



- (51) **International Patent Classification:**
G06F 17/40 (2006.01) *G06F 3/06* (2006.01)
- (21) **International Application Number:**
PCT/US2016/026790
- (22) **International Filing Date:**
8 April 2016 (08.04.2016)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
201641006473 24 February 2016 (24.02.2016) IN
- (71) **Applicant:** ENTIT SOFTWARE LLC [US/US]; 1140 Enterprise Way, Building G, Sunnyvale, CA 94089 (US).
- (72) **Inventors:** OBEROI, Danny; Salarpuria GR Tech Park, Akash Block, GF to 8th Floor Khatha No. 69/3, Mahadevpura City, Municipal Council, Bangalore, Karnataka 560066 (IN). SINGH, Jaspreet; Sy. No. 192, Whitefield Road, Mahadevpura Post, Bangalore, Karnataka 560048 (IN). GANDHI, Ibha; Sy. No. 192, Whitefield Road, Mahadevpura Post, Bangalore, Karnataka 560048 (IN).
- (74) **Agents:** LEE, Rachel Jeong-Eun et al.; Hewlett Packard Enterprise, 3404 E. Harmony Road Mail Stop 79, Fort Collins, Colorado 80528 (US).

(81) **Designated States** (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

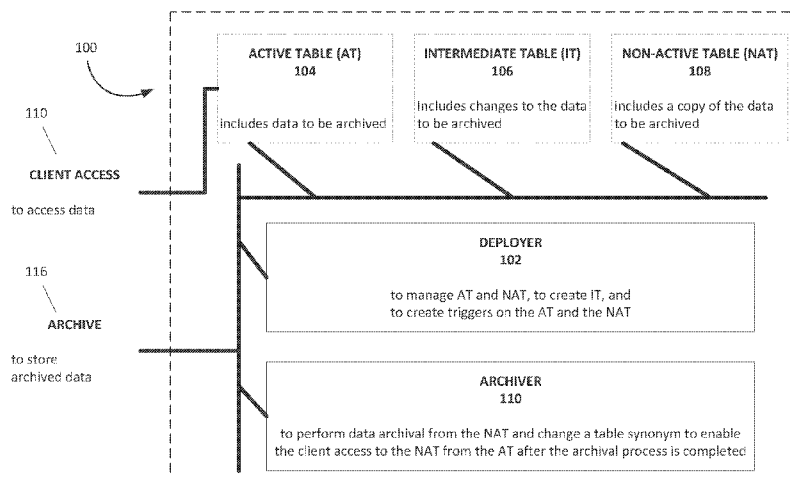
(84) **Designated States** (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

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

[Continued on next page]

(54) **Title:** STRUCTURED DATA ARCHIVAL WITH REDUCED DOWNTIME

**Fig. 1**

(57) **Abstract:** Structured data archival with reduced downtime is disclosed. One example is a system including a deployer that manages an active table (AT), and a non-active table (NAT), and creates an intermediate table (IT) to record, during data archival, changes to the data to be archived. The deployer creates triggers on the AT and the NAT to facilitate the record, by the IT, of the changes to the data to be archived. An archiver initiates the data archival by archiving the copy of the data to be archived from the NAT, merges the recorded data from the IT to the NAT upon receiving an indication that the client access to the AT is not enabled, and switches the client access from the AT to the NAT by changing a table synonym, where the client access to the NAT is enabled upon completion of the data archival.





Published:

— *with international search report (Art. 21(3))*

STRUCTURED DATA ARCHIVAL WITH REDUCED DOWNTIME

Background

[0001] Many businesses provide online access to client databases. Such access is generally available around the clock. These databases are often large, and may be frequently updated. Data in the databases may be archived. System downtime may be necessary for data archival, making the client data inaccessible to the client.

Brief Description of the Drawings

[0002] Figure 1 is a functional block diagram illustrating one example of a system for structured data archival with reduced downtime.

[0003] Figure 2 is a block diagram illustrating one example of a computer readable medium for structured data archival with reduced downtime.

[0004] Figure 3 is a flow diagram illustrating one example of a method for structured data archival with reduced downtime.

[0005] Figure 4 is a flow diagram illustrating one example of a method for creating an intermediate table for structured data archival with reduced downtime.

[0006] Figure 5 is a flow diagram illustrating one example of a method for initiating data archival for structured data archival with reduced downtime.

[0007] Figure 6 is a flow diagram illustrating one example of a method for creating triggers on active and non-active tables.

[0008] Figure 7 is a flow diagram illustrating another example of a method for structured data archival with reduced downtime.

Detailed Description

[0009] Many large and medium scale businesses are dependent on large business databases that are continuously available to clients. For example,

banking sites, e-commerce sites, online trading sites, and so forth, are accessible to clients at all times. These databases generally grow at a rapid rate, and data archiving is generally utilized to meet such data growth.

[0010] Structured data archival enables businesses to retire outdated applications through an automated process of extracting, validating, and deleting data. Archiving results in significantly reduced capital expenses and associated administrative costs, improved productivity for users of business applications, efficient Information Technology (“IT”) staffing solutions, reduced response times to legal and/or compliance queries, and maximized value extraction for data. Once archiving is complete, a reduced size for the production database enables ease of maintenance.

[0011] Data archival is generally more challenging than performing data backups. For example, the time taken to delete archived data may involve a considerable time investment. In such instances, database resources are locked until deletion of archived data is completed, and such downtime may be in hours or days. Downtime may cause a business to incur significant losses, as all services are put on hold when the systems are down. At the same time, archiving is necessary to maintain business efficiency, and may be necessary to maintain data for legal compliance purposes, system performance, and data growth. As a result, for business databases that need to be continually available, as for example, databases for Airlines, insurance companies, retail, banking applications, and so forth, data archival becomes a necessity on one hand, and a considerable challenge on the other.

[0012] As described herein, predictive archiving of a structured dataset may be performed with reduced downtime. The term “archive”, as used herein, generally refers to permanent or long-term preservation of contents of a dataset in a suitable database, generally referred to as an archive. The term “reduced downtime” generally refers to a reduction of an amount of time required for a downtime so that the dataset may be archived. The term “predictive” refers to a real-time selection of portions of the structured dataset that are to be archived, where the selections are instrumental in minimizing the downtime. As described herein, this predictive archiving may be facilitated by the use of triggers. The

term “trigger” generally refers to a specialized stored procedure for a database that may be designed to automatically execute in response to a condition being satisfied in the database server. For example, DML triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements associated with portions of data on a table or a view in a database. A predictive archive deployer creates triggers on active and non-active managed tables. A predictive archive runtime selector switches between active and non-active managed tables which brings a down time from hours to minutes.

[0013] As described in various examples herein, structured data archival with reduced downtime is disclosed. One example is a system including a deployer that manages an active table (AT), and a non-active table (NAT), and creates an intermediate table (IT) to record, during data archival, changes to the data to be archived. The deployer creates triggers on the AT and the NAT to facilitate the record, by the IT, of the changes to the data to be archived. An archiver initiates the data archival by archiving the copy of the data to be archived from the NAT, merges the recorded data from the IT to the NAT upon receiving an indication that the client access to the AT is not enabled, and switches the client access from the AT to the NAT by changing a table synonym, where the client access to the NAT is enabled upon completion of the data archival.

[0014] In the following detailed description, reference is made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration specific examples in which the disclosure may be practiced. It is to be understood that other examples may be utilized, and structural or logical changes may be made without departing from the scope of the present disclosure. The following detailed description, therefore, is not to be taken in a limiting sense, and the scope of the present disclosure is defined by the appended claims. It is to be understood that features of the various examples described herein may be combined, in part or whole, with each other, unless specifically noted otherwise.

[0015] Figure 1 is a functional block diagram illustrating one example of a system for structured data archival with reduced downtime. System 100

includes a deployer 102, and an archiver 110. The deployer 102 is to manage an active table (AT) 104, and a non-active table (NAT) 108, where client access to the AT 104 is enabled, and where the AT 104 includes data to be archived, and where the client access to the NAT 108 is not enabled, and where the NAT 108 includes a copy of the data to be archived. The term “table” generally refers to any database object utilized to store data. In some examples, the table may be utilized to store data in a structured database. In some examples, the table may have a primary/unique (e.g., not null) key.

[0016] Generally, the AT 104 may be available for client access 110. For example, AT 104 may be a table supporting a client portal available to a customer in an online banking system. As another example, AT 104 may be a table supporting an interactive trading platform available to a customer in an online trading platform. The AT 104 may be modified in real-time based on client interactions. The AT 104 may be in a schema that is specific to client access 110. Such schema may depend on the type of client access 110 (e.g., banking platform, trading platform, online shopping platform, etc.), and may depend on modifications made by a client. For example, different business may provide different client interfaces, and the schema for the ATs 104 may be tailored to such different business needs.

[0017] A purpose of the archival process is to archive 116 data contents of the AT 104 so that the data contents are available for client access 110 with reduced downtime. As described herein, this may be achieved by copying contents of the AT 104 into the NAT 108, archiving data from the NAT 108, recording the changes to the AT 104 in an intermediate table (IT) 106, initiating a reduced downtime when client access 110 to the AT 104 is suspended, merging contents of the IT 106 into NAT 108, swapping links between the AT 104 and NAT 108, so that NAT 108 is now available for client access 110.

[0018] The NAT 108 is a copy of the contents of the AT 104. The NAT 108 is generally not available for client access 110. The NAT 108 may be created in a schema that is based on the schema for the AT 104. In some examples, the schema for the NAT 108 may be identical to the schema for the AT 104. In some examples, the schema for the NAT 108 may be different from the schema

for AT 104. For example, the schema for the NAT 108 may be designed to fit storage parameters for archival. In some examples, the schema for the NAT 108 may be designed to comply with archival rules and/or policies.

[0019] The deployer 102 is to create an intermediate table (IT) 106 to record, during data archival, changes to the data to be archived. As described herein, the AT 104 is available for client access 110. A copy of the AT 104 is made onto the NAT 106. However, changes to the data contents of the AT 104 may occur during and/or subsequent to copying of the data contents from the AT 104 to the NAT 108. For example, AT 104 may be a table supporting a client portal available to a customer in an online banking system. The contents of AT 104 may be copied onto NAT 108 at a first time. Subsequently, the customer may log into the client portal and make changes to the online banking account (e.g., transfer funds between accounts, withdraw funds, etc.). Such activity is not saved in the NAT 108. However, the IT 106 records such intermediate changes to the AT 104 that are not copied onto the NAT 108.

[0020] In some examples, the deployer 102 is to create a new schema for the NAT 108, and create the IT 106 based on one of the new schema for the NAT 108 and an existing schema for the AT 104. As described herein, the deployer 102 may create a new schema for the NAT 108 based on a number of factors (e.g., schema for the AT 104, client type, client access type, archival parameters, archive rules, archive policies, and so forth). Accordingly, the deployer 102 may create the IT 106 based on the existing schema for the AT 104 or based on the new schema for the NAT 108.

[0021] The deployer 102 creates triggers on the AT 104 and the NAT 108 to facilitate the record, by the IT 106, of the changes to the data to be archived. As described herein, data in the AT 104 may change subsequent to copying its contents onto the NAT 108. The IT 106 records such changes based on the triggers that are created by the deployer 102. In some examples, the triggers may include one of insert, update, and delete triggers. For example, in the context of an online banking system, the AT 104 may be a table supporting a customer's banking account. Subsequent to copying data contents from the AT 104 to the NAT 108, a new deposit may be made into the customer's account.

Accordingly, the deployer 102 may create an insert trigger on the AT 104 to indicate to the IT 106 that this new transaction is to be inserted into NAT 108. As another example, subsequent to copying data contents from the AT 104 to the NAT 108, the new deposit will alter the total available balance in the customer account. Accordingly, the deployer 102 may create an update trigger on the AT 104 to indicate to the IT 106 that the balance amount is to be updated in the NAT 108. Also, for example, subsequent to copying data contents from the AT 104 to the NAT 108, the new deposit may facilitate removal of a “hold” status in the customer account. Accordingly, the deployer 102 may create a delete trigger on the AT 104 to indicate to the IT 106 that the “hold” is to be deleted in the NAT 108. Although the deployer 102 creates the triggers in the AT 104 and NAT 108, these triggers are generally disabled until archival begins. In some examples, the deployer 102 is to record information associated with the AT 104 and the NAT 108 in a metadata table.

[0022] System 100 includes an archiver 110 to initiate the data archival by archiving the copy of the data to be archived from the NAT 108. Accordingly, contents of the AT 104 continue to be available for client access 110, whereas a copy of the contents are archived from the copy in the NAT 108. The archiver 110 determines if a system downtime is available. The term “downtime” as used herein generally refers to a period of time when the client access 110 is disabled. Downtime may result in huge losses for a company, as services may be put on hold when systems are down. Generally, the archiver 110 may identify the information associated with the AT 104 from the metadata table that was created by the deployer 102. The archiver 110 may enable triggers on the AT 104.

[0023] In some examples, the archiver 110 may select and copy the entire data from the AT 104 to the NAT 108. In some examples, such a selection may be based on the information from the metadata table. The archiver 110 may start archiving on the NAT 108 based on rules and policy, including, for example, the select, copy and/or delete triggers. For example, based on rules and archiving policies, the archiver 110 may archive a selected portion of data from the NAT 108.

[0024] In some examples, the archiver 110 may pause and wait for an indication that the client access to the AT 104 is not enabled, i.e., an indication that a downtime has been initiated. The archiver 110 may merge the recorded data from the IT 106 to the NAT 108 upon receiving the indication that the client access to the AT 104 is not enabled. The archiver 110 merges the changes to the data to be archived from the IT 106 to the NAT 108, based on the triggers.

[0025] The archiver 110 switches the client access 110 from the AT 104 to the NAT 108 by changing a table synonym, where the client access to the NAT 108 is enabled upon completion of the data archival. In some examples, the table synonym is an alternate name for the NAT 108. When the table synonym is changed, the client access 110 is switched from the AT 104 to the NAT 108, where the NAT 108 is a new active table. This may be achieved, for example, by pointing a managed table synonym to the new active table which is the previous NAT 108. Changing the table synonym minimizes the amount of time for swapping access links from the AT 104 to the NAT 108. The archiver 110 then updates the metadata with the information about the new active table. In some examples, the archiver 110 is to provide, upon completion of the data archival, a second indication that the client access 110 to the new active table, NAT 108, is enabled. In some examples, this signals termination of the downtime, enabling the system to be operational for client access 110, thereby making the new active table, old NAT 108, available for client access 110.

[0026] Generally, after validating that data is successfully archived, the new NAT 108 and the IT 106 may be truncated. In some examples, such truncation may be performed manually, or via a separate cleanup operation. A bidirectional handshake protocol may be utilized to confirm truncation of data. Although additional storage space may be utilized during runtime, such storage space may be reclaimed after completion of the data archival. In some examples, a storage capacity load on a main system may increase while archival operations are being performed; however, another downtime may generally not be required to perform administrative activities like rebuilding of indexes and tables.

[0027] System 100 may be implemented by a computing device. As used herein, a computing device may be a desktop computer, laptop (or notebook) computer, workstation, tablet computer, mobile phone, smart device, switch, router, server, blade enclosure, or any other processing device or equipment including a processing resource. In examples described herein, a processing resource may include, for example, one processor or multiple processors included in a single computing device or distributed across multiple computing devices. The components of system 100 (e.g., 102, 104) may be any combination of hardware and programming to implement the functionalities described herein. In examples described herein, such combinations of hardware and programming may be implemented in a number of different ways. For example, the programming for the components may be processor executable instructions stored on at least one non-transitory machine-readable storage medium and the hardware for the components may include at least one processing resource to execute those instructions. In some examples, the hardware may also include other electronic circuitry to at least partially implement at least one component of system 100. In some examples, the at least one machine-readable storage medium may store instructions that, when executed by the at least one processing resource, at least partially implement some or all of components 102, and 104 of system 100. In such examples, system 100 may include the at least one machine-readable storage medium storing the instructions and the at least one processing resource to execute the instructions. In other examples, the functionalities of any components of system 100 may be at least partially implemented in the form of electronic circuitry.

[0028] For example, the deployer 102 may be a combination of hardware and programming (e.g., processor executable instructions). For example, the programming of deployer 102 may include instructions executable to create an intermediate table to record, during data archival, changes to the data to be archived. Also, for example, the programming of deployer 102 may include instructions executable to create triggers on the AT and the NAT to facilitate the record, by the IT, of the changes to the data to be archived. As another example, the deployer 102 may include hardware to physically store, for

example, the triggers for the active tables, intermediate tables, and non-active tables. Also, for example, the deployer 102 may include a combination of hardware and software programming to dynamically interact with the other components of system 100.

[0029] Likewise, the archiver 110 may be a combination of hardware and programming (e.g., processor executable instructions) to initiate data archival by archiving the copy of the data to be archived from the NAT. For example, the programming of archiver 110 may include instructions executable to merge the recorded data from the IT to the NAT upon receiving an indication that the client access to the AT is not enabled. Also, for example, archiver 110 may include hardware to physically store, for example, instructions to switch the client access from the AT to the NAT by changing a table synonym, where the client access to the NAT is enabled upon completion of the data archival. The archiver 110 may include a combination of hardware and software programming to dynamically interact with the other components of system 100.

[0030] Generally, as described herein, the components of system 100 may include software programming and physical networks to be communicatively linked to other components of system 100. In some instances, the components of system 100 may include a processor and a memory, while programming code is stored and on that memory and executable by a processor to perform designated functions.

[0031] A computing device, as used herein, may be, for example, a web-based server, a local area network server, a cloud-based server, a notebook computer, a desktop computer, an all-in-one system, a tablet computing device, a mobile phone, an electronic book reader, or any other electronic device suitable for provisioning a computing resource to perform a unified visualization interface. The computing device may include a processor and a computer-readable storage medium.

[0032] Figure 2 is a block diagram illustrating one example of a computer readable medium for structured data archival with reduced downtime.

Processing system 200 includes a processor 202, a computer readable medium 208, input device 204, and output device 206. Processor 202, computer

readable medium 208, input device 204, and output device 206 are coupled to each other through a communication link (e.g., a bus).

[0033] Processor 202 executes instructions included in the computer readable medium 208. Computer readable medium 208 includes table managing instructions 210 to manage an active table (AT), and a non-active table (NAT), wherein client access to the AT is enabled, and wherein the AT includes data to be archived, and wherein the client access to the NAT is not enabled, and wherein the NAT includes a copy of the data to be archived.

[0034] Computer readable medium 208 includes table creation instructions 212 to create an intermediate table (IT) to record, during data archival, changes to the data to be archived.

[0035] Computer readable medium 208 includes trigger creation instructions 214 to create triggers on the AT and the NAT to facilitate the record, by the IT, of the changes to the data to be archived.

[0036] Computer readable medium 208 includes archival initiation instructions 216 to initiate the data archival by archiving the copy of the data to be archived from the NAT.

[0037] Computer readable medium 208 includes trigger enabling instructions 218 to enable the triggers on the AT based on information associated with the AT, wherein the information is retrieved from a metadata table.

[0038] Computer readable medium 208 includes data merging instructions 220 to merge the recorded data from the IT to the NAT upon receiving an indication that the client access to the AT is not enabled.

[0039] Computer readable medium 208 includes client access switching instructions 222 to switch the client access from the AT to the NAT by changing a table synonym, where the client access to the NAT is enabled upon completion of the data archival.

[0040] In some examples, computer readable medium 208 includes instructions to provide a second indication that the client access to the NAT is enabled.

[0041] In some examples, computer readable medium 208 includes instructions to create a new schema for the NAT, and to create the IT based on one of the new schema for the NAT and an existing schema for the AT.

[0042] Input device 204 includes a keyboard, mouse, data ports, and/or other suitable devices for inputting information into processing system 200. In some examples, input device 204 is used to receive, via a computing device, a first indication to initiate data archival. Output device 206 includes a monitor, speakers, data ports, and/or other suitable devices for outputting information from processing system 200. In some examples, output device 206 is used to provide, via a computing device, a second indication that the client access to the NAT is enabled.

[0043] As used herein, a computer readable medium may be any electronic, magnetic, optical, or other physical storage apparatus to contain or store information such as executable instructions, data, and the like. For example, any computer readable storage medium described herein may be any of Random Access Memory (RAM), volatile memory, non-volatile memory, flash memory, a storage drive (e.g., a hard drive), a solid state drive, and the like, or a combination thereof. For example, the computer readable medium 208 can include one of or multiple different forms of memory including semiconductor memory devices such as dynamic or static random access memories (DRAMs or SRAMs), erasable and programmable read-only memories (EPROMs), electrically erasable and programmable read-only memories (EEPROMs) and flash memories; magnetic disks such as fixed, floppy and removable disks; other magnetic media including tape; optical media such as compact disks (CDs) or digital video disks (DVDs); or other types of storage devices.

[0044] As described herein, various components of the processing system 200 are identified and refer to a combination of hardware and programming configured to perform a designated visualization function. As illustrated in Figure 2, the programming may be processor executable instructions stored on tangible computer readable medium 208, and the hardware may include processor 202 for executing those instructions. Thus, computer readable medium 208 may store program instructions that, when executed by processor 202, implement the various components of the processing system 200.

[0045] Such computer readable storage medium or media is (are) considered to be part of an article (or article of manufacture). An article or article of

manufacture can refer to any manufactured single component or multiple components. The storage medium or media can be located either in the machine running the machine-readable instructions, or located at a remote site from which machine-readable instructions can be downloaded over a network for execution.

[0046] Computer readable medium 208 may be any of a number of memory components capable of storing instructions that can be executed by Processor 202. Computer readable medium 208 may be non-transitory in the sense that it does not encompass a transitory signal but instead is made up of one or more memory components configured to store the relevant instructions. Computer readable medium 208 may be implemented in a single device or distributed across devices. Likewise, processor 202 represents any number of processors capable of executing instructions stored by computer readable medium 208. Processor 202 may be integrated in a single device or distributed across devices. Further, computer readable medium 208 may be fully or partially integrated in the same device as processor 202 (as illustrated), or it may be separate but accessible to that device and processor 202. In some examples, computer readable medium 208 may be a machine-readable storage medium.

[0047] Figure 3 is a flow diagram illustrating one example of a method for structured data archival with reduced downtime. In some examples, such an example method may be implemented by a system such as, for example, system 100 of Figure 1. In some examples, such an example method may be implemented in the form of executable instructions stored on a machine-readable storage medium, such as storage medium 208, and/or in the form of electronic circuitry.

[0048] At 300, an active table (AT) and a non-active table (NAT) are managed, where client access to the AT is enabled, and where the AT includes data to be archived, and where the client access to the NAT is not enabled, and where the NAT includes a copy of the data to be archived.

[0049] At 302, an intermediate table (IT) is created to record, during data archival, changes to the data to be archived.

[0050] At 304, triggers are created on the AT and the NAT to facilitate the record, by the IT, of the changes to the data to be archived.

[0051] At 306, the data archival is initiated by archiving the copy of the data to be archived from the NAT.

[0052] At 308, the recorded data from the IT is merged to the NAT upon receiving an indication that the client access to the AT is not enabled.

[0053] At 310, the client access is switched from the AT to the NAT by changing a table synonym, where the client access to the NAT is enabled upon completion of the data archival.

[0054] At 312, a second indication is provided that the client access to the NAT is enabled.

[0055] Figure 4 is a flow diagram illustrating one example of a method for creating an intermediate table for structured data archival with reduced downtime. In some examples, such an example method may be implemented by a system such as, for example, system 100 of Figure 1. In some examples, such an example method may be implemented in the form of executable instructions stored on a machine-readable storage medium, such as storage medium 208, and/or in the form of electronic circuitry.

[0056] At 400, an active table (AT) and a non-active table (NAT) are managed, where client access to the AT is enabled, and where the AT includes data to be archived, and where the client access to the NAT is not enabled, and where the NAT includes a copy of the data to be archived.

[0057] At 402, an intermediate table (IT) is created to record, during data archival, changes to the data to be archived.

[0058] At 404, triggers are created on the AT and the NAT to facilitate the record, by the IT, of the changes to the data to be archived.

[0059] At 406, the data archival is initiated by archiving the copy of the data to be archived from the NAT.

[0060] At 408, the recorded data from the IT is merged to the NAT upon receiving an indication that the client access to the AT is not enabled.

[0061] At 410, the client access is switched from the AT to the NAT by changing a table synonym, where the client access to the NAT is enabled upon completion of the data archival.

[0062] At 412, a second indication is provided that the client access to the NAT is enabled.

[0063] At 414, a new schema is created for the NAT.

[0064] At 416, the IT is created based on one of the new schema for the NAT and an existing schema for the AT.

[0065] Figure 5 is a flow diagram illustrating one example of a method for initiating data archival for structured data archival with reduced downtime. In some examples, such an example method may be implemented by a system such as, for example, system 100 of Figure 1. In some examples, such an example method may be implemented in the form of executable instructions stored on a machine-readable storage medium, such as storage medium 208, and/or in the form of electronic circuitry.

[0066] At 500, an active table (AT) and a non-active table (NAT) are managed, where client access to the AT is enabled, and where the AT includes data to be archived, and where the client access to the NAT is not enabled, and where the NAT includes a copy of the data to be archived.

[0067] At 502, an intermediate table (IT) is created to record, during data archival, changes to the data to be archived.

[0068] At 504, triggers are created on the AT and the NAT to facilitate the record, by the IT, of the changes to the data to be archived.

[0069] At 506, the data archival is initiated by archiving the copy of the data to be archived from the NAT.

[0070] At 508, the recorded data from the IT is merged to the NAT upon receiving an indication that the client access to the AT is not enabled.

[0071] At 510, the client access is switched from the AT to the NAT by changing a table synonym, where the client access to the NAT is enabled upon completion of the data archival.

[0072] At 512, a second indication is provided that the client access to the NAT is enabled.

[0073] At 514, information associated with the AT and the NAT is recorded in a metadata table.

[0074] At 516, the information associated with the AT is identified from the metadata table.

[0075] At 518, the triggers on the AT are enabled based on the identified information.

[0076] Figure 6 is a flow diagram illustrating one example of a method for creating triggers on active and non-active tables. In some examples, such an example method may be implemented by a system such as, for example, system 100 of Figure 1. In some examples, such an example method may be implemented in the form of executable instructions stored on a machine-readable storage medium, such as storage medium 208, and/or in the form of electronic circuitry.

[0077] At 600, the example flow may be started.

[0078] At 602, a new schema may be created for a non-active table (NAT).

[0079] At 604, a new or existing schema may be selected for an intermediate table (IT).

[0080] At 606, the new schema for the IT may be selected based on the new schema for the NAT.

[0081] At 608, the new schema for the IT may be selected based on the existing schema for the active table (AT).

[0082] At 610, triggers may be created for the ATs and NATs, including insertion, update, and/or deletion of triggers so that new events, occurring during active archival, may be recorded on the IT.

[0001] At 612, triggers may be disabled on the ATs and the NATs. The triggers may be kept disabled until active data archival is initiated.

[0002] At 614, metadata information about the ATs and NATs may be stored, for example, in metadata tables.

[0003] At 616, the process flow may be terminated.

[0004] Figure 7 is a flow diagram illustrating another example of a method for structured data archival with reduced downtime. In some examples, such an example method may be implemented by a system such as, for example,

system 100 of Figure 1. In some examples, such an example method may be implemented in the form of executable instructions stored on a machine-readable storage medium, such as storage medium 208, and/or in the form of electronic circuitry.

[0005] At 700, the process may be started.

[0006] At 702, active tables (ATs) may be identified.

[0007] At 704, triggers may be enabled on the ATs.

[0008] At 706, a copy of data on the ATs may be made onto the non-active tables (NATs).

[0009] At 708, data may be archived from the NATs based on retention rules and policies.

[0010] At 710, a determination may be made if the archival process is to continue.

[0011] At 712, upon a determination that the archival process is not to continue, a download time may be specified and/or scheduled.

[0012] At 714, the process may be paused to wait for an indication that the client access to the AT is not enabled, i.e., an indication that a downtime has been initiated.

[0013] At 716, a background process may be run to determine a state of the downtime period.

[0014] At 718, it is determined if the downtime period has started. Upon a determination that the downtime period has not started, the process returns to a pause state at 714.

[0015] At 718, upon a determination that the downtime period has started, the process moves to 720.

[0016] At 710, upon a determination that the archival process is to continue, the process moves to 720.

[0017] At 720, the system downtime is started.

[0018] At 722, data is merged from the ITs to the NATs.

[0019] At 724, NATs are swapped with ATs (e.g., by changing a table synonym for the NATs).

[0020] At 726, metadata is updated for the new ATs.

[0021] At 728, a notification indicative of archive completion is generated.

[0022] At 730, the process may be terminated.

[0023] Examples of the disclosure provide a generalized system for structured data archival with reduced downtime. The generalized system significantly reduces capital expenses and associated administrative costs, improves business application users' productivity and overall IT staff efficiency, enables responses to legal and compliance requests quickly and effectively, and allows data to be leveraged to its highest value. Upon completion of the data archival process as described herein, the production database becomes smaller in size and may be maintained more efficiently.

[0024] Although specific examples have been illustrated and described herein, a variety of alternate and/or equivalent implementations may be substituted for the specific examples shown and described without departing from the scope of the present disclosure. This application is intended to cover any adaptations or variations of the specific examples discussed herein.

CLAIMS

- 1 1. A system for structured data archival with reduced downtime comprising:
2 a deployer to:
3 manage an active table (AT), and a non-active table (NAT),
4 wherein client access to the AT is enabled, and wherein the AT includes
5 data to be archived, and wherein the client access to the NAT is not
6 enabled, and wherein the NAT includes a copy of the data to be
7 archived,
8 create an intermediate table (IT) to record, during data
9 archival, changes to the data to be archived, and
10 create triggers on the AT and the NAT to facilitate the
11 record, by the IT, of the changes to the data to be archived; and
12 an archiver to:
13 initiate the data archival by archiving the copy of the data to
14 be archived from the NAT,
15 merge the recorded data from the IT to the NAT upon
16 receiving an indication that the client access to the AT is not enabled,
17 and
18 switch the client access from the AT to the NAT by
19 changing a table synonym, wherein the client access to the NAT is
20 enabled upon completion of the data archival.
- 1 2. The system of claim 1, wherein the archiver is to provide a second
2 indication that the client access to the NAT is enabled.
- 1 3. The system of claim 1, wherein the deployer is to:
2 create a new schema for the NAT; and
3 create the IT based on one of the new schema for the NAT and an
4 existing schema for the AT.

1 4. The system of claim 1, wherein the triggers include one of insert, update,
2 and delete triggers.

1 5. The system of claim 1, wherein the archiver is to initiate the data archival
2 by enabling the triggers.

1 6. The system of claim 1, wherein the deployer is to record information
2 associated with the AT and the NAT in a metadata table.

1 7. The system of claim 6, wherein the archiver is to identify the information
2 associated with the AT from the metadata table.

1 8. A method for structured data archival with reduced downtime
2 comprising:
3 managing an active table (AT), and a non-active table (NAT),
4 wherein client access to the AT is enabled, and wherein the AT includes
5 data to be archived, and wherein the client access to the NAT is not
6 enabled, and wherein the NAT includes a copy of the data to be
7 archived;
8 creating an intermediate table (IT) to record, during data archival,
9 changes to the data to be archived;
10 creating triggers on the AT and the NAT to facilitate the record, by
11 the IT, of the changes to the data to be archived;
12 initiating the data archival by archiving the copy of the data to be
13 archived from the NAT;
14 merging the recorded data from the IT to the NAT upon receiving
15 an indication that the client access to the NAT is not enabled;
16 switching the client access from the AT to the NAT by changing a
17 table synonym, wherein the client access to the NAT is enabled upon
18 completion of the data archival; and
19 providing a second indication that the client access to the NAT is
20 enabled.

- 1 9. The method of claim 8, comprising:
2 creating a new schema for the NAT; and
3 creating the IT based on one of the new schema for the NAT and
4 an existing schema for the AT.
- 1 10. The method of claim 8, comprising enabling the triggers upon the
2 initiating of the data archival.
- 1 11. The method of claim 10, comprising:
2 recording information associated with the AT and the NAT in a
3 metadata table;
4 identifying the information associated with the AT from the
5 metadata table; and
6 enabling the triggers on the AT based on the identified information.
- 1 12. The method of claim 8, wherein the triggers include one of insert, update,
2 and delete triggers.
- 1 13. A non-transitory computer readable medium for structured data archival
2 with reduced downtime, comprising executable instructions to:
3 manage an active table (AT), and a non-active table (NAT),
4 wherein client access to the AT is enabled, and wherein the AT includes
5 data to be archived, and wherein the client access to the NAT is not
6 enabled, and wherein the NAT includes a copy of the data to be
7 archived;
8 create an intermediate table (IT) to record, during data archival,
9 changes to the data to be archived;
10 create triggers on the AT and the NAT to facilitate the record, by
11 the IT, of the changes to the data to be archived;
12 initiate the data archival by archiving the copy of the data to be
13 archived from the NAT;

14 enable the triggers on the AT based on information associated
15 with the AT, wherein the information is retrieved from a metadata table;
16 merge the recorded data from the IT to the NAT upon receiving an
17 indication that the client access to the AT is not enabled; and
18 switch the client access from the AT to the NAT by changing a
19 table synonym, wherein the client access to the NAT is enabled upon
20 completion of the data archival.

1 14. The computer readable medium of claim 13, comprising instructions to
2 provide a second indication that the client access to the NAT is enabled.

1 15. The computer readable medium of claim 13, comprising instructions to:
2 create a new schema for the NAT; and
3 create the IT based on one of the new schema for the NAT and an
4 existing schema for the AT.

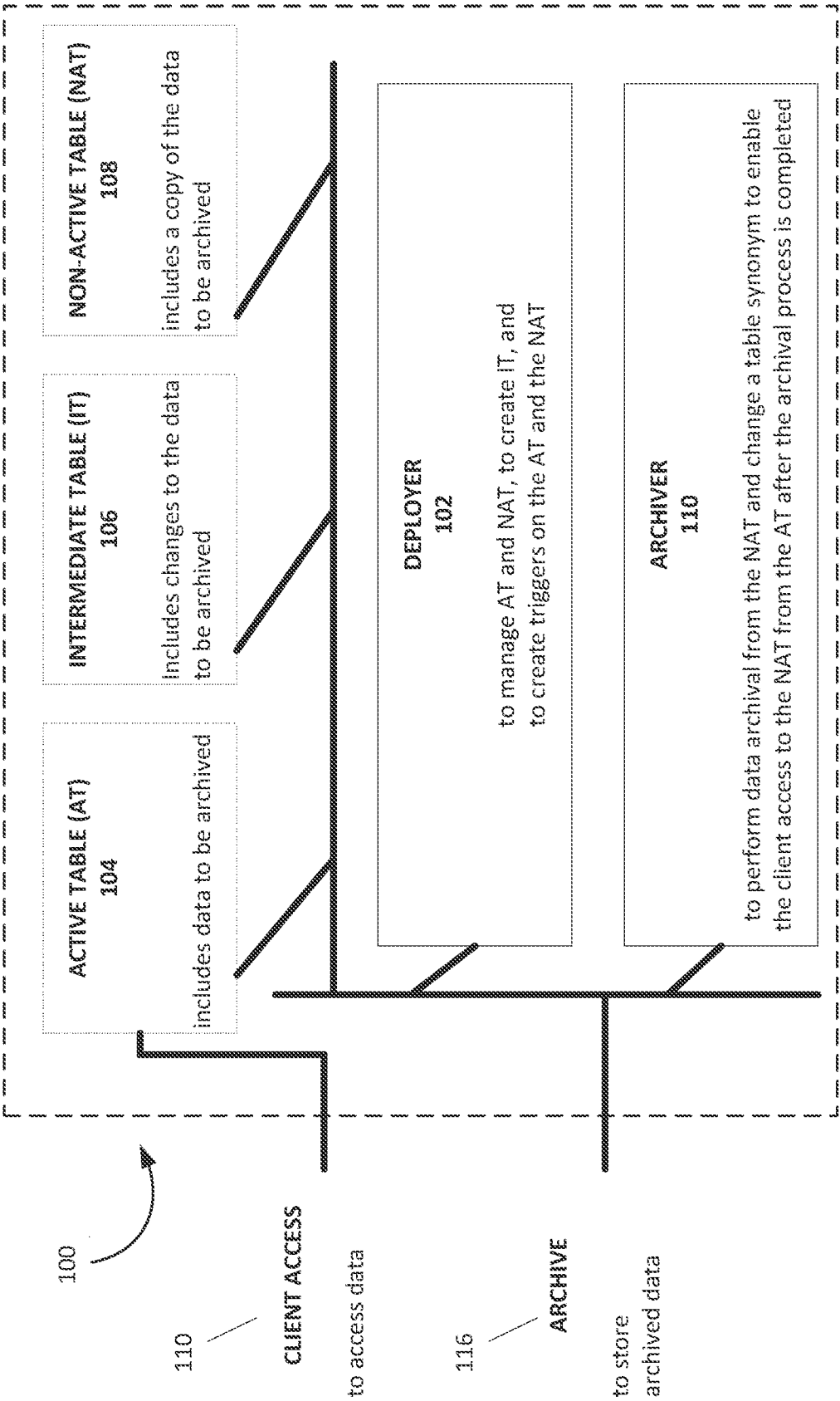


Fig. 1

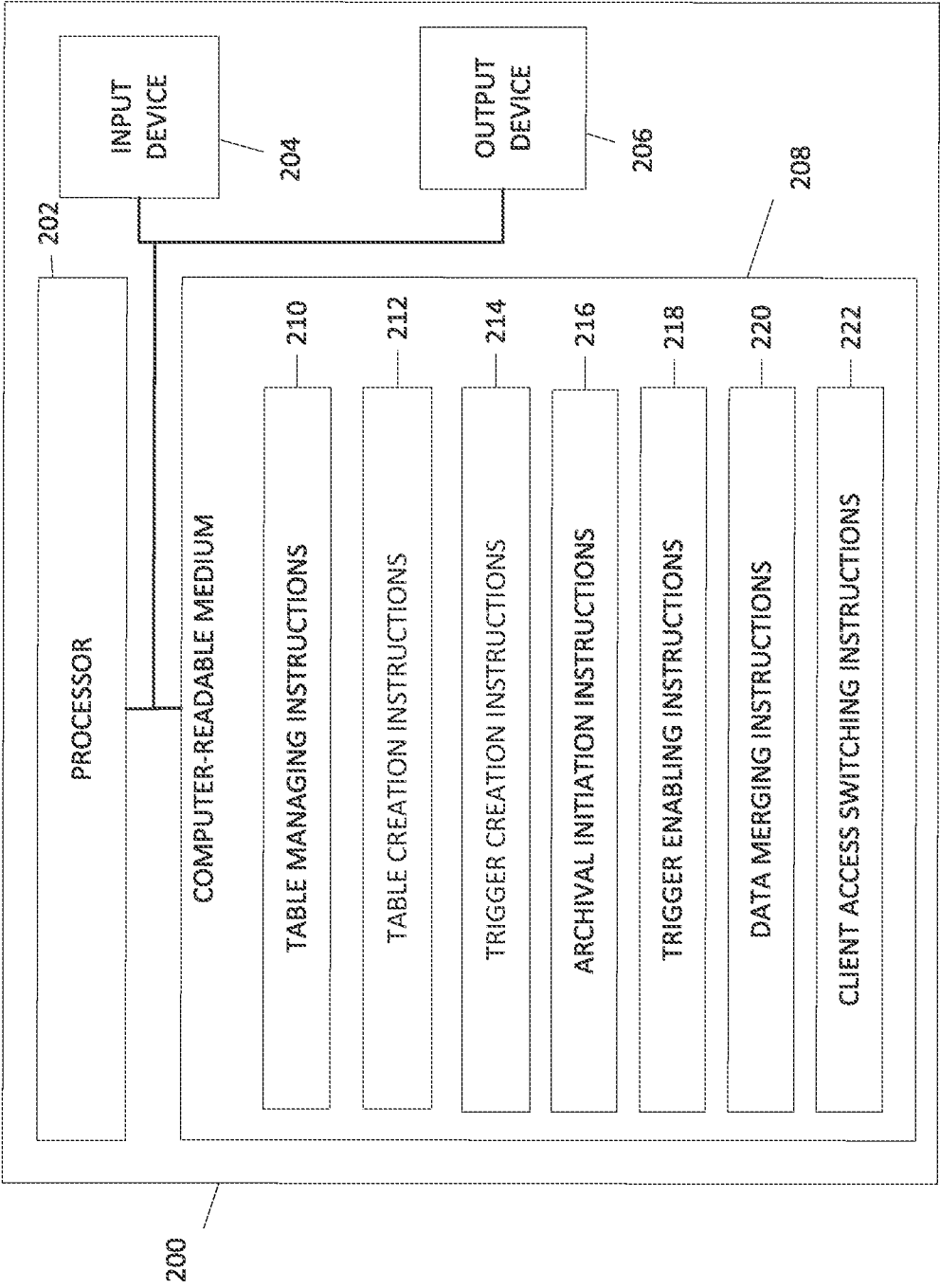


Fig. 2

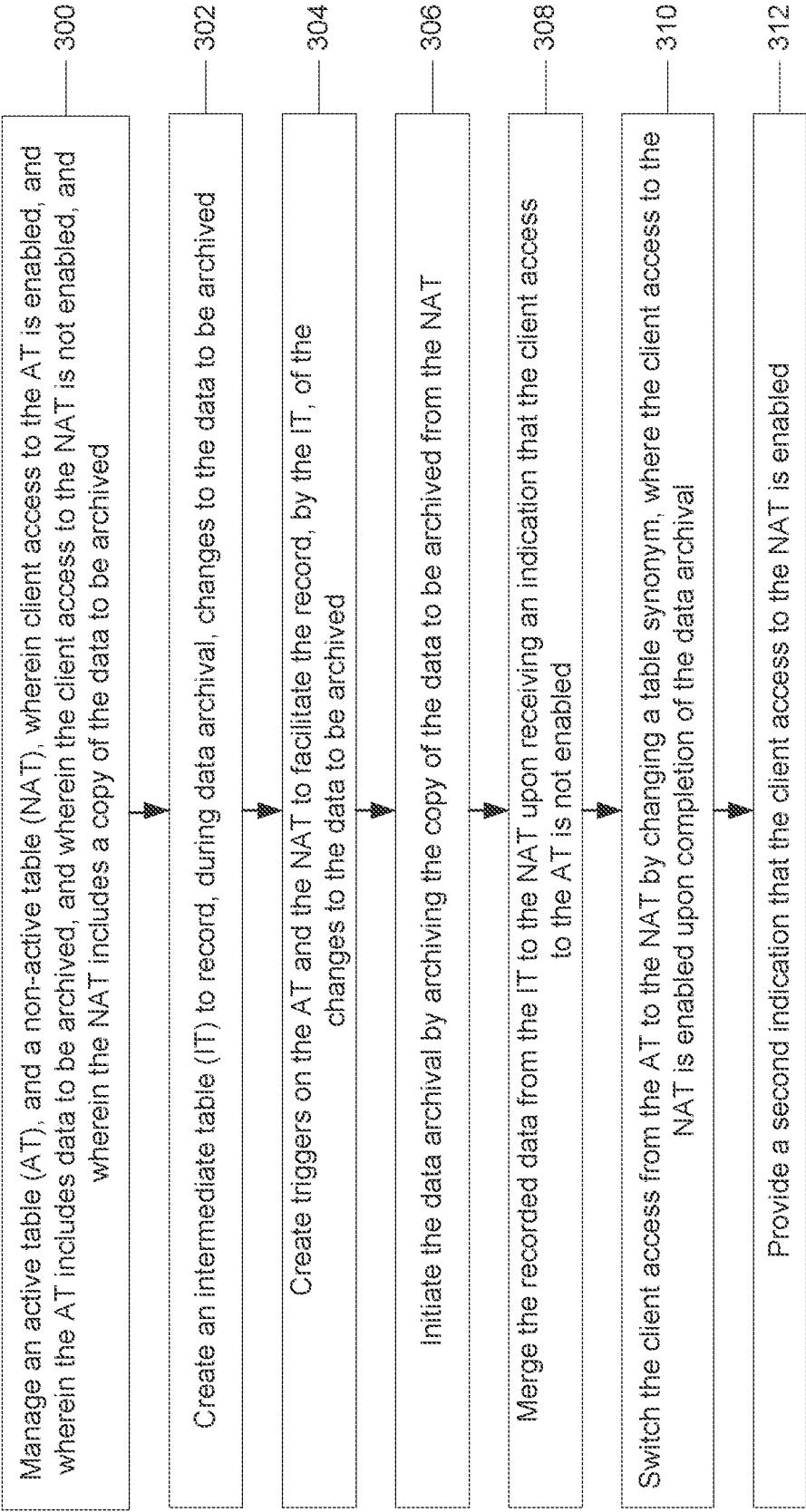


Fig. 3

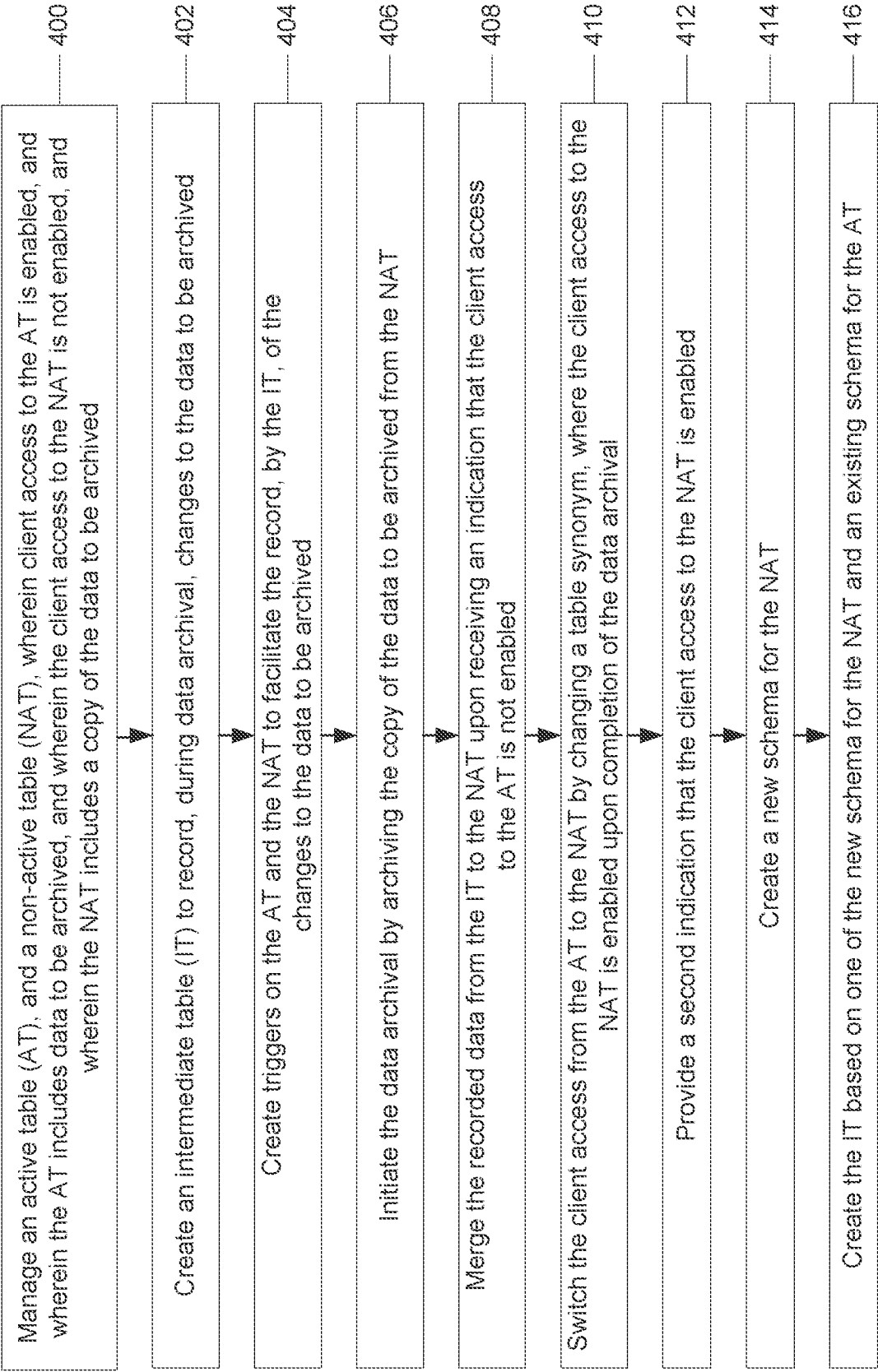


Fig. 4

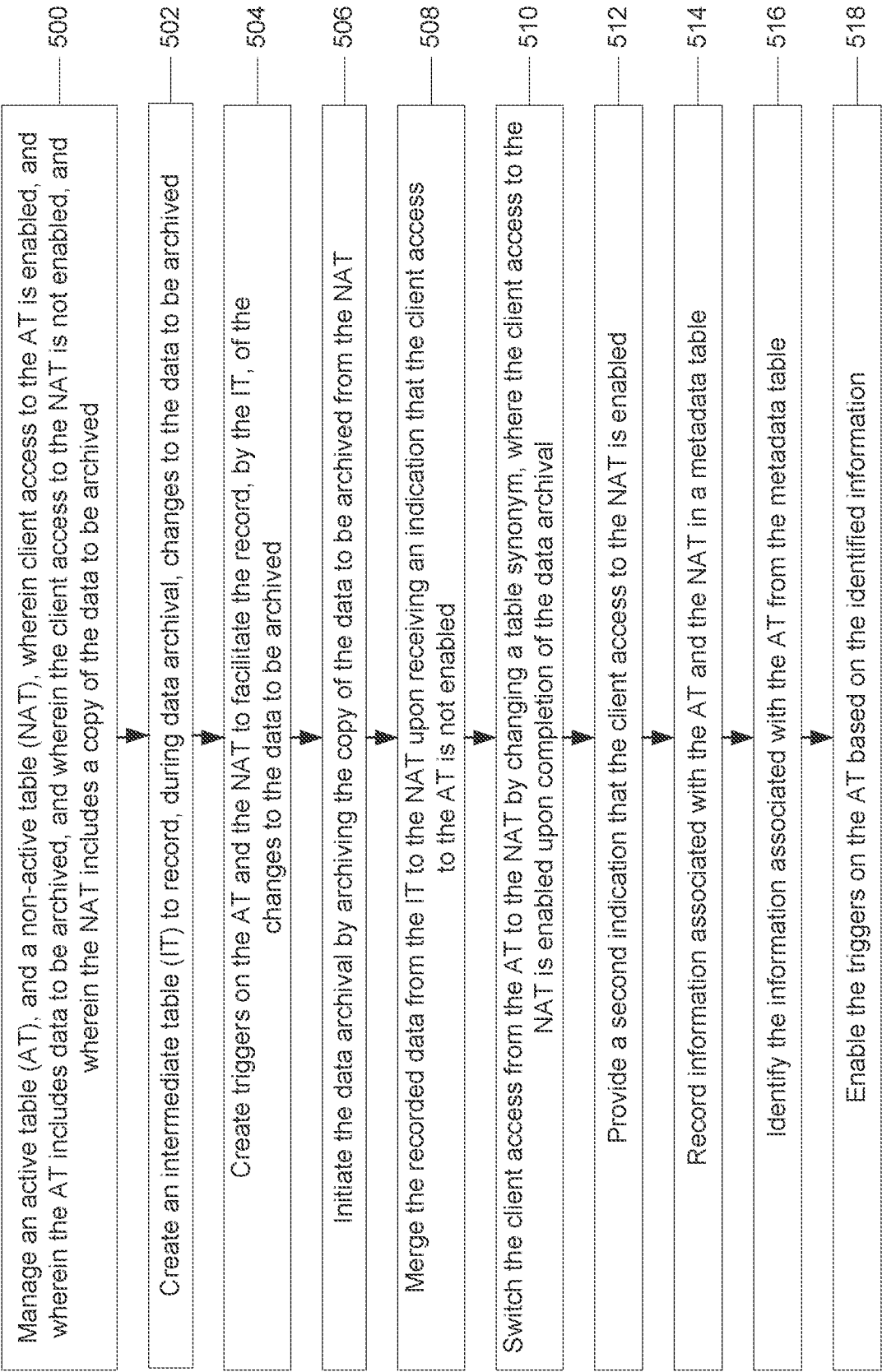


Fig. 5

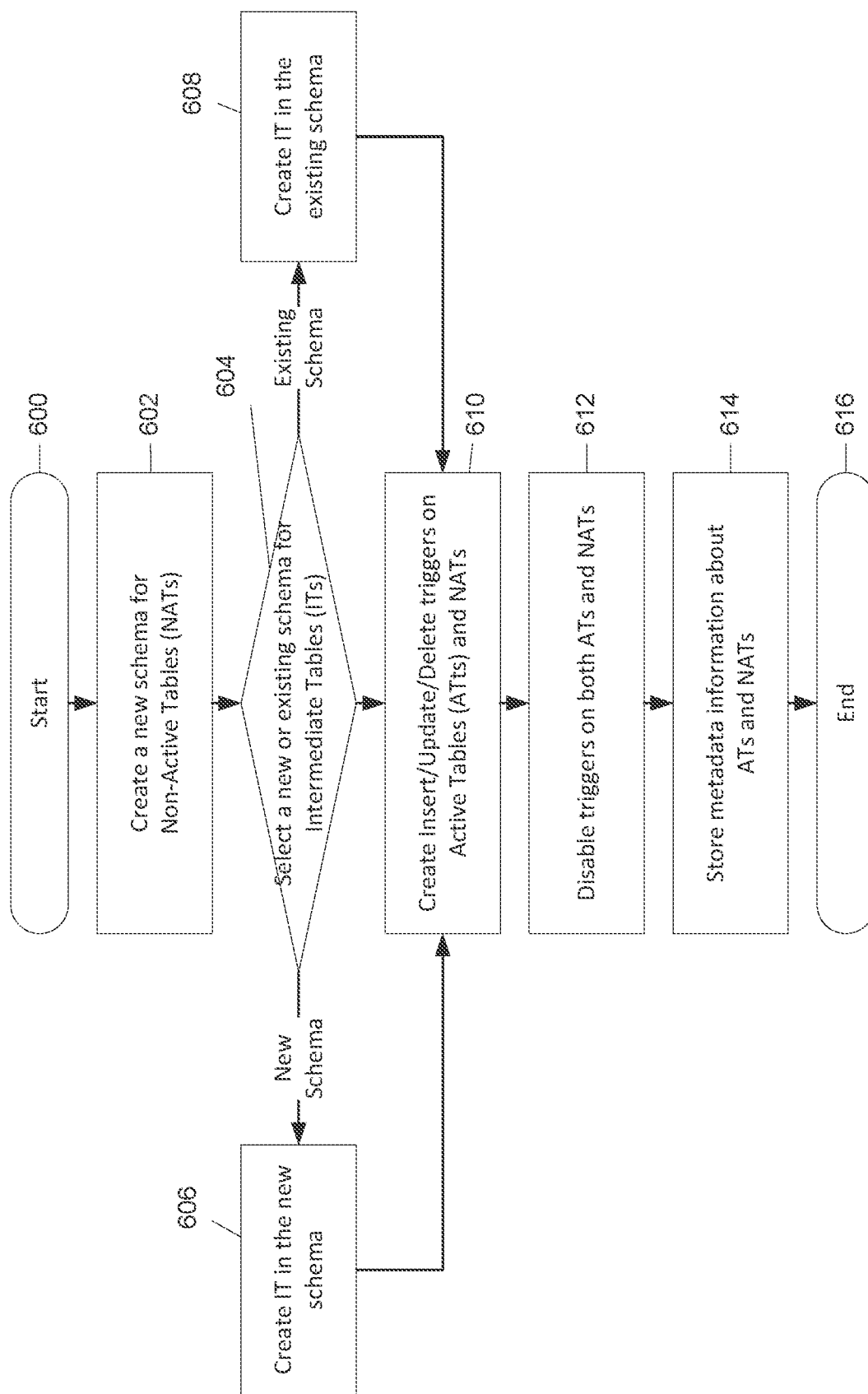


Fig. 6

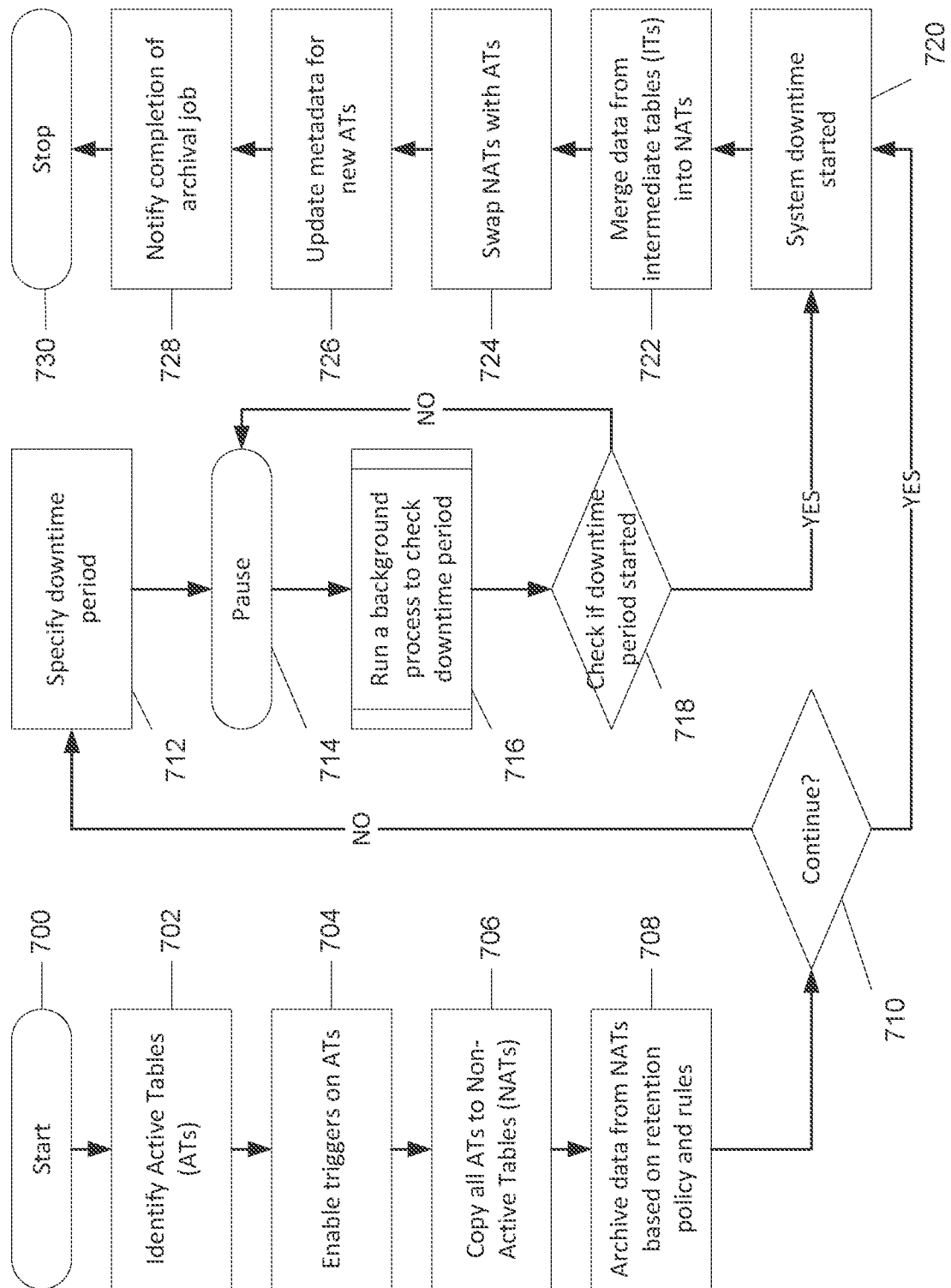


Fig. 7

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2016/026790**A. CLASSIFICATION OF SUBJECT MATTER****G06F 17/40(2006.01)i, G06F 3/06(2006.01)i**

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 17/40; G06F 15/16; G06F 12/16; G06F 17/30; G06F 11/08; G06F 3/06

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS(KIPO internal) & Keywords: archive, record, change, reduce, merge, copy, switch, and similar terms.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2009-0157775 A1 (MAY PEDERSON et al.) 18 June 2009 See paragraphs [0008], [0014], and [0029]; and figure 1.	1-15
A	US 2004-0210608 A1 (HOWARD F. LEE et al.) 21 October 2004 See paragraphs [0024], [0037], [0060], [0136], and [0225]; and figure 2.	1-15
A	WO 2015-183316 A1 (HEWLETT-PACKARD DEVELOPMENT COMPANY, L. P.) 03 December 2015 See paragraphs [0013], [0030], [0039]-[0040], and [0048]; and figure 2.	1-15
A	US 2007-0033252 A1 (RICKY F. COMBEST) 08 February 2007 See paragraphs [0016], [0018], and [0079]; and figure 2.	1-15
A	US 2007-0112892 A1 (JOHN TERRY LANGLEY et al.) 17 May 2007 See paragraphs [0007], [0023], and [0053]; and figure 1.	1-15



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

24 November 2016 (24.11.2016)

Date of mailing of the international search report

24 November 2016 (24.11.2016)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

NHO, Ji Myong

Telephone No. +82-42-481-8528



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2016/026790

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2009-0157775 A1	18/06/2009	US 8452730 B2	28/05/2013
US 2004-0210608 A1	21/10/2004	US 7155465 B2	26/12/2006
WO 2015-183316 A1	03/12/2015	None	
US 2007-0033252 A1	08/02/2007	None	
US 2007-0112892 A1	17/05/2007	US 2008-313245 A1	18/12/2008
		US 7433902 B2	07/10/2008
		US 8015155 B2	06/09/2011