESCRIPTION OF THE FIGURES

[0007] FIG. 1 is block diagram of an apparatus for processing a database query in accordance with an embodiment of the present invention.

[0008] FIG. 2 is a block diagram of an example of a database that may be used in an apparatus for processing a database query in accordance with an embodiment of the present invention.

[0009] FIG. 3 is a first exemplary method for processing a database query in accordance with an embodiment of the present invention.

[0010] FIG. 4 is a second exemplary method for processing a database query in accordance with an embodiment of the present invention.

DETAILED DESCRIPTION

[0011] The present invention provides methods and apparatus for efficiently processing a database query. More specifically, the present methods and apparatus allow efficient access to a large portion of a range (e.g., the entire range) specified by the database query. Consequently, the present methods and apparatus provide performance gains for processing database queries compared to conventional database query processing.

[0012] FIG. 1 is block diagram of an apparatus for processing a database query in accordance with an embodiment of the present invention. With reference to FIG. 1, the apparatus 100 includes a processor 102 for processing a database query. The processor 102 is coupled to a storage device 104, such as a disk drive or the like, which includes a database 106. Details of the database 106 are described below with reference to FIG. 2. The processor 104 is adapted to execute code, which is adapted to efficiently query the database 108. More specifically, the processor 104 is adapted to execute code adapted to perform a first and/or second exemplary method in accordance with an embodiment of the present invention. Details of the first and second exemplary methods are described below with reference to FIGS. 3 and 4, respectively. Additionally, the apparatus 100 may include a communication (e.g., a port) for communicating with other apparatuses.

[0013] FIG. 2 is a block diagram of an example of a database that may be used in an apparatus for processing a database query in accordance with an embodiment of the present invention. With reference to FIG. 2, the database 108 may store data relevant to, for example, a specific customer or application. Such data may be organized in a manner specified by a user, such as a database administrator. For example, the data in the database 108 may include a first column 200 representing dates. More specifically, the first column 200 includes a data entry (e.g., in a row corresponding to the column) for each quarter from the first quarter of year 2000 to the third quarter of year 2004. The database 108 includes a second column 202 of data representing revenue. For example, the second column 202 of data includes a data entry, corresponding to each quarter in the first column 200, representing revenue.

[0014] Further, the database 108 includes data based on other data in the database 108. More specifically, the database 108 may include a third column 204 of data, each entry of which represents a total revenue for a year (e.g., the sum of the revenue data for each quarter of the year). Similarly, the database 108 may include a fourth column 208 of data, each entry of which represents average
quarterly revenue for a year (e.g., an average of the revenue data for the quarters of the year). For example, each data entry in the third 206 and/or fourth column 208 is based on data entries in the second column 204 that correspond to quarters of a particular year from the first column 202. As described further below, data in the database 108 may be grouped together to form one or more (e.g., a plurality) of whole segments 210-216. Although data in the database 108 of FIG. 2 includes four whole segments 210-216, data in the database 108 may include a larger or smaller number of whole segments. Further, a whole segment may include a larger or smaller amount of data than the whole segments 210-216 illustrated in FIG. 2.

[0015] The structure of the database 108 (e.g., arrangement of data in rows and columns) described above is exemplary. The database 108 may be structured in a different manner. For example, the database 108 may be a relational database, flat-file database, object database, distributed database, or the like. Further, the description of data populating the database 108 is exemplary. The database 108 may include different data.

[0016] The operation of the apparatus 100 for processing a database query is now described with reference to FIG. 2 and with reference to FIG. 3, which illustrates a first exemplary method 300 for processing a database query in accordance with an embodiment of the present invention. With reference to FIG. 3, in step 302, the method 300 begins. In step 304, a query result for each of a plurality of whole segments of data included in a database is pre-computed. As stated, the database 108 may be grouped into one or more (e.g., a plurality) of whole segments. For example, data entries of the second column 204 of the database 108 that correspond to quarters of a particular year (e.g., 2000, 2001, 2002 or 2003) of the first column 202 may be grouped into respective whole segments 210-216 (shown in phantom). Therefore, the example database 108 includes four whole segments. A query for information, such as a sum, average, or the like, from the database (e.g., from a range of data in the database) are frequently received. Conventionally, in response to such query, the database accesses every record, which includes one or more entries, in the range of data to process the query. However, in step 304, a query for information, such as a sum, average, or the like, is performed on each whole segment 210-216 in the database 108 (e.g., in advance of computing a result for the query for information from the range of data). In this manner, the result of the query on each of the plurality of whole segments 210-216 is computed before receiving and/or computing the result of a query for information from a range of data in the database (e.g., pre-computed). Such result may be stored as an entry in the database 108 (e.g., in the third 206 or fourth column 208 of the database 108). While this example shows additional columns explicitly added to the database table for use in storing pre-computed values, a hybrid index could be used instead. The hybrid index would store the pre-computed values as metadata, and queries using the index would retrieve the metadata for whole segments from the index rather than directly from the table.

[0017] For example, the sum of data in each whole segment 210-216, which represents total revenue for years 2000, 2001, 2002 and 2003, respectively, is computed. Additionally or alternatively, the average of data in each whole segment 210-216, which represents average quarterly revenue for years 2000, 2001, 2002 and 2003, respectively, is computed. Although, the third column 206 of the database 108 stores annual data and the fourth column 208 of average quarterly data, the hybrid index may be configured to store data representing other periods of time (e.g., daily, monthly, etc.).

[0018] In step 306, a query specifying a defined range of data in the database is received. More specifically, as stated above, a query for information such as a sum, average, or the like, from a range of data in the database may be received by the database. For example, the database may be queried for total revenue from the third quarter of year 2000 to the second quarter of year 2003.

[0019] In step 308, it is determined if any whole segments are within the defined range. For example, the apparatus 100 determines that two whole segments 212-214 are within the predefined range. Because the result of such query for each whole segment 210-216 in the database 108 is pre-computed, by determining whether any whole segments 210-216 are within the defined range, the apparatus 100 may avoid having to access every record in the defined range to process the query. More specifically, the apparatus 100 may not need to access every record corresponding to such whole segments 212-214 (e.g., in response to the query) to process the query.

[0020] In step 310, the query is performed on any partial segments of data within the defined range. More specifically, the apparatus 100 may determine the defined range includes one or more partial segments (e.g., parts of a whole segment) of data. In the example above, the apparatus 100 determines the defined range includes a part (e.g., revenue for the last two quarters of year 2000) of the whole segment 210 that represents revenue for year 2000 and includes a part (e.g., revenue for the first two quarters of year 2003) of the whole segment 216 that represents the revenue for year 2003. The apparatus 100 performs the query on a range defined by each such partial segment. For example, the apparatus 100 may determine total revenue for the third and fourth quarter of year 2000 by accessing the entries in such range. Similarly, the apparatus 100 may determine total revenue for the first and second quarter of year 2003 by accessing the entries in such range. In this manner, the result of the query on any partial segments of data within the defined range is determined.

[0021] In step 312, the result of the query specifying the defined range is determined based on the pre-computed query results for any whole segments determined to be within the defined range and the result of the queries on any partial segments within the defined range. More specifically, the pre-computed query result for each whole segment 212-214 determined to be within the defined range is accessed. For example, entries in the third column 206 corresponding to the total revenue could be used instead. The hybrid index would retrieve the pre-computed values as metadata, and queries using the index would retrieve the metadata for partial segments from the index rather than directly from the table.

[0022] The operation of the apparatus 100 for processing a database query is now described with reference to FIG. 2
and with reference to FIG. 4, which illustrates a second exemplary method 400 for processing a database query in accordance with an embodiment of the present invention. With reference to FIG. 4, in step 402, the method 400 begins. In step 404, a request to perform a query on a first portion of data in a database is received. More specifically, the apparatus 100 may query a defined range of the database 108 for information. The defined range serves as the first portion. For example, the apparatus 100 may query the database for total revenue from the third quarter of year 2000 to the second quarter of year 2003, in which case revenue entries from the third quarter of year 2000 to the second quarter of year 2003 are included in the first portion.

In step 406, it is determined whether a result of the query has been pre-computed for at least one second portion of data in the database, wherein the second portion is included in the first portion. A second portion may be, for example, quarterly revenue data for year 2001 and/or quarterly revenue data for year 2002, which is entirely included in the first portion of data. The second portion may have a pre-computed result associated with it. Therefore, in step 406, the apparatus 100 may determine whether a total revenue for year 2001 (e.g., a sum of the quarterly revenues for year 2001) and/or quarterly revenue data for year 2002 has been pre-computed. If it is determined in step 406 that a result of the query has not been pre-computed for at least one second portion of data in the database, step 408 is performed. Alternatively, if it is determined a result of the query has been pre-computed for at least one second portion of data in the database, step 410 is performed.

In step 408, the query is performed on the second portion and the result of such query is stored as a pre-computed result. For example, similar to step 304 of method 300, the apparatus may query a second portion of data, which is entirely included in the first portion of data, for information before arriving at the result of the query on the first portion of data. For example, the apparatus 100 may pre-compute total revenue for year 2001 by querying the database for total revenue from the first quarter of year 2001 to the fourth quarter of year 2001, in which case revenue entries from the first quarter of year 2001 to the fourth quarter of year 2001 are included in the second portion, and store the result of the query as a pre-computed result in a third column 206 of the database 108. Similarly, the apparatus 100 may pre-compute total revenue for year 2002 by querying the database for total revenue from the first quarter of year 2002 to the fourth quarter of year 2002, in which case revenue entries from the first quarter of year 2002 to the fourth quarter of year 2002 are included in the second portion, and store the result of the query as stored as needed (e.g., only in response to a request to perform a query on the first portion of data), thereby avoiding unnecessary processing. Thereafter, step 410 is performed.

In step 410, it is determined that the result of the query on the first portion of the database is based on at least one pre-computed result. Upon receiving a request to perform the query on the first portion of data in the database, the apparatus 100 may determine that pre-computed results may be employed to determine the query result.

In step 412, the pre-computed result is retrieved. For example, the apparatus may access the third column 206 of the database 108 and retrieve the value of the total revenue for year 2001 and/or year 2002.

In step 414, the query is performed on a third portion of data in the database, wherein the third portion is a portion of the first data other than the second portion. For example, the apparatus 100 may query the database for total revenue for the third and fourth quarters of year 2000 and the first and second quarters of year 2003, in which case revenue entries from the third and fourth quarters of year 2000 and the first and second quarters of year 2003 are included in the third portion.

In step 416, the result of the query on the first portion is computed based on the at least one pre-computed result and the result of the query on the third portion of data. For example, the at least one pre-computed result is added to the result of the query on the third portion of data to yield the result of the query on the first portion of data in the database. Thereafter, step 418 is performed. In step 418, the method 400 ends.

The foregoing description discloses only exemplary embodiments of the invention. Modifications of the above disclosed apparatus and methods which fall within the scope of the invention will be readily apparent to those of ordinary skill in the art. For example, although in one or more embodiments above, data in the database was grouped into segments or portions by date, the data in the database may be grouped into segments or portions based on another parameter associated with the data. Further, in one or more embodiments, data in the whole segments or second portion, and therefore, results based on such data, which is stored in a hybrid index, may be monitored to determine how frequently such data is accessed. Data used infrequently may be discarded. In some embodments, an algorithm may be used to determine how frequently such data is determined. Based upon the monitoring, the apparatus may determine whether to continue storing the results based upon such data. In this manner, unused data may not unnecessarily occupy database space. Further, a threshold age may be employed such that only pre-computed results performed on a recent date range of data (e.g., defined by the threshold age, such as data stored within the last year, month, week, etc.) are stored. For example, if the database includes a long history of data but only data within a recent date range is accessed, according to the present methods and apparatus, only pre-computed results performed on such date range are stored, thereby saving database space. Pre-computed results performed on older date ranges may be removed.

Accordingly, while the present invention has been disclosed in connection with exemplary embodiments thereof, it should be understood that other embodiments may fall within the spirit and scope of the invention, as defined by the following claims.
1. A method, comprising:
pre-computing a query result for each of a plurality of whole segments of data included in a database;
receiving a query specifying a defined range of data in the database;
determining if any of the whole segments are within the defined range;
performing the query on any partial segments of data within the defined range; and
determining the result of the query based on the pre-computed query results for any whole segments determined to be within the defined range and the result of the query on any partial segments within the defined range.

2. The method of claim 1 wherein determining the result of the query based on the pre-computed query results for any whole segments determined to be within the defined range and the result of the query on any partial segments within the defined range includes determining the result of the query by retrieving the pre-computed query results and determining the result of the query on any partial segments of data within the defined range.

3. The method of claim 1 further comprising:
   storing the pre-computed query results in the database; and
   removing pre-computed query results performed on a date range of data older than a specified age from the database.

4. The method of claim 1 wherein data in the database is grouped into one or more whole segments and partial segments based on a date associated with the data.

5. The method of claim 1 wherein pre-computing the query result for each of a plurality of whole segments of data included in a database includes pre-computing the query result for each of a plurality of whole segments of data included in a database in response to receiving the query specifying the defined range of data in the database.

6. A method of processing a database query for information, comprising:
   receiving a request to perform a query on a first portion of data in a database;
   determining the result of the query on the first portion of the database is based on at least one pre-computed result, wherein the pre-computed result is a result of the query on a second portion of data in the database that is included in the first portion;
   performing the query on a third portion of data in the database, wherein the third portion is a portion of the first portion other than the second portion; and
   computing the result of the query on the first portion based on the pre-computed result and the result of the query on the third portion.

7. The method of claim 6 wherein computing the result of the query on the first portion based on the pre-computed result and the result of the query on the third portion includes merging the pre-computed result and the result of the query on the third portion of data.

8. The method of claim 6 wherein the second portion of data is larger than the third portion of data.

9. The method of claim 6 wherein computing the result of the query on the first portion based on the pre-computed result and the result of the query on the third portion includes computing the result of the query on the first portion by retrieving the pre-computed result and computing the result of the query on the third portion.

10. The method of claim 6 further comprising:
    storing the pre-computed query results in the database; and
    removing pre-computed query results performed on a date range of data older than a specified age from the database.

11. The method of claim 6 wherein the data in the database is grouped into the first, second and third portion of data based on a date associated with the data.

12. The method of claim 11 wherein at least one of a day, month and year is associated with the second portion of data.

13. The method of claim 6 wherein the pre-computed result is computed in response to receiving the request to perform a query on the first portion of data in the database.

14. The method of claim 6 further comprising:
    monitoring the pre-computed result; and
    determining whether to continue to store the pre-computed result based on the monitoring.

15. An apparatus, comprising:
    a database for storing data;
    a processor coupled to the database and adapted to execute code adapted to:
    pre-compute a query result for each of a plurality of whole segments of data included in the database;
    receive a query specifying a defined range of data in the database;
    determine if any of the whole segments are within the defined range;
    perform the query on any partial segments of data within the defined range; and
    determine the result of the query based on the pre-computed query results for any whole segments determined to be within the defined range and the result of the query on any partial segments within the defined range.

16. The apparatus of claim 15 wherein the code is further adapted to determine the result of the query by retrieving the pre-computed query results and determining the result of the query on any partial segments of data within the defined range.

17. The apparatus of claim 15 wherein the code is further adapted to:
    store the pre-computed query results in the database; and
    remove pre-computed query results performed on a date range of data older than a specified age from the database.

18. The apparatus of claim 15 wherein data in the database is grouped into one or more whole segments and partial segments based on a date associated with the data.

19. The apparatus of claim 15 wherein the code is further adapted to pre-compute a query result for each of a plurality
of whole segments of data included in a database in response to receiving the query specifying the defined range of data in the database.

20. An apparatus for processing a database query, comprising:

- a database for storing data;
- a processor coupled to the database and adapted to execute code adapted to:
  receive a request to perform a query on a first portion of data in the database;
  determine the result of the query on the first portion of the database is based on a pre-computed result, wherein the pre-computed result is a result of the query on a second portion of data in the database that is included in the first portion;
  perform the query on a third portion of data in the database, wherein the third portion is a portion of the first portion other than the second portion; and
  compute the result of the query on the first portion based on the pre-computed result and the result of the query on the third portion.

21. The apparatus of claim 20 wherein the code is further adapted to merge the pre-computed result and the result of the query on the third portion.

22. The apparatus of claim 20 wherein the second portion of data is larger than the third portion of data.

23. The apparatus of claim 20 wherein the code is further adapted to compute the result of the query on the first portion by retrieving the pre-computed result and computing the result of the query on the third portion.

24. The apparatus of claim 20 wherein the code is further adapted to:

- store the pre-computed query results in the database; and
- remove pre-computed query results performed on a date range of data older than a specified age from the database.

25. The apparatus of claim 20 wherein the data in the database is grouped into the first, second and third portion of data based on a date associated with the data.

26. The apparatus of claim 25 wherein at least one of a day, month and year is associated with the second portion of data.

27. The apparatus of claim 20 wherein the code is further adapted to compute the pre-computed result in response to the request to perform a query on the first portion of data in the database.

28. The apparatus of claim 20 wherein the code is further adapted to:

- monitor the pre-computed result; and
- determine whether to continue to store the pre-computed result based on the monitoring.

29. A computer program product, comprising:

- a medium readable by a computer, the computer readable medium having computer program code adapted to:
  pre-compute a query result for each of a plurality of whole segments of data included in a database;
  receive a query specifying a defined range of data in the database;
  determine if any of the whole segments are within the defined range;
  perform the query on any partial segments of data within the defined range; and
  determine the result of the query based on the pre-computed query results for any whole segments determined to be within the defined range and the result of the query on any partial segments within the defined range.

30. The computer program product of claim 29 wherein the computer program code is further adapted to determine the result of the query by retrieving the pre-computed query results and determining the result of the query on any partial segments of data within the defined range.

31. The computer program product of claim 29 wherein the computer program code is further adapted to:

- store the pre-computed query results in the database; and
- remove pre-computed query results performed on a date range of data older than a specified age from the database.

32. The computer program product of claim 29 wherein data in the database is grouped into one or more whole segments and partial segments based on a date associated with the data.

33. The computer program product of claim 29 wherein the computer program code is further adapted to compute the pre-computed query result in response to receiving the query specifying the defined range of data in the database.

34. A computer program product, comprising:

- a medium readable by a computer, the computer readable medium having computer program code adapted to:
  receive a request to perform a query on a first portion of data in a database;
  determine the result of the query on the first portion of the database is based on a pre-computed result, wherein the pre-computed result is a result of the query on a second portion of data in the database that is included in the first portion;
  perform the query on a third portion of data in the database, wherein the third portion is a portion of the first portion other than the second portion; and
  compute the result of the query on the first portion based on the pre-computed result and the result of the query on the third portion.

35. The computer program product of claim 34 wherein the computer program code is further adapted to merge the pre-computed result and the result of the query on the third portion of data.

36. The computer program product of claim 34 wherein the second portion of data is larger than the third portion of data.

37. The computer program product of claim 34 wherein the computer program code is further adapted to compute the result of the query on the first portion by retrieving the pre-computed result and computing the result of the query on the third portion.

38. The computer program product of claim 34 wherein the computer program code is further adapted to:

- store the pre-computed query results in the database; and
remove pre-computed query results performed on a date range of data older than a specified age from the database.

39. The computer program product of claim 34 wherein the data in the database is grouped into the first, second and third portion of data based on a date associated with the data.

40. The computer program product of claim 39 wherein at least one of a day, month and year is associated with the second portion of data.

41. The computer program product of claim 39 wherein the computer program code is further adapted to compute the pre-computed result in response to receiving the request to perform a query on the first portion of data in the database.

42. The computer program product of claim 39 wherein the computer program code is further adapted to:

- monitor the pre-computed result; and
- determine whether to continue to store the pre-computed result based on the monitoring.