

METHOD AND DEVICE FOR EXECUTING A DATABASE QUERY

TECHNICAL FIELD

[0001] Various aspects relate to computer-implemented method and device for
5 executing a (real-time) database query.

BACKGROUND

[0002] The following discussion of the background to the invention is intended to facilitate an understanding of the present invention. However, it should be appreciated that the discussion is not an acknowledgment or admission that any of the material referred to
10 was published, known or part of the common general knowledge in any jurisdiction as at the priority date of the application.

[0003] In various scenarios, a device (e.g., a user device) may retrieve data from a (e.g., online) database. For this, the device may send a database query to the database requesting the data. It may be desired to retrieve the requested data as fast as possible (e.g., to increase
15 user convenience and/or for further processing of the requested data). As an illustrative example, a user may request a time series of a currency foreign exchange (FX) rate for a certain time period up to a current point in point and wants to retrieve updates in real-time. This may be limited by the time required to update the database with incoming new real-time data points, by the (query execution) time required to select data points associated with
20 the requested data (e.g., the requested time series) from the updated database, and by the time required to determine the requested data (e.g., the requested time series) based on the selected data points.

[0004] Hence, a need exists to provide an improved method and device for executing a real-time database query.

SUMMARY

[0005] Various aspects of the present disclosure relate to a computer-implemented method for executing a (real-time) database query. The method executed by at least one processor may include: a first table of a database receiving real-time data, the first table including a plurality of prior data points of a data set, wherein the real-time data represent one or more new data points of the data set, wherein the first table receiving the real-time data triggers a materialized view of the database to receive the one or more new data points; the materialized view determining one or more new aggregated data points by aggregating all data points of the one or more new data points which are associated with a same time interval of consecutive time intervals of equal length; the materialized view inserting the aggregated data points into a second table, the second table including a plurality of prior aggregated data points, wherein each of the plurality of prior aggregated data points is associated with a corresponding time interval of the consecutive time intervals and is an aggregation of all data points of the plurality of prior data points which are associated with the corresponding time interval; in the case that at least one new aggregated data point of the one or more new aggregated data points and a prior aggregated data point of the plurality of prior aggregated data points are associated with a same time interval of the consecutive time intervals, aggregating the at least one new aggregated data point and the prior aggregated data point within the second table; and in response to the database query requesting query data from the database, retrieving the query data from the second table.

[0006] Various aspects of the present disclosure relate to a device for executing a (real-time) database query, the device may include at least one processors configured to: implement a database including a first table, a materialized view, and a second table, wherein the first table includes a plurality of prior data points of a data set, wherein the second table includes a plurality of prior aggregated data points, wherein each of the

plurality of prior aggregated data points is associated with a corresponding time interval of consecutive time intervals of equal length and is an aggregation of all data points of the plurality of prior data points which are associated with the corresponding time interval; receive real-time data representing one or more new data points of the data set; insert the
5 one or more new data points into the first table, wherein the materialized view is configured to receive the one or more new data points in response to inserting the one or more new data points into the first table, to determine one or more new aggregated data points by aggregating all data points of the one or more new data points which are associated with a same time interval of the consecutive time intervals, and to insert the one or more new
10 aggregated data points into the second table; wherein the second table is configured to, in the case that at least one new aggregated data point of the one or more new aggregated data points and a prior aggregated data point of the plurality of prior aggregated data points are associated with a same time interval of the consecutive time intervals, aggregate the at least one new aggregated data point and the prior aggregated data point within the second table;
15 receive a database query requesting query data from the database; and in response to the database query, retrieve the query data from the second table.

[0007] Generating such a second table in which all data points which are associated with a same time interval are aggregated to one aggregated data point and selecting, in response to a database query, the information from the second table allows to reduce the query
20 executing time significantly. In an exemplary scenario, each prior data point may correspond to a tick of an asset price or a foreign exchange (FX) rate and a respective aggregated data point may be an aggregation of all prior data points for a corresponding time interval of one minute. In the case that a downstream application sends a database query to retrieve information for charting a one-minute-chart of the asset price or FX rate, (when using, for
25 example, the Clickhouse® database management system) selecting the information from the

first table may take a query execution time of about 3573 milliseconds (ms), whereas selecting the information from the second table may take a query execution time of about 80 ms. Thus, the present disclosure allows to increase the query execution time speed by about 44 times. Hence, various aspects relate to an approach to optimize database queries, such as time series queries.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] Throughout the drawings, it should be noted that like reference numbers are used to depict the same or similar elements, features, and structures. The drawings are not necessarily to scale, emphasis instead generally being placed upon illustrating aspects of the disclosure. In the following description, some aspects of the disclosure are described with reference to the following drawings, in which:

FIG 1. is a block diagram showing an example system according to various aspects of the present disclosure;

FIG 2. is a block diagram showing a database device for executing a database query according to various aspects of the present disclosure;

FIG 3. is a block diagram showing an example electronic device according to various aspects of the present disclosure;

FIG. 4 is a flow chart showing a method for executing a real-time database query according various aspects of the present disclosure.

DETAILED DESCRIPTION

[0009] The following detailed description refers to the accompanying drawings that show, by way of illustration, specific details and aspects in which the disclosure may be practiced. One or more aspects are described in sufficient detail to enable those skilled in the art to practice the disclosure. Other aspects may be utilized and structural, logical, and/or

electrical changes may be made without departing from the scope of the disclosure. The various aspects of the disclosure are not necessarily mutually exclusive, as some aspects can be combined with one or more other aspects to form new aspects or embodiments. Various aspects are described in connection with methods and various aspects are described in
5 connection with devices. However, it may be understood that aspects described in connection with methods may similarly apply to the devices, and vice versa.

[0010] Various aspects relate to a database device which provides a significantly reduced database query execution time. In the following, various aspects are described for the exemplary scenario of foreign exchange (FX) rates and the charting of a time series
10 thereof. It is understood that this serves as an example for illustrating the present invention and that the principles can be applied accordingly to any other scenario including, for example, other (real-time) data and/or another use of the data.

[0011] **FIG.1** shows a system 10 according to various aspects. The system 10 may include one or more servers 100, a database device (short: device) 200, and an electronic
15 device 300. The one or more servers 100 may be, for example, part of a server cluster. The one or more servers 300 may be configured to provide (e.g., transmit) real-time data 102 (e.g., as a data stream) to the database device 200. Illustratively, the one or more servers 100 may be configured to stream real-time data 102 to the database device 200.

[0012] The database device 200 may be configured to receive and process the real-time
20 data 102 as detailed with reference to FIG.2. The electronic device 300 may be configured to transmit a database query 302 to the database device 200 to request data from the database device 200. The database device 200 may be configured to determine the requested data in response to receiving the database query 302 and to provide (e.g., transmit) the requested data 202 to the electronic device 300.

[0013] FIG.2 shows the database device 200 according to various aspects. The database device 200 may include at least one (e.g., one or more than one) processor 204. The at least one processor 204 may be configured to implement a database 206.

[0014] According to various aspects, the database device 200 may include a memory device (short: memory). The at least one processor 204 may employ the memory for implementing the database 206. The memory device may store instructions for implementing the database 206. The memory device may store the (implemented) database 206. The at least one processor 204 may be configured to employ a database management system (DBMS) for implementing the database 206. The DBMS may, for example, be column-oriented DMBS, such as Clickhouse®. In the case of a column-oriented DBMS, each table of the database 206 may be a column-oriented table.

[0015] The database 206 may include a first table 208. The first table 208 may be a source table. The first table 208 may include a plurality of prior data points of a data set. The term “prior” may indicate that the data points are not associated with a current time interval, but a past time period. The current time interval may be given by a predefined period of time in the past of the present time and may be associated with real-time. The past time period may be elapsed prior to the predefined period of time. Data points which are acquired within the predefined period of time prior to the present time may be referred to as real-time data points. Data points which are not yet in stored in a table and which are ingested into the table may be referred to as new data points. Hence, a new data point may be a new entry to the database 206. According to various aspects, new data points detailed herein may be real-time data points.

[0016] The database 206 may include a second table 210. The second table 210 may be a destination table. The second table 210 may include a plurality of prior aggregated data points. Each prior aggregated data point may be associated with a corresponding time

interval of consecutive time intervals of equal length. As an illustrative example, the length of each time interval of the consecutive time intervals may be one minute and the second table 210 may include one prior aggregated data point for each one-minute time interval. The length of each time interval of the consecutive time intervals may be such that the first
5 table 208 includes (e.g., on average) two or more (e.g., a plurality of) prior data points within each time interval. Each prior aggregated data point may be an aggregation of all prior data points of the plurality of prior data points which are associated with the corresponding time interval. For example, a specific (e.g., one-minute) time interval may include one prior aggregated data point and this prior aggregated data point may be an aggregation of the two
10 or more prior data points which are associated with this specific (e.g., one-minute) time interval. Hence, the prior data points may be aggregated in a time-dependent manner; therefore, the aggregated data points may also be referred to as time-dependent-aggregated data points.

[0017] Each data point may include a plurality of data elements. Aggregating data points
15 may include a determination of a respective value of one or more new data elements different from the data elements of the data points. For example, besides time information the prior data points (and also the new data points) may include a respective value of a specific parameter and an aggregated data point may include a data element representing an average of the values of the specific parameter within the corresponding time period of the
20 aggregated data point and/or a data element representing a maximum value of the specific parameter within the corresponding time period of the aggregated data point and/or a minimum value of the specific parameter within the corresponding time period of the aggregated data point, etc. Aggregating data points may also be referred to as transforming the data points.

[0018] The at least one processor 204 may be configured to determine the second table 210. The at least one processor 204 may be configured to determine a respective prior aggregated data point of the plurality of prior aggregated data points by aggregating all data points of the plurality of prior data points which are associated with the corresponding time interval of the respective prior aggregated data point.

[0019] The database 206 may include a materialized view 212. In general, a materialized view is a database object which allows a preprocessing of data points prior to inserting them into a table. For example, complex aggregates may be precalculated on (e.g., large) tables and stored in a materialized form. Accessing the materialized view may be significantly faster than accessing the source table (e.g., the first table 208) by using a conventional view. The increase in performance is due to the fact that considerably fewer input/output operations are required to determine the precalculated aggregates of the materialized view. Using a materialized view may also be referred to as materializing or materialization. Hence, a materialized view may be a special trigger that stores the result of a SELECT query on data, as it is inserted, into a target table.

[0020] The materialized view 212 may be configured to aggregate the data points within the first table. Illustratively, the materialized view 212 may store the preprocessed prior data points of the first table 208 in a virtual table.

[0021] The materialized view 212 may be configured such that, whenever one or more new data points are inserted into the first table 208, these one or more new data points are also provided to the materialized view 212. Hence, inserting one or more new data points into the first table 208 may trigger the materialized view 212 to (automatically) receive the one or more new data points. Illustratively, the materialized view 212 may be updated automatically.

[0022] The materialized view 212 may be configured to, whenever the materialized view 212 receives one or more new data points 214, determine one or more new aggregated data points 216 by aggregating all data points of the one or more new data points which are associated with a same time interval of the consecutive time intervals. Hence, for each time interval at least one of the one or more new data points 214 is associated with, the materialized view 212 may determine a respective new aggregated data point. The materialized view 212 may be configured to insert the one or more new aggregated data points 216 into the second table 210.

[0023] Thus, the materialized view 212 may be configured to insert one or more aggregated data points in real-time into the second table 210.

[0024] The second table 210 may be configured to (automatically) deduplicate (in 218) its entries to only include one aggregated data point for each time interval of the consecutive time intervals. This deduplication 218 may be carried out in the background. Hence, in the case that the one or more new aggregated data points are inserted into the second table 210 and that at least one new aggregated data point of the one or more new aggregated data points is associated with a same time interval as a prior aggregated data point, the second table 210 may (in 218) (automatically) deduplicate them by aggregating the at least one new aggregated data point and the prior aggregated data point (e.g., to update the prior aggregated data point) within the second table 210.

[0025] As an illustrative example, the at least one processor 204 may be configured to implement the database 206 using the column-oriented DMBS Clickhouse®. The at least one processor 204 may be configured to generate the first table 208 using the MergeTree engine of Clickhouse®. The at least one processor 204 may be configured to generate the second table 210 using the AggregatingMergeTree (e.g., ReplicatedAggregatingMergeTree) engine of Clickhouse®. Engines in

the MergeTree family are designed for inserting a very large amount of data into a table. The data is quickly written to the table part by part, then rules are applied for merging the parts in the background. This approach is much more efficient than continually rewriting the data in storage during insert.

5 **[0026]** According to various aspects, the at least one processor 204 may be configured to receive the real-time data 102 which may represent the one or more new data points 214. The at least one processor 204 may then insert the one or more new data points 214 into the first table 208, thereby triggering the materialized view 212 to receive the one or more new data points 214, to aggregate the one or more new data points 214 to one or more new
10 aggregated data points 216, and to insert the one or more new aggregated data points 216 into the second table 210. Inserting the one or more new aggregated data points 216 into the second table 210 may then trigger the second table 210 to deduplicate (in 218) aggregated data points associated with the same time interval (by aggregating them).

[0027] The at least one processor 204 may be configured to receive the database
15 query 302 from the electronic device 300. The database query 302 may request query data from the database 206. The at least one processor 204 may be configured to, in response to the database query 302, retrieve (e.g., using one or more select queries) data from the second table 210 in order to determine the query data 202. According to various aspects, the query data 202 may be a time series of equally spaced intervals having a same respective length
20 as each of the consecutive time intervals. Thus, the query data 202 can be directly retrieved from the second table 210 without further processing. If the data associated with the time series may be retrieved from the first table 208, the at least one processor 204 would have to aggregate the retrieved data points in order to determine the time series.

[0028] As an illustrative example, each data point stored in the first table 208 (hence
25 each prior data point and each new data point) may correspond to a tick of an asset price or

a (currency) foreign exchange rate (e.g., a US dollar, USD, to Singapore dollar, SGD, exchange rate). Thus, the data points may not be generated at equally spaced points in time but whenever a change in the asset price or FX rate exceeds a tick size thereof. In the second table 210, on the other hand, each aggregated data point may be bijectively assigned to one time interval of the consecutive time intervals and ,therefore, equally spaced in time.

[0029] In this exemplary scenario, the length of each time interval of the consecutive time intervals may correspond to one of the time interval lengths of charting such an asset price or FX rate. For example, the length of each time interval may be one of the following: one minute, five minutes, fifteen minutes, thirty minutes, one hour, five hours, one day, one week, one month, etc. This length may also be referred to as granularity.

[0030] Illustratively, the database 206 may downsample the data points from tick-level to data points covering a longer time intervals (e.g., one-minute time intervals). Illustratively, the present approach provides a time-dependent downsampling.

[0031] In combination with the exemplary case that the at least one processor 204 is configured to implement the database 206 using the column-oriented DMBS Clickhouse®, the query execution time may be reduced from 3573 ms when retrieving the data from the first table 208 to about 80 ms when retrieving the data from the second table 210 (hence, a reduction of about 44 times). If another DBMS would be used, such as HBase, the factor of reduction would even be greater.

[0032] Illustratively, according to various aspects granularity and/or precision may be compromised to have a faster query execution time. The approach disclosed herein may require low effort for its implementation. For example, writing and implementing a materialized view is much simpler than writing application-end logic to scan and collate results from several day-level (e.g., HBase) tables and combining them into one data structure.

[0033] According to various aspects, combining the use of the materialized view 212 with the deduplication within the second table 210 allows to significantly reduce the query execution time (e.g., for retrieving time series data). At the same time, the approach allows to provide the data in real-time. If a scheduled batch process would be used to downsample the prior data points, this would not allow to be real-time.

[0034] The one or more servers 100 may be, for example, a (Apache) Kafka server (cluster). The one or more servers 100 may be configured to stream the real-time data 102 (e.g., as Kafka Streams) to the database device 200.

[0035] According to various aspects, the database 206 may include a respective (second) table for one or more lengths of each time interval, such as one or more of the above lengths. In this case, the database 206 may include a respective materialized view for each (second) table. This may allow to increase the query execution time for each time series query which requests a time series with one of these lengths. For example, a user may request data for a one-minute chart and data for a five-hour chart: The at least one processor 204 may retrieve the data for the one-minute chart from (second) table storing the aggregates having a time interval length of one minute and may retrieve the data for the five-hour chart from (second) table storing the aggregates having a time interval length of five hours. Thus, both information can be directly retrieved from a respective table without requiring any further determination.

[0036] **FIG.3** is a block diagram showing an example electronic device 300 according various aspects. The electronic device 300 may be a laptop computer, a desktop computer, a tablet computer, an automobile computer, a smart phone, a personal digital assistant, a server, or other electronic devices capable of running computer applications. In some aspects, the electronic device 300 includes one or more processors 316, an input/output (I/O) module 304, a memory (in some aspects referred to as memory device) 306, a power unit

308, and one or more network interfaces 310. The electronic device 300 can include additional components. In some embodiments, the one or more processors 316, input/output (I/O) module 304, memory 306, power unit 308, and the one or more network interfaces 310 may be housed together in a common housing or other assembly.

5 **[0037]** The one or more processors 316 may be configured to execute instructions, for example, to generate output data based on data inputs. The instructions can include programs, codes, scripts, modules, or other types of data stored in memory (e.g., memory 306). Additionally or alternatively, the instructions may be encoded as pre-programmed or re-programmable logic circuits, logic gates, or other types of hardware or firmware
10 components or modules. A processor of the one or more processors 316 may be, or may include, a multicore processor having a plurality of cores, and each such core may have an independent power domain and can be configured to enter and exit different operating or performance states based on workload. Additionally or alternatively, a processor of the one or more processors 316 may be, or may include, a general-purpose microprocessor, as a
15 specialized co-processor or another type of data processing apparatus. In some cases, the one or more processors 316 may perform high-level operation of the electronic device 300. For example, the one or more processors 316 may be configured to execute or interpret software, scripts, programs, functions, executables, or other instructions stored in the memory 306.

20 **[0038]** The example I/O module 304 may include a mouse, keypad, touch screen, scanner, optical reader, and/or stylus (or other input device(s)) through which a user of the electronic device 300 may provide input to the electronic device 300, and may also include one or more audio speakers for providing audio output and a video display device for providing textual, audiovisual, and/or graphical output.

[0039] The memory 306 may include computer-readable storage media, for example, a volatile memory device, a non-volatile memory device, or both. The memory 306 may include one or more read-only memory devices, random-access memory devices, buffer memory devices, or a combination of these and other types of memory devices. In some instances, one or more components of the memory can be integrated or otherwise associated with another component of the electronic device 300. The memory 306 may store instructions that are executable by the one or more processors 316. In some examples, the memory 306 may store instructions for an operating system 312 and for application programs 314.

[0040] The example power unit 308 provides power to the other components of the electronic device 300. For example, the other components may operate based on electrical power provided by the power unit 308 through a voltage bus or other connection. In some embodiments, the power unit 308 may include a battery or a battery system, for example, a rechargeable battery. In some embodiments, the power unit 308 may include an adapter (e.g., an AC adapter) that receives an external power signal (from an external source) and converts the external power signal to an internal power signal conditioned for a component of the electronic device 300. The power unit 308 may include other components or operate in another manner.

[0041] The electronic device 300 may be configured to operate in a wireless, wired, or cloud network environment (or a combination thereof). In some embodiments, the electronic device 300 may access the network using the one or more network interfaces 310. The one or more network interfaces 310 may include one or more adapters, modems, connectors, sockets, terminals, ports, slots, and the like. The wireless network that the electronic device 300 accesses may operate, for example, according to a wireless network standard or another type of wireless communication protocol. For example, the wireless network may be

configured to operate as a Wireless Local Area Network (WLAN), a Personal Area Network (PAN), a metropolitan area network (MAN), or another type of wireless network. Examples of WLANs include networks configured to operate according to one or more of the 802.11 family of standards developed by IEEE (e.g., Wi-Fi networks), and others. Examples of

5 PANs include networks that operate according to short-range communication standards (e.g., BLUETOOTH®, Near Field Communication (NFC), ZigBee), millimeter wave communications, and others. The wired network that the electronic device 300 accesses may, for example, include Ethernet, SONET, circuit-switched networks (e.g., using components such as SS7, cable, and the like), and others.

10 **[0042]** According to various aspects, the electronic device 300 may be configured to transmit the database query 302 using the one or more network interfaces 310. According to various aspects, the database query 302 may be a downstream application programming interface, API, call of the electronic device 300 requesting the query data 202 from the database 206.

15 **[0043]** As detailed herein, in an exemplary scenario, the query data 202 may include a time series of an FX rate (up to real-time). The I/O module 304 may include a display device and the one or more processors 316 may be configured to provide control instructions to show (e.g., chart) a diagram (e.g., a chart) illustrating the time series of the FX rate on the display device. The time series of the FX rate may be associated with a specific currency

20 pair, a tenor, a granularity, etc. The time series may be a time series for a predefined time period (e.g., a number of M days) prior to the present time.

[0044] According to some aspects, the database device 200 may be part of the electronic device 300. Hence, in this case the one or more processors 316 may be configured to implement the database 206 and the memory 306 of the electronic device 300 may be

configured to store the database 206. Illustratively, the database 206 may be implemented locally on the electronic device 300.

[0045] FIG.4 shows a flow diagram of a method 400 for executing a (real-time) database query. The method 400 may be executed by at least one (e.g., one or more than
5 one) processor.

[0046] The method 400 may include (in 402) a first (e.g., source) table of a database receiving real-time data. The first table may include a plurality of prior data points of a data set. The real-time data may represent one or more new data points of the data set. The first table receiving the real-time data may trigger a materialized view of the database to receive
10 the one or more new data points.

[0047] The method 400 may include (in 404) the materialized view determining one or more new (time-dependent-) aggregated data points by aggregating (e.g., averaging) all data points of the one or more new data points which are associated with a same time interval of consecutive time intervals of equal length.

[0048] The method 400 may include (in 406) the materialized view inserting the aggregated data points into a second (e.g., destination) table. The second table may include a plurality of prior aggregated data points. Each of the plurality of prior aggregated data points may be associated with a corresponding time interval of the consecutive time intervals and may be an aggregation of (e.g., an average over) all data points of the plurality
15 of prior data points which are associated with the corresponding time interval.

[0049] The method 400 may include (in 408), in the case that at least one new aggregated data point of the one or more new aggregated data points and a prior aggregated data point of the plurality of prior aggregated data points are associated with a same time interval of the consecutive time intervals, aggregating the at least one new aggregated data
20

point and the prior aggregated data point (e.g., to update the prior aggregated data point) within the second table (e.g., in the background).

[0050] The method 400 may include (in 410) in response to the database query requesting query data from the database, retrieving (e.g., using one or more select queries) the query data from the second table.

[0051] The method 400 may include any aspects described with reference to the system 10 (e.g., with reference to the database device 200). For example, each data point of the data set may correspond to a tick of an asset price or a currency (foreign) exchange rate and the query data may be a time series of equally spaced intervals, each having a length corresponding to the length of each of the consecutive time intervals (e.g., having a length of one minute). According to various aspects, the method 400 may include charting the query data as the time series (e.g., on a display device of the electronic device 300).

[0052] While the method described above is illustrated and described as a series of steps or events, it will be appreciated that any ordering of such steps or events are not to be interpreted in a limiting sense. For example, some steps may occur in different orders and/or concurrently with other steps or events apart from those illustrated and/or described herein. In addition, not all illustrated steps may be required to implement one or more aspects or embodiments described herein. Also, one or more of the steps depicted herein may be carried out in one or more separate acts and/or phases.

[0053] A non-transitory computer-readable medium storing computer executable code including instructions for executing a (real-time) database query is also provided according to the method 400 as described above. The aforementioned instructions may include a computer program element or a computer program product.

[0054] Some of the subject matter and operations described in this specification can be implemented in digital electronic circuitry, or in computer software, firmware, or hardware,

including the structures disclosed in this specification and their structural equivalents, or in combinations of one or more of them. Some of the subject matter described in this specification can be implemented as one or more computer programs, i.e., one or more modules of computer program instructions, encoded on a computer storage medium for
5 execution by, or to control the operation of, data-processing apparatus. A computer storage medium can be, or can be included in, a computer-readable storage device, a computer-readable storage substrate, a random or serial access memory array or device, or a combination of one or more of them. Moreover, while a computer storage medium is not a propagated signal, a computer storage medium can be a source or destination of computer
10 program instructions encoded in an artificially generated propagated signal. The computer storage medium can also be, or be included in, one or more separate physical components or media (e.g., multiple CDs, disks, or other storage devices).

[0055] The following examples pertain to various aspects of the present disclosure. It is noted that aspects described with reference to a device may be accordingly implemented in
15 a method, and vice versa.

[0056] Example 1 is a computer-implemented method for executing a (real-time) database query, the method executed by at least one processor, and including: a first (e.g., source) table of a database receiving real-time data, the first table including a plurality of prior data points of a data set, wherein the real-time data represent one or more new data
20 points of the data set, wherein the first table receiving the real-time data triggers a materialized view of the database to receive the one or more new data points; the materialized view determining one or more new (time-dependent-) aggregated data points by aggregating (e.g., averaging) all data points of the one or more new data points which are associated with a same time interval of consecutive time intervals of equal length; the
25 materialized view inserting the aggregated data points into a second (e.g., destination) table,

the second table including a plurality of prior aggregated data points, wherein each of the plurality of prior aggregated data points is associated with a corresponding time interval of the consecutive time intervals and is an aggregation of (e.g., an average over) all data points of the plurality of prior data points which are associated with the corresponding time interval; in the case that at least one new aggregated data point of the one or more new aggregated data points and a prior aggregated data point of the plurality of prior aggregated data points are associated with a same time interval of the consecutive time intervals, aggregating the at least one new aggregated data point and the prior aggregated data point (e.g., to update the prior aggregated data point) within the second table (e.g., in the background); and in response to the database query requesting query data from the database, retrieving (e.g., using one or more select queries) the query data from the second table.

[0057] In Example 2, the subject matter of Example 1 can optionally include that each data point of the data set corresponds to a tick of an asset price or a currency foreign exchange rate.

[0058] In Example 3, the subject matter of Example 1 or 2 can optionally include that the query data are a time series of equally spaced intervals, each of the equally spaced intervals having a length corresponding to the length of each of the consecutive time intervals.

[0059] In Example 4, the subject matter of any one of Examples 1 to 3 can optionally include that the length of each time interval of the consecutive time intervals is one minute.

[0060] In Example 5, the subject matter of Examples 2 and 4 can optionally include that aggregating all data points of the one or more new data points which are associated with a same time interval of the consecutive time intervals includes downsampling the data points from tick-level to minute-level.

[0061] In Example 6, the method of any one of Examples 1 to 5 can optionally further include: receiving the database query from a downstream application programming interface, API, call.

[0062] In Example 7, the subject matter of any one of Examples 1 to 6 can optionally
5 include that the database is a column-oriented database (e.g., implemented by Clickhouse® database management system).

[0063] In Example 8, the subject matter of any one of Examples 1 to 7 can optionally include that the first table receives the real-time data (e.g., as a stream) from one or more (upstream) servers (e.g., an upstream (e.g., Apache Kafka) server cluster).

[0064] In Example 9, the method of any one of Examples 1 to 8 can optionally further
10 include: generating the second table by aggregating all data points of the plurality of prior data points which are associated with a same time interval of the consecutive time intervals to determine the prior aggregated data point associated with the time interval.

[0065] Example 10 is a non-transitory computer-readable medium storing computer
15 executable code including instructions for executing a database query according to the method of any one of Examples 1 to 9.

[0066] Example 11 is a device for executing a (real-time) database query, the device including: at least one processors configured to: implement a database (e.g., employing a database management system, such as Clickhouse®) including a first (e.g., source) table, a
20 materialized view, and a second (e.g., destination) table, wherein the first table includes a plurality of prior data points of a data set, wherein the second table includes a plurality of prior aggregated data points, wherein each of the plurality of prior aggregated data points is associated with a corresponding time interval of consecutive time intervals of equal length and is an aggregation of (e.g., an average over) all data points of the plurality of prior data
25 points which are associated with the corresponding time interval; receive real-time data

representing one or more new data points of the data set; insert the one or more new data points into the first table, wherein the materialized view is configured to (e.g., automatically) receive the one or more new data points in response to inserting the one or more new data points into the first table, to determine one or more new (time-dependent-) aggregated data points by aggregating (e.g., averaging) all data points of the one or more new data points which are associated with a same time interval of the consecutive time intervals, and to insert the one or more new aggregated data points into the second table; wherein the second table is configured to, in the case that at least one new aggregated data point of the one or more new aggregated data points and a prior aggregated data point of the plurality of prior aggregated data points are associated with a same time interval of the consecutive time intervals, aggregate the at least one new aggregated data point and the prior aggregated data point (e.g., to update the prior aggregated data point) within the second table (e.g., in the background); receive a database query requesting query data from the database; and in response to the database query, retrieve (e.g., using one or more select queries) the query data from the second table.

[0067] In Example 12, the subject matter of Example 11 can optionally include that each data point of the data set corresponds to a tick of an asset price or a currency foreign exchange rate.

[0068] In Example 13, the subject matter of Example 11 or 12 can optionally include that the query data are a time series of equally spaced intervals, each of the equally spaced intervals having a length corresponding to the length of each of the consecutive time intervals.

[0069] In Example 14, the subject matter of any one of Examples 11 to 13 can optionally include that the length of each time interval of the consecutive time intervals is one minute.

[0070] In Example 15, the subject matter of Examples 12 and 14 can optionally include that the at least one processor is configured to aggregate data points which are associated with a same time interval of the consecutive time intervals to an aggregated data point to downsample the data points from tick-level to minute-level.

5 **[0071]** In Example 16, the subject matter of any one of Examples 11 to 15 can optionally include that the at least one processors is configured to receive the database query from a downstream application programming interface, API, call.

[0072] In Example 17, the subject matter of any one of Examples 11 to 16 can optionally include that the database is a column-oriented database (e.g., implemented by Clickhouse®
10 database management system).

[0073] In Example 18, the subject matter of any one of Examples 11 to 17 can optionally include that the at least one processor is configured to receive the real-time data (e.g., as a stream) from one or more (upstream) servers (e.g., an upstream (e.g., Apache Kafka) server cluster).

15 **[0074]** In Example 19, the subject matter of any one of Examples 11 to 18 can optionally include that the at least one processors is configured to determine a respective prior aggregated data point of the plurality of prior aggregated data points by aggregating all data points of the plurality of prior data points which are associated with the corresponding time interval of the respective prior aggregated data point.

20 **[0075]** In Example 19, the subject matter of any one of Examples 11 to 18 can optionally further include a memory (device) storing the database.

[0076] It should be understood that the singular terms "a", "an", and "the" include plural references unless context clearly indicates otherwise. Similarly, the word "or" is intended to include "and" unless the context clearly indicates otherwise.

[0077] It will be further understood that the terms “comprise” (and any form of comprise, such as “comprises” and “comprising”), “have” (and any form of have, such as “has” and “having”), “include” (and any form of include, such as “includes” and “including”), and “contain” (and any form of contain, such as “contains” and “containing”) are open-ended linking verbs. As a result, a method or device that “comprises,” “has,” “includes” or “contains” one or more steps or elements possesses those one or more steps or elements, but is not limited to possessing only those one or more steps or elements. Likewise, a step of a method or an element of a device that “comprises,” “has,” “includes” or “contains” one or more features possesses those one or more features, but is not limited to possessing only those one or more features. Furthermore, a device or structure that is configured in a certain way is configured in at least that way, but may also be configured in ways that are not listed.

[0078] Approximating language, as used herein throughout the specification and claims, may be applied to modify any quantitative representation that could permissibly vary without resulting in a change in the basic function to which it is related. Accordingly, a value modified by a term or terms, such as “about,” “substantially”, is not limited to the precise value specified but within tolerances that are acceptable for operation of the embodiment for an application for which it is intended. In some instances, the approximating language may correspond to the precision of an instrument for measuring the value.

[0079] As used herein, the phrase of the form of “at least one of A or B” may include A or B or both A and B. Correspondingly, the phrase of the form of “at least one of A or B or C”, or including further listed items, may include any and all combinations of one or more of the associated listed items.

[0080] The term “exemplary” may be used herein to mean “serving as an example, instance, or illustration”. Any aspect or design described herein as “exemplary” is not necessarily to be construed as preferred or advantageous over other aspects or designs.

[0081] The terms “at least one” and “one or more” may be understood to include a
5 numerical quantity greater than or equal to one (e.g., one, two, three, four, [...], etc.). The term “a plurality” may be understood to include a numerical quantity greater than or equal to two (e.g., two, three, four, five, [...], etc.). The phrase “at least one of” with regard to a group of elements may be used herein to mean at least one element from the group consisting of the elements. For example, the phrase “at least one of” with regard to a group of elements
10 may be used herein to mean a selection of: one of the listed elements, a plurality of one of the listed elements, a plurality of individual listed elements, or a plurality of a multiple of listed elements.

[0082] The words “plural” and “multiple” in the description and in the claims expressly refer to a quantity greater than one. Accordingly, any phrases explicitly invoking the
15 aforementioned words (e.g., “a plurality of (objects)”, “multiple (objects)”) referring to a quantity of objects expressly refer to more than one of the said objects. The terms “group (of)”, “set (of)”, “collection (of)”, “series (of)”, “sequence (of)”, “grouping (of)”, etc., and the like in the description and in the claims, if any, refer to a quantity equal to or greater than one, i.e. one or more.

[0083] The term “data” as used herein may be understood to include information in any
20 suitable analog or digital form, e.g., provided as a file, a portion of a file, a set of files, a signal or stream, a portion of a signal or stream, a set of signals or streams, and the like. Further, the term “data” may also be used to mean a reference to information, e.g., in form of a pointer. The term “data”, however, is not limited to the aforementioned examples and
25 may take various forms and represent any information as understood in the art. Any type of

information, as described herein, may be handled for example via one or more processors in a suitable way, e.g. as data.

[0084] The terms “processor” or “controller” as, for example, used herein may be understood as any kind of entity that allows handling data. The data may be handled according to one or more specific functions executed by the processor or controller. Further, a processor or controller as used herein may be understood as any kind of circuit, e.g., any kind of analog or digital circuit. A processor or a controller may thus be or include an analog circuit, digital circuit, mixed-signal circuit, logic circuit, processor, microprocessor, Central Processing Unit (CPU), Graphics Processing Unit (GPU), Digital Signal Processor (DSP), Field Programmable Gate Array (FPGA), integrated circuit, Application Specific Integrated Circuit (ASIC), etc., or any combination thereof. Any other kind of implementation of the respective functions, which will be described below in further detail, may also be understood as a processor, controller, or logic circuit. It is understood that any two (or more) of the processors, controllers, or logic circuits detailed herein may be realized as a single entity with equivalent functionality or the like, and conversely that any single processor, controller, or logic circuit detailed herein may be realized as two (or more) separate entities with equivalent functionality or the like.

[0085] The term “memory” as used herein may be understood as a computer-readable medium (e.g., a non-transitory computer-readable medium), in which data or information can be stored for retrieval. References to “memory” included herein may thus be understood as referring to volatile or non-volatile memory, including random access memory (RAM), read-only memory (ROM), flash memory, solid-state storage, magnetic tape, hard disk drive, optical drive, among others, or any combination thereof. Furthermore, it is appreciated that registers, shift registers, processor registers, data buffers, among others, are also embraced herein by the term memory. It is also appreciated that a single component

referred to as “memory” or “a memory” may be composed of more than one different type of memory, and thus may refer to a collective component including one or more types of memory. It is readily understood that any single memory component may be separated into multiple collectively equivalent memory components, and vice versa. Furthermore, while
5 memory may be depicted as separate from one or more other components (such as in the drawings), it is understood that memory may be integrated within another component, such as on a common integrated chip.

[0086] The term “module” detailed herein refers to, or forms part of, or includes an Application Specific Integrated Circuit (ASIC); an electronic circuit; a combinational logic
10 circuit; a field programmable gate array (FPGA); a processor (shared, dedicated, or group) that executes code; other suitable hardware components that provide the described functionality; or a combination of some or all of the above, such as in a system-on-chip. The term module may include memory (shared, dedicated, or group) that stores code executed by the processor.

15 **[0087]** The term “first”, “second”, “third” detailed herein are used to distinguish one element from another similar element and may not necessarily denote order or relative importance, unless otherwise stated. For example, a first table and a second table may be used to distinguish two tables including different data points.

[0088] Some of the operations described in this specification can be implemented as
20 operations performed by a data processing apparatus on data stored on one or more computer-readable storage devices or received from other sources.

[0089] The term “data processing apparatus” encompasses all kinds of apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, a system on a chip, or multiple ones, or combinations, of the
25 foregoing. The apparatus can include special purpose logic circuitry, e.g., an FPGA (field

programmable gate array) or an ASIC (application specific integrated circuit). The apparatus can also include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, a cross-platform runtime
5 environment, a virtual machine, or a combination of one or more of them.

[0090] A computer program (also known as a program, software, software application, script, or code) can be written in any form of programming language, including compiled or interpreted languages, declarative or procedural languages, and it can be deployed in any form, including as a stand-alone program or as a module, component, subroutine, object, or
10 other unit suitable for use in a computing environment. A computer program may, but need not, correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program, or in multiple coordinated files (e.g., files that store one or more modules, sub programs, or portions of code). A computer
15 program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[0091] Various aspects detailed herein refer to a “database”. In general, a database may be an organized collection of data. For this purpose, the database may include one or more
20 tables. Each table of the one or more tables may include (e.g., store) dataset (or more datasets) in an organized manner. A database (e.g., the organized collection of data, the tables or other database objects, operations which can be applied on the objects (e.g., a table) of the database, etc.) may be configured in accordance with a corresponding “database management system” (DBMS). Hence, a DBMS may be employed in order to implement a
25 database.

[0092] A dataset may be a (structured) collection of data points, wherein each data point may include two or more data elements. Each data element of a data point may be associated with an entry in a table which includes the data point.

[0093] A database query may be a request for (query) data from a database (e.g., a request to retrieve data).

[0094] The DBMS may be, for example, a column-oriented DBMS, such as Clickhouse®. In a column-oriented DBMS, data is stored in columns of a table, with values from the same columns stored together. In the case of column-oriented databases (e.g., tables thereof), only some columns (of the table) may be read in order to respond to a database query (of a user). By this, a response to the database query may be substantially faster as compared to row-oriented databases in which the data are stored in rows and all columns of a row are read in order to respond to a database query (of a user). Hence, a query execution time in the case of column-oriented databases may be less than a query execution time in the case of row-oriented databases. The “query execution time” may be a time period between reception of a database query requesting query data from a database and output of the requested query data. Illustratively, the query execution time may be the time required to complete the database query. The requested data may be a subset of a dataset stored in the database.

[0095] Various aspects detailed herein refer to the term “real-time”. In general, the term “real-time” may characterize the operation of an information technology system that can reliably deliver certain results within a predefined period of time. Hence, a process (e.g., a processing, transmission, etc.) may be real-time in the case that the time required to provide a result of the process is less than the predefined period of time. The time required to provide the result may be latency time of the process. Illustratively, a process may be real-time in the case that there is no unnecessary delay.

[0096] As an example, in the scenario of the present application, an asset price or a (currency) foreign exchange rate may change by a tick. One or more servers (e.g., a server cluster) may observe the change and obtain a corresponding data point. The one or more servers may then transmit one or more of such data points to the device 300 and the device 5 300 may update the second table based on the one or more data points. Once the second table is updated, a user can retrieve the updated data points from the second table by sending a database query. In this example, the time period between the asset price or (currency) foreign exchange rate may changing by the tick and the user receiving the updated data points may be less than the predefined period of time in order to provide the information in 10 “real-time” to the user.

[0097] In general, the term “time series” or “time-series” may refer to a sequence of data points over a time interval. Herein, a time-series may refer to data having a respective data point for each time interval of a plurality of consecutive time intervals. The time intervals of the plurality of consecutive time intervals may be equally spaced. Hence, each 15 time interval of the plurality of consecutive time intervals may have a same length. It is understood that consecutive time intervals are characterized in that there is no gap in time between a time interval and an immediately subsequent time interval.

[0098] A “tick” of an asset price or a (currency) foreign exchange (FX) rate may be a measure of a minimum upward or downward movement in the asset price/foreign exchange 20 rate. Each asset and each FX rate may have a corresponding tick size with a tick being a change in accordance with the corresponding tick size. Therefore, a tick is the minimum incremental amount at which an asset/FX rate can be traded. Illustratively, a tick may represent the standard upon which asset/FX rate may fluctuate. Hence, if a price moves N times per second, N data points are generated, whereas if prices do not move, no data point 25 is generated.

[0099] A foreign exchange (FX) rate may also be referred to as currency FX rate or as FX rate of a currency (or currency pair).

[00100] Some of the processes and logic flows described in this specification can be performed by one or more programmable processors executing one or more computer
5 programs to perform actions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit).

[00101] While this specification contains many details, these should not be understood
10 as limitations on the scope of what may be claimed, but rather as descriptions of features specific to particular examples. Certain features that are described in this specification or shown in the drawings in the context of separate embodiments can also be combined. Conversely, various features that are described or shown in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable
15 sub-combination.

[00102] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be
20 advantageous. Moreover, the separation of various system components in the embodiments described above should not be understood as requiring such separation in all embodiments, and it should be understood that the described program components and systems can generally be integrated together in a single product or packaged into multiple products.

[00103] A number of embodiments have been described. Nevertheless, it will be understood that various modifications can be made. Accordingly, other embodiments are within the scope of the following claims.

CLAIMS

What is claimed is:

1. A computer-implemented method for executing a database query, the method executed by at least one processor, and comprising:

a first table of a database receiving real-time data, the first table comprising a plurality of prior data points of a data set, wherein the real-time data represent one or more new data points of the data set, wherein the first table receiving the real-time data triggers a materialized view of the database to receive the one or more new data points;

the materialized view determining one or more new aggregated data points by aggregating all data points of the one or more new data points which are associated with a same time interval of consecutive time intervals of equal length;

the materialized view inserting the aggregated data points into a second table, the second table comprising a plurality of prior aggregated data points, wherein each of the plurality of prior aggregated data points is associated with a corresponding time interval of the consecutive time intervals and is an aggregation of all data points of the plurality of prior data points which are associated with the corresponding time interval;

in the case that at least one new aggregated data point of the one or more new aggregated data points and a prior aggregated data point of the plurality of prior aggregated data points are associated with a same time interval of the consecutive time intervals, aggregating the at least one new aggregated data point and the prior aggregated data point within the second table; and

in response to the database query requesting query data from the database, retrieving the query data from the second table.

2. The method according to claim 1, wherein each data point of the data set corresponds to a tick of an asset price or a currency foreign exchange rate.
3. The method according to claim 1 or 2, wherein the query data are a time series of equally spaced intervals, each of the equally spaced intervals having a length corresponding to the length of each of the consecutive time intervals.
4. The method according to any one of claims 1 to 3, wherein the length of each time interval of the consecutive time intervals is one minute.
5. The method according to claim 4, wherein each data point of the data set corresponds to a tick of an asset price or a currency foreign exchange rate, and wherein aggregating all data points of the one or more new data points which are associated with a same time interval of the consecutive time intervals comprises downsampling the data points from tick-level to minute-level.
6. The method according to any one of claims 1 to 5, further comprising: receiving the database query from a downstream application programming interface call.
7. The method according to any one of claims 1 to 6, wherein the database is a column-oriented database.
8. The method according to any one of claims 1 to 7, wherein the first table receives the real-time data from one or more servers.

9. The method according to any one of claims 1 to 8, further comprising:
generating the second table by aggregating all data points of the plurality of prior data points which are associated with a same time interval of the consecutive time intervals to determine the prior aggregated data point associated with the time interval.
10. A non-transitory computer-readable medium storing computer executable code comprising instructions for executing a database query according to the method of claim 1 to 9.
11. A device for executing a database query, the device comprising:
at least one processors configured to:
implement a database comprising a first table, a materialized view, and a second table, wherein the first table comprises a plurality of prior data points of a data set, wherein the second table comprises a plurality of prior aggregated data points, wherein each of the plurality of prior aggregated data points is associated with a corresponding time interval of consecutive time intervals of equal length and is an aggregation of all data points of the plurality of prior data points which are associated with the corresponding time interval;
receive real-time data representing one or more new data points of the data set;
insert the one or more new data points into the first table, wherein the materialized view is configured to receive the one or more new data points in response to inserting the one or more new data points into the first table, to determine one or more new aggregated data points by aggregating all data points of the one or more new data points which are associated with a same time interval of the

consecutive time intervals, and to insert the one or more new aggregated data points into the second table;

wherein the second table is configured to, in the case that at least one new aggregated data point of the one or more new aggregated data points and a prior aggregated data point of the plurality of prior aggregated data points are associated with a same time interval of the consecutive time intervals, aggregate the at least one new aggregated data point and the prior aggregated data point within the second table;

receive a database query requesting query data from the database; and

in response to the database query, retrieve the query data from the second table.

12. The device according to claim 11, wherein each data point of the data set corresponds to a tick of an asset price or a currency foreign exchange rate.

13. The device according to claim 11 or 12, wherein the query data are a time series of equally spaced intervals, each of the equally spaced intervals having a length corresponding to the length of each of the consecutive time intervals.

14. The device according to any one of claims 11 to 13, wherein the length of each time interval of the consecutive time intervals is one minute.

15. The device according to claim 14, wherein each data point of the data set corresponds to a tick of an asset price or a currency foreign exchange rate, and wherein the at least one processor is configured to aggregate data points which are associated with a

same time interval of the consecutive time intervals to an aggregated data point to downsample the data points from tick-level to minute-level.

16. The device according to any one of claims 11 to 15, wherein the at least one processors is configured to receive the database query from a downstream application programming interface call.

17. The device according to any one of claims 11 to 16, wherein the database is a column-oriented database.

18. The device according to any one of claims 11 to 17, wherein the at least one processor is configured to receive the real-time data from one or more servers.

19. The device according to any one of claims 11 to 18, wherein the at least one processors is configured to determine a respective prior aggregated data point of the plurality of prior aggregated data points by aggregating all data points of the plurality of prior data points which are associated with the corresponding time interval of the respective prior aggregated data point.

20. The device according to any one of claims 11 to 18, further comprising a memory device storing the database.