December 2022

METHOD AND SYSTEM FOR BI CUBE REFRESH AUTOMATION

PRADEEP KUMAR M K
Visa

VELKUMAR THANGAVEL
Visa

ABINAYA SOLAIYAN
Visa

CHIA-CHIH LU
Visa

KUMAR REDDY
Visa

TITLE: “METHOD AND SYSTEM FOR BI CUBE REFRESH AUTOMATION”

VISA

PRADEEP KUMAR M K

VELKUMAR THANGAVEL

ABINAYA SOLAIYAN

CHIA-CHIH LU

KUMAR REDDY

TECHNICAL FIELD

[0001] This disclosure relates generally to the field of data structures such as data cubes. More particularly, the present disclosure relates to a method and system for Business Intelligence (BI) cube refresh automation.

BACKGROUND

[0002] In the existing technologies, large amount of data is being generated in order of millions of terabytes. Datasets may be generated in many areas such as meteorology, genomics, connectomics, complex physics simulations, biological and environmental research and the like. However, such big data cannot be stored in relational database management systems. Thus, there is a need for an efficient database which can handle and store large amount of data. One such database is a data cube which can store a large amount of data in a simple manner.

[0003] Data cube is a muti-dimensional array of values used to bring together the data to be organized and modeled for analysis. In data cubes the relevant data is consolidated and is stored in the cubes. Also, a cube is a section of data which is built from tables in a database that contains calculations. The data cubes provide a faster response to the queries since all the relevant data are aggregated in a cube. However, in data cube the creation and the modification of a data requires a new report each time which is a time-consuming process.

[0004] The current data cube system lacks the capability to refresh the data cubes and to check cube health status. The data cubes are refreshed manually every month to show the latest data to the users, which is a very tedious and a time-consuming process. The refresh process performed manually may take around 16-18 hours to complete the refresh process. Particularly, in the manual refresh process every month to show the latest data to the users, the old month data in the cube is deleted and the new month data is added. In the manual refresh process a continuous monitoring of the cubes is required to check the cube status and its dependency. Thus, the delay caused in the manual refresh process may impact the business, and additionally the users may not even be able to see the latest data on time which may lead to Service Level Agreement (SLA) breach.

[0005] One of the existing systems provides a system for monitoring performance of computing systems. The monitoring system monitors the performance of the computing systems. Initially the set of tasks provided for the server to perform are identified. Further, the performance of the server in executing the tasks provided is monitored. Finally, an evaluation is performed whether the conditions are satisfied for adjusting one or more operating parameters of the server system based on the measured performance metrics.

[0006] Thus, there is a need for a method and system which provides an improved BI cube refresh process.

SUMMARY

[0007] The present disclosure relates to a method and system for Business Intelligence (BI) cube refresh automation. In the present disclosure, initially the cube refresh automation system monitors an Extract Transform Load (ETL) Load on the data cube and triggers a cube refresh process upon the ETL determination. Further, status of I-server is checked. Once the I-server status is determined to be positive, a back-up copy of workflow logs of the previous month in the data cubes are maintained in a separate file, and thus by deleting previous month workflow logs and inserting the new month workflow logs in the data cubes. The insertion of the new month workflow logs is split across clustered servers to balance the load. Upon the insertion, the results of the insertion process in one or more clusters are merged together. Further, the status of the insertion is monitored based on the obtained insertion results. The status of the insertion may be successful (Insertion of workflow logs is successfully performed) or unsuccessful (Failure of insertion operation). Furthermore, the cube refresh automation system deletes the triggers for the successful insertion. If an insertion is unsuccessful, the cube refresh automation system triggers the insertion operation for the second time. If the insertion operation is successful at the second time, the trigger for the insertion is deleted. In case, the deletion of the trigger is unsuccessful, the deletion of the trigger is triggered by the cube refresh automation system for the second time. Finally, upon successful insertion of new month workflow logs and deletion of the triggers, the cube refresh automation system sends the information inserted in the data cubes and the cube health status to the users via an email, message and the like.

[0008] In the present research work, system and method provide an improved BI cube refresh process. In the present disclosure, the cube refresh process is automated, therefore, the data cubes are refreshed automatically when the latest data is available in the tables. Further, a notification will be triggered for all the success or failure activities on the data cubes, which provides a health status of the data cubes. In some embodiments, the notification to the users may be provided in the form of emails, messages and the like. In the present research work, the refresh process will automatically add the latest month's data and delete the previous month's data inside the data cubes. Upon completion of the refresh process, the cube refresh automation system may send the notification about the latest data available in the data cubes to users, supporting team and the like. Additionally, the automated refresh process is a continuous process which saves nearly 16-18 hours of manual intervention, thereby enhancing efficiency of the BI cube refresh process.

[0009] In the present disclosure, the cube refresh automation system may automatically insert or delete monthly data into the data cubes after the data load event triggers. The cube refresh automation system is capable enough to run by itself to execute the incremental refresh report or decremental refresh reports.

[0010] These and other features and characteristics of the present invention, as well as the methods of operation and functions of the related elements of structures and the combination of parts and economies of manufacture, will become more apparent upon consideration of the following description with reference to the accompanying drawings, all of which form a part of this specification, wherein like reference numerals designate corresponding parts in the various figures. It is to be expressly understood, however, that the drawings are for the purpose of illustration and description only and are not intended as a definition of the limits of the invention. As used in the specification, the singular form of "a," "an," and "the" include plural referents unless the context clearly dictates otherwise.

BRIEF DESCRIPTION OF THE DRAWINGS AND APPENDICES

[0011] Additional advantages and details of non-limiting embodiments are explained in greater detail below with reference to the exemplary embodiments that are illustrated in the accompanying schematic figures, in which:

[0012] FIG.1 shows an exemplary architecture for performing Business Intelligence (BI) cube refresh automation, in accordance with some embodiments of the present disclosure;

[0013] FIG.2 shows a flow diagram that illustrates a method of facilitating BI cube refresh automation, in accordance with some embodiments of the present disclosure;

[0014] FIG.3 shows an exemplary scenario of a cube refresh automation process, in accordance with some embodiments of the present disclosure;

[0015] FIG. 4A and FIG. 4B show exemplary graphs illustrating product comparisons based on card count and drafts respectively, as indicated in exemplary insert logs in accordance with some embodiments of the present disclosure;

[0016] FIG. 4C and FIG. 4D show exemplary graphs illustrating Risk score auth and risk score fraud respectively, as indicated in the exemplary insert logs in accordance with some embodiments of the present disclosure;

[0017] FIG. 4E and FIG. 4F show exemplary graphs illustrating Direct AFT and Direct OCT as indicated in the exemplary insert logs in accordance with some embodiments of the present disclosure;

[0018] FIG. 4G and FIG. 4H show exemplary graphs illustrating authorization approvals as indicated in the exemplary delete logs in accordance with some embodiments of the present disclosure; and

[0019] FIG.4I shows an exemplary graph illustrating currency validation as indicated in the exemplary delete logs in accordance with some embodiments of the present disclosure.

DESCRIPTION OF THE DISCLOSURE

[0020] In the present document, the word "exemplary" is used herein to mean "serving as an example, instance, or illustration." Any embodiment or implementation of the present subject

matter described herein as "exemplary" is not necessarily to be construed as preferred or advantageous over other embodiments.

[0021] While the disclosure is susceptible to various modifications and alternative forms, specific embodiment thereof has been shown by way of example in the drawings and will be described in detail below. It should be understood, however that it is not intended to limit the disclosure to the particular forms disclosed, but on the contrary, the disclosure is to cover all modifications, equivalents, and alternative falling within the spirit and the scope of the disclosure.

[0022] The terms “comprises”, “comprising”, or any other variations thereof, are intended to cover a non-exclusive inclusion, such that a setup, device or method that comprises a list of components or steps does not include only those components or steps but may include other components or steps not expressly listed or inherent to such setup or device or method. In other words, one or more elements in a device or system or apparatus proceeded by “comprises… a” does not, without more constraints, preclude the existence of other elements or additional elements in the device or system or apparatus.

[0023] The terms "an embodiment", "embodiment", "embodiments", "the embodiment", "the embodiments", "one or more embodiments", "some embodiments", and "one embodiment" mean "one or more (but not all) embodiments of the invention(s)" unless expressly specified otherwise.

[0024] The terms "including", "comprising", “having” and variations thereof mean "including but not limited to", unless expressly specified otherwise.

[0025] For purposes of the description hereinafter, it is to be understood that the invention may assume various alternative variations and step sequences, except where expressly specified to the contrary. It is also to be understood that the specific devices and processes illustrated in the attached drawings, and described in the following specification, are simply exemplary embodiments or aspects of the invention. Hence, specific dimensions and other physical characteristics related to the embodiments or aspects disclosed herein are not to be considered as limiting.

[0026] As used herein, the terms "communication" and "communicate" may refer to the reception, receipt, transmission, transfer, provision, and/or the like of information (e.g., data, signals, messages, instructions, commands, and/or the like). For one unit (e.g., a device, a system, a component of a device or system, combinations thereof, and/or the like) to be in communication with another unit means that the one unit is able to directly or indirectly receive information from and/or transmit information to the other unit. This may refer to a direct or indirect connection (e.g., a direct communication connection, an indirect communication connection, and/or the like) that is wired and/or wireless in nature. Additionally, two units may be in communication with each other even though the information transmitted may be modified, processed, relayed, and/or routed between the first and second unit. For example, a first unit may be in communication with a second unit even though the first unit passively receives information and does not actively transmit information to the second unit. As another example, a first unit may be in communication with a second unit if at least one intermediary unit (e.g., a third unit located between the first unit and the second unit) processes information received from the first unit and communicates the processed information to the second unit. In some non-limiting embodiments, a message may refer to a network packet (e.g., a data packet and/or the like) that includes data. It will be appreciated that numerous other arrangements are possible.

[0027] As used herein, the term "computing device" may refer to one or more electronic devices that are configured to directly or indirectly communicate with or over one or more networks. A computing device may be a mobile or portable computing device, a desktop computer, a server, and/or the like. Furthermore, the term "computer" may refer to any computing device that includes the necessary components to receive, process, and output data, and normally includes a display, a processor, a memory, an input device, and a network interface. A "computing system" may include one or more computing devices or computers. An "application" or "Application Program Interface" (API) refers to computer code or other data sorted on a computer-readable medium that may be executed by a processor to facilitate the interaction between software components, such as a client-side front-end and/or server-side back-end for receiving data from the client. An "interface" refers to a generated display, such as one or more graphical user interfaces (GUIs) with which a user may interact, either directly or indirectly (e.g., through a

keyboard, mouse, touchscreen, etc.). Further, multiple computers, e.g., servers, or other computerized devices, such as an autonomous vehicle including a vehicle computing system, directly or indirectly communicating in the network environment may constitute a “system” or a “computing system”.

[0028] It will be apparent that systems and/or methods, described herein, can be implemented in different forms of hardware, software, or a combination of hardware and software. The actual specialized control hardware or software code used to implement these systems and/or methods is not limiting of the implementations. Thus, the operation and behavior of the systems and/or methods are described herein without reference to specific software code, it being understood that software and hardware can be designed to implement the systems and/or methods based on the description herein.

[0029] Some non-limiting embodiments or aspects are described herein in connection with thresholds. As used herein, satisfying a threshold may refer to a value being greater than the threshold, more than the threshold, higher than the threshold, greater than or equal to the threshold, less than the threshold, fewer than the threshold, lower than the threshold, less than or equal to the threshold, equal to the threshold, etc.

[0030] FIG. 1 shows an exemplary architecture for performing Business Intelligence (BI) cube refresh automation, in accordance with some embodiments of the present disclosure.

[0031] In FIG. 1, a schematic diagram of a system **100** includes a data cube 101-1 to data cube 101-N (collectively referred as one or more data cubes **101**), a cube refresh automation system **103**, a user **105** and an I-server **107**. Initially, the cube refresh automation system **103** may monitor Extract Transform Load (ETL) load on the one or more data cubes **101** and may initiate a trigger for cube refresh process. The cube refresh automation system **103** may include, but not limited to, a laptop computer, a desktop computer, a Personal Computer (PC), a notebook, a smartphone, a tablet, e-book readers, a server, a network server, and cloud server. The one or more data cubes **101** may include a database or a datastore system which may store large quantities of data in the one or more data cubes **101** of the data cube system. Once the cube refresh process is triggered, the cube refresh automation system **103** may initially check status of I-server **107**. The I-server

107 may be associated with the cube refresh automation system **103** via a communication network. The communication network may be a wired or a wireless communication network. The status of the I-server **107** may be detected to be one of "active" or "inactive". When status of I-server **107** is detected to be active, the cube refresh automation system **103** may copy the old workflow logs in the one or more data cubes **101** into another back-up file. Upon copying, the cube refresh automation system **103** may delete the old workflow logs in the one or more data cubes **101** and may replace with the new workflow logs in the one or more data cubes **101**. During the insertion, the cube refresh automation system **103** may monitor the insertion status of the new workflow logs in the one or more data cubes **101**. The cube refresh automation system **103** deletes the insertion trigger for the workflow logs which have been successfully inserted. For the unsuccessful insertion, the cube refresh automation system **103** may trigger the insertion operation for the second time. In an embodiment, the cube refresh automation system **103** may trigger the insertion operation until the insertion operation is successfully inserted. In another embodiment, the cube refresh automation system **103** may trigger the insertion operation for a predefined number of times which may be initially preconfigured in the cube refresh automation system **103**. Upon successful insertions, the cube refresh automation system **103** may delete the triggers of insertion. Further, if the triggers are not successfully deleted, the cube refresh automation system **103** may trigger the deletion of the triggers for the second time. Upon the successful deletion of the triggers, the cube refresh automation system **103** may send notifications to the users **105** which includes the information inserted in the one or more data cubes **101** and the cube health status. The cube health status may include the details regarding every success or failure of data inserted into the one or more data cubes **101**.

[0032] FIG.2 shows a flow diagram that illustrates a method of facilitating BI cube refresh automation, in accordance with some embodiments of the present disclosure.

[0033] As illustrated in FIG. 2, the method **200** comprises one or more blocks implemented for facilitating the BI cube refresh automation. The method **200** may be described in the general context of computer executable instructions. Generally, computer executable instructions can include routines, programs, objects, components, data structures, procedures, modules, and functions, which perform specific functions or implement specific abstract data types.

[0034] The order in which the method **200** is described is not intended to be construed as a limitation, and any number of the described method blocks can be combined in any order to implement the method. Additionally, individual blocks may be deleted from the methods without departing from the spirit and scope of the subject matter described herein. Furthermore, the method can be implemented in any suitable hardware, software, firmware, or combination thereof.

[0035] The steps of the method shown may be carried out by the cube refresh automation system **103**. At block **201**, initially the cube refresh automation system **103** may monitor the ETL load. Upon the detection of an operation request on the one or more data cubes **101**, the cube refresh automation system **103** may initiate a trigger for cube refresh process. The trigger may be a call by the cube refresh automation system **103** to perform a particular task defined in the call function. At block **203**, once the trigger is initiated, the cube refresh automation system **103** checks the status of an I-server **107**. The I-server **107** may enable the users to access, edit, run, compile and debug all the data stored on the server and the one or more data cubes **101**. During the cube refresh process, the status of the I-server **107** must be active. The active status of the I-server **107** may indicate that the I-server **107** is up and running. If the status of I-server **107** is determined to be inactive (Not running), the cube refresh automation system **103** may wait for a predefined period of time until status of the I-server **107** is determined to be active. If the I-server **107** status is determined to be inactive even after a predetermined period, then a notification may be triggered to indicate the failure of the I-server **107**. At block **205**, once the status of the I-server **107** is determined to be active, the cube refresh automation system **103** may perform a back-up of the old workflow logs by copying the old workflow logs into a new file and then delete the old workflow logs in the one or more data cubes **101**. For instance, old workflow logs may be the workflow data of the previous month. Further, the new workflow logs may be inserted into the one or more data cubes **101**. For instance, new workflow logs may be the workflow logs of the current month. At block **207**, the cube refresh automation system **103** may insert the new workflow logs into the one or more data cubes **101**. The cube refresh automation system **103** may split the insertion data across the clustered servers for instance server 01 as indicated in **207A**, server 02 as indicated in **207B** and so on, to balance the load on the servers. Upon splitting, the insertion results from the clustered servers (sever 01, server 02 and so on) may be merged. At block **209**, the cube refresh automation system **103** may monitor merged insertion results. At block **211**, if the insertion of new workflow

logs into the one or more data cubes **101** is successful, then the cube refresh automation system **103** may delete the insertion trigger. At block **213**, also, the cube refresh automation system **103** may trigger the insertion for the second time, when the new workflow logs were not inserted successfully during the first attempt. In an embodiment, the number of iterations for triggering the insertion of new workflow logs after unsuccessful insertion may be predetermined. In yet another embodiment, the cube refresh automation system **103** may trigger the insertion operation until the new workflow log data is successfully inserted. At block **215**, the cube refresh automation system **103** may delete the insertion trigger after the successful insertion of the new workflow logs during the second attempt. At block **217**, Upon the successful insertion of the new workflow logs, the triggers for the insertion may be deleted by the cube refresh automation system **103**. At block **219**, in an embodiment, if the deletion of the triggers becomes unsuccessful, then the cube refresh automation system **103** may trigger the deletion of trigger for the second time. In another embodiment, the cube refresh automation system **103** may trigger the deletion of insertion trigger until the insertion trigger is successfully deleted. In an embodiment, number of iterations for triggering the deletion of insertion trigger may be predetermined. At block **221**, finally, upon the successful deletion of the insertion trigger, the cube refresh automation system **103** may provide the cube information and the cube health status to the user **105**.

[0036] FIG.3 shows an exemplary scenario of a cube refresh automation process, in accordance with some embodiments of the present disclosure.

[0037] As illustrated in Fig. 3 (continued), upon the detection of ETL load, the status of the I-server **107** may be checked. If the status of the I-server **107** is determined to be not running actively, then the cube refresh automation system **103** may wait for a predetermined time for instance 30 min until the status of I-server **107** is active. The predetermined time may be limited to 30 min, it may vary accordingly. Once the predetermined period is exceeded, the trigger for checking the status of I-server **107** is re-initiated. Once the trigger is reinitiated, the failure counter is incremented, and the initiate failure count value is updated. For instance, consider the I-server **107** was inactive for 3 trials, the initiate failure count may be updated to 3 and each time the I-server **107** is inactive the parameter “failure count” is incremented accordingly. Once the “failure count” value exceeds the predetermined value, then a notification may be sent to the users indicating that the I-server **107** is not running actively. For instance, the predetermined value for

the number of times the accepting reinitiate request for checking the status of the I-server **107** may be 5 as shown in Fig. 3. Once, the status of the I-server **107** is determined to be active, the cube refresh automation system **103** may move the previous month workflow logs to a new file and rename the workflow log file appropriately. The cube refresh automation system **103** may split the insertion of new workflow logs into clustered servers. Particularly the cube refresh automation system **103** may trigger the insert operation for batch 1 and may trigger the insert operation for batch 2. Further, the insertion results from the server batch 1 and the server batch 2 are merged. Upon merging, the first insertion failure log is monitored. If the insertion of the new workflow log is performed successfully, the trigger for the insertion may be deleted by the cube refresh automation system **103**. If the insertion operation is unsuccessful, then the cube refresh automation system **103** may provide a second trigger for the insertion operation. The second trigger may be provided after a predetermined time, for instance 30 min. Further, the cube refresh automation system **103** may delete the insertion trigger after the successful insertion of the new workflow log in the second attempt. After the second attempt, if still the insertion operation is unsuccessful, the cube refresh automation system **103** may wait for a predetermined period to perform a successful insertion. Once the threshold period is reached, the insertion trigger is deleted upon successive trial of insertion. In case the deletion of the trigger is unsuccessful, then the cube refresh automation system **103** may provide a second trigger for the deletion of the insertion trigger. Finally, upon the successful deletion of the insertion trigger, the cube refresh automation system **103** may provide the cube information which includes the new workflow logs and the health status of the cubes to the user **105**.

[0038] For example, the cube refresh automation system may insert the new workflow logs. During the refresh process, the exemplary new workflow logs which may be inserted are as indicated below:

Refresh Process : Insert new month

10/6/22 8:16:07 PM UTC PREPARING process 'Insert Trigger_Batch2' on 'October 6, 2022 5:07:12 PM UTC'
10/6/22 8:16:07 PM UTC RUNNING process:Insert Trigger_Batch2
10/6/22 8:16:07 PM UTC process data:{Suppress hidden object(s) in the results=FALSE, Split output into three specified files=TRUE, Log output to specified file=FALSE, Break log file=FALSE, Script File (.scp)=FALSE, Connect to a Project Source=TRUE, Execute script statements=TRUE, Output file=, Success File=E:\VVM\VVM CUBE REFRESH\VVM_PROD_LOGS\INSERT_TRIGGER1ST_SUCCESS.TXT, Results

File=E:\VVM\VVM CUBE REFRESH\VVM_PROD_LOGS\INSERT_TRIGGER1ST_RESULT.TXT, Execute script statements=PUBLISH INCREMENTAL REFRESH REPORT 'IR_PRODUCT_COMPARISON_CARDCOUNT_ISSR_INSERT_CURRENT' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";
10/6/22 8:16:07 PM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_PRODUCT_COMPARISON_DRAFT_ISSR_INSERT_CURRENT' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";
10/6/22 8:16:07 PM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_RISKSCORE_AUTH_ISSR_INSERT_CURRENT' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";
10/6/22 8:16:07 PM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_RISKSCORE_FRAUD_ISSR_INSERT_LAST6MONTHS' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";
10/6/22 8:16:07 PM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_VISA_DIRECT_ISSR_AFT_INSERT_CURRENT' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";
10/6/22 8:16:07 PM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_VISA_DIRECT_ISSR_OCT_INSERT_CURRENT' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";
10/6/22 8:16:07 PM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_VISADIRECT_OCT_ACQR_AUTH_INSERT_CURRENT' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";
10/6/22 8:16:07 PM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_VISADIRECT_AFT_ISSR_AUTH_INSERT_CURRENT' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";
10/6/22 8:16:07 PM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_VISADIRECT_OCT_ISSR_AUTH_INSERT_CURRENT' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_INSERT" FOR PROJECT "VVM";, Connection-less Session=FALSE, Export results to an XML file=FALSE, Split output into three default files (results, failure, and success)=FALSE, Log output to default file=FALSE, Display Output on the Console=FALSE, Include file log header=FALSE, Connection-less Session=FALSE, XML file=, Stop script execution on error=FALSE, Log output to default file=FALSE, Include instructions in the log file(s)=FALSE, Password=*******, Script File (.scp)=, Failure File=E:\VVM\VVM CUBE REFRESH\VVM_PROD_LOGS\INSERT_TRIGGER1ST_FAILURE.TXT, Include error codes in the log file(s)=FALSE}

[0039] FIG. 4A and FIG. 4B show exemplary graphs illustrating product comparisons based on card count and drafts respectively, as indicated in the exemplary insert logs above. FIG. 4C and FIG. 4D show exemplary graphs illustrating Risk score auth and risk score fraud respectively, as indicated in the exemplary insert logs above. FIG. 4E and FIG. 4F show exemplary graphs illustrating Direct AFT and Direct OCT as indicated in the exemplary insert logs above.

[0040] For example, the cube refresh automation system may delete the old workflow logs. During the refresh process, the exemplary old workflow logs which may be deleted are as indicated below:

Refresh Process : Delete old month

10/7/22 3:53:30 AM UTC PREPARING process 'Delete Trigger' on 'October 7, 2022 1:31:29 AM UTC'
10/7/22 3:53:30 AM UTC RUNNING process:Delete Trigger
10/7/22 3:53:30 AM UTC process data:{Suppress hidden object(s) in the results=FALSE, Split output into three specified files=TRUE, Log output to specified file=FALSE, Break log file=FALSE, Script File (.scp)=FALSE, Connect to a Project Source=TRUE, Execute script statements=TRUE, Output file=, Success File=E:\VVM\VVM CUBE REFRESH\VVM_PROD_LOGS\DELETE_TRIGGER1ST_SUCCESS.TXT, Results File=E:\VVM\VVM CUBE REFRESH\VVM_PROD_LOGS\DELETE_TRIGGER1ST_RESULT.TXT, Execute script statements=PUBLISH INCREMENTAL REFRESH REPORT 'IR_AUTHORIZATION_APPROVALS_ACQR_DELETE_LASTMONTH' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_DELETE" FOR PROJECT "VVM";
10/7/22 3:53:30 AM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_AUTHORIZATION_APPROVALS_ISSR_DELETE_LASTMONTH' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_DELETE" FOR PROJECT "VVM";
10/7/22 3:53:30 AM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_CURRENCY_DELETE' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_DELETE" FOR PROJECT "VVM";
10/7/22 3:53:30 AM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_PRODUCT_COMPARISON_CARDCOUNT_ISSR_DELETE_LASTMONTH' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_DELETE" FOR PROJECT "VVM";
10/7/22 3:53:30 AM UTC PUBLISH INCREMENTAL REFRESH REPORT 'IR_PRODUCT_COMPARISON_DRAFT_ISSR_DELETE_LASTMONTH' IN FOLDER "PUBLIC OBJECTS\REPORTS\INCREMENTAL REFRESH\IR_DELETE" FOR PROJECT "VVM";
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs'...
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\CubeMonthInfo & Logs.zip' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\CubeMonthInfo & Logs.zip'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\CubeMonthInfo & Logs.zip'.

10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\CubeMonthSubs_Failure.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\CubeMonthSubs_Failure.txt'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\CubeMonthSubs_Failure.txt'.
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\CubeMonthSubs_Result.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\CubeMonthSubs_Result.txt'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\CubeMonthSubs_Result.txt'.
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\CubeMonthSubs_Success.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\CubeMonthSubs_Success.txt'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\CubeMonthSubs_Success.txt'.
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Delete_Trigger1st_Failure.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\Delete_Trigger1st_Failure.txt'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Delete_Trigger1st_Failure.txt'.
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Delete_Trigger1st_Result.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\Delete_Trigger1st_Result.txt'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Delete_Trigger1st_Result.txt'.
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Delete_Trigger1st_Success.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\Delete_Trigger1st_Success.txt'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Delete_Trigger1st_Success.txt'.
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Insert_Trigger1st_Failure.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\Insert_Trigger1st_Failure.txt'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Insert_Trigger1st_Failure.txt'.
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Insert_Trigger1st_Result.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\Insert_Trigger1st_Result.txt'...
10/6/22 5:07:09 PM UTC FEEDBACK: Deleting 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Insert_Trigger1st_Result.txt'.
10/6/22 5:07:09 PM UTC FEEDBACK: Copying 'E:\VVM\VVM Cube Refresh\VVM_PROD_Logs\Insert_Trigger1st_Success.txt' into 'E:\VVM\VVM Cube Refresh\WF_LOG_FILES_BKP\VVM_PROD_Logs\Insert_Trigger1st_Success.txt'...

[0041] FIG.4G and FIG.4H show exemplary graphs illustrating authorization approvals as indicated in the exemplary delete logs above. FIG.4I shows an exemplary graph illustrating currency validation as indicated in the exemplary delete logs above.

[0042] Finally, the language used in the specification has been principally selected for readability and instructional purposes, and it may not have been selected to delineate or circumscribe the inventive subject matter. Accordingly, the disclosure of the embodiments of the disclosure is intended to be illustrative, but not limiting, of the scope of the disclosure.

[0043] With respect to the use of substantially any plural and/or singular terms herein, those having skill in the art can translate from the plural to the singular and/or from the singular to the plural as is appropriate to the context and/or application. The various singular/plural permutations may be expressly set forth herein for sake of clarity.

[0044] Although the invention has been described in detail for the purpose of illustration based on what is currently considered to be the most practical and preferred embodiments, it is to be understood that such detail is solely for that purpose and that the invention is not limited to the disclosed embodiments, but, on the contrary, is intended to cover modifications and equivalent arrangements that are within the spirit and scope of the disclosure. For example, it is to be understood that the present invention contemplates that, to the extent possible, one or more features of any embodiment can be combined with one or more features of any other embodiment.

METHOD AND SYSTEM FOR BI CUBE REFRESH AUTOMATION

ABSTRACT

The present disclosure relates to a system for BI cube refresh automation. In the present disclosure, initially the status of the I-server **107** is determined. Once the I-server is active, the cube refresh automation system **103** may back-up the old workflow logs and deletes the old workflow logs from one or more data cubes **101**. Further, the cube refresh automation system may split the insertion of new workflow logs into clustered server and then merges the insertion results from the clustered servers. The merger insertion results are further monitored. Upon successful insertion, the insertion trigger is deleted. When the insertion is unsuccessful, the cube refresh automation system provides a successive trigger for insertion. Further, upon the successful insertion the trigger for the insertion is deleted. Finally, the cube refresh automation system **103** provides the information inserted in one more data cubes and cube health status to the user through notification.

FIG. 2

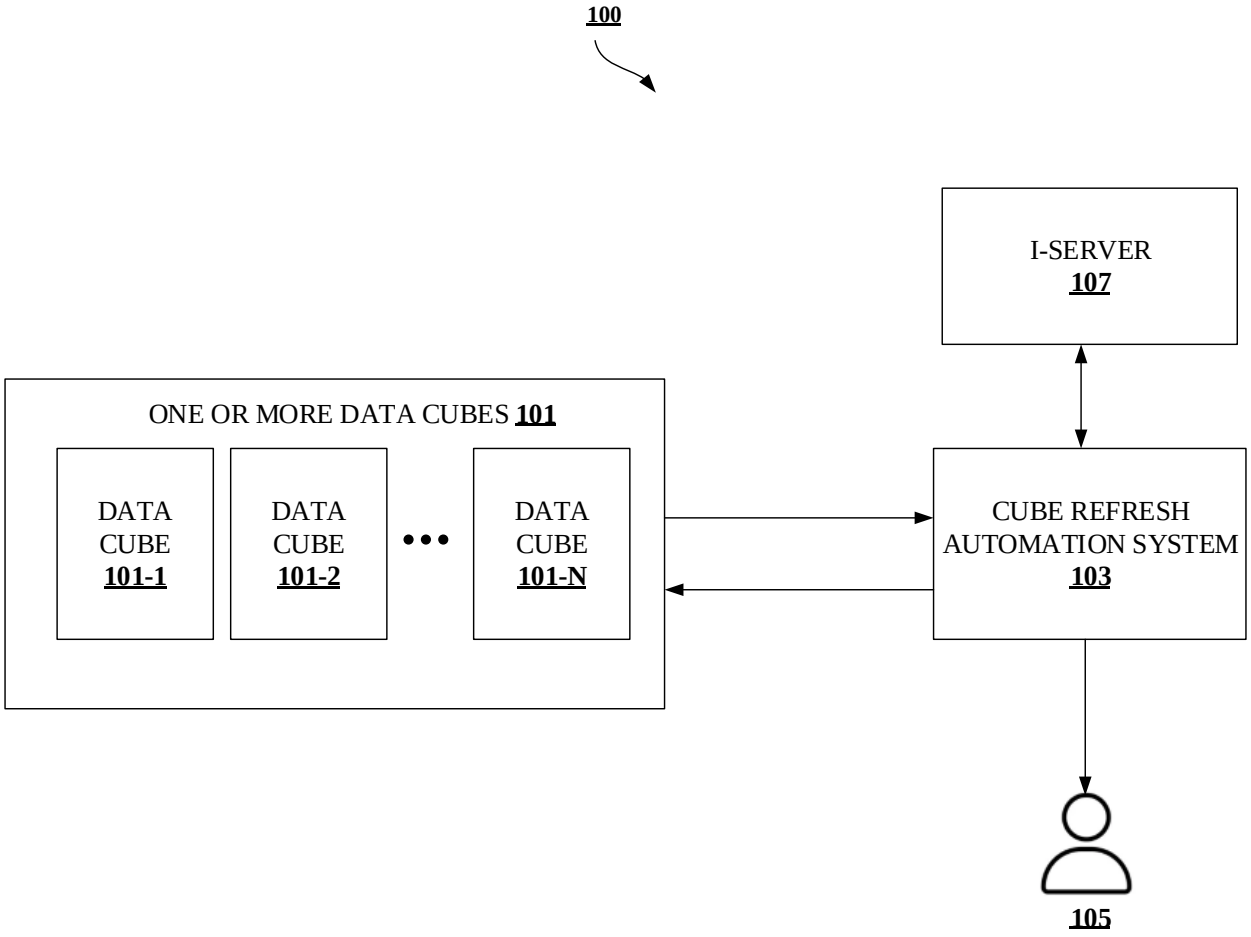
100
I-SERVER
107
ONE OR MORE DATA CUBES 101
DATA CUBE 101-1
DATA CUBE 101-2
DATA CUBE 101-N
CUBE REFRESH AUTOMATION SYSTEM 103
105

FIG. 1

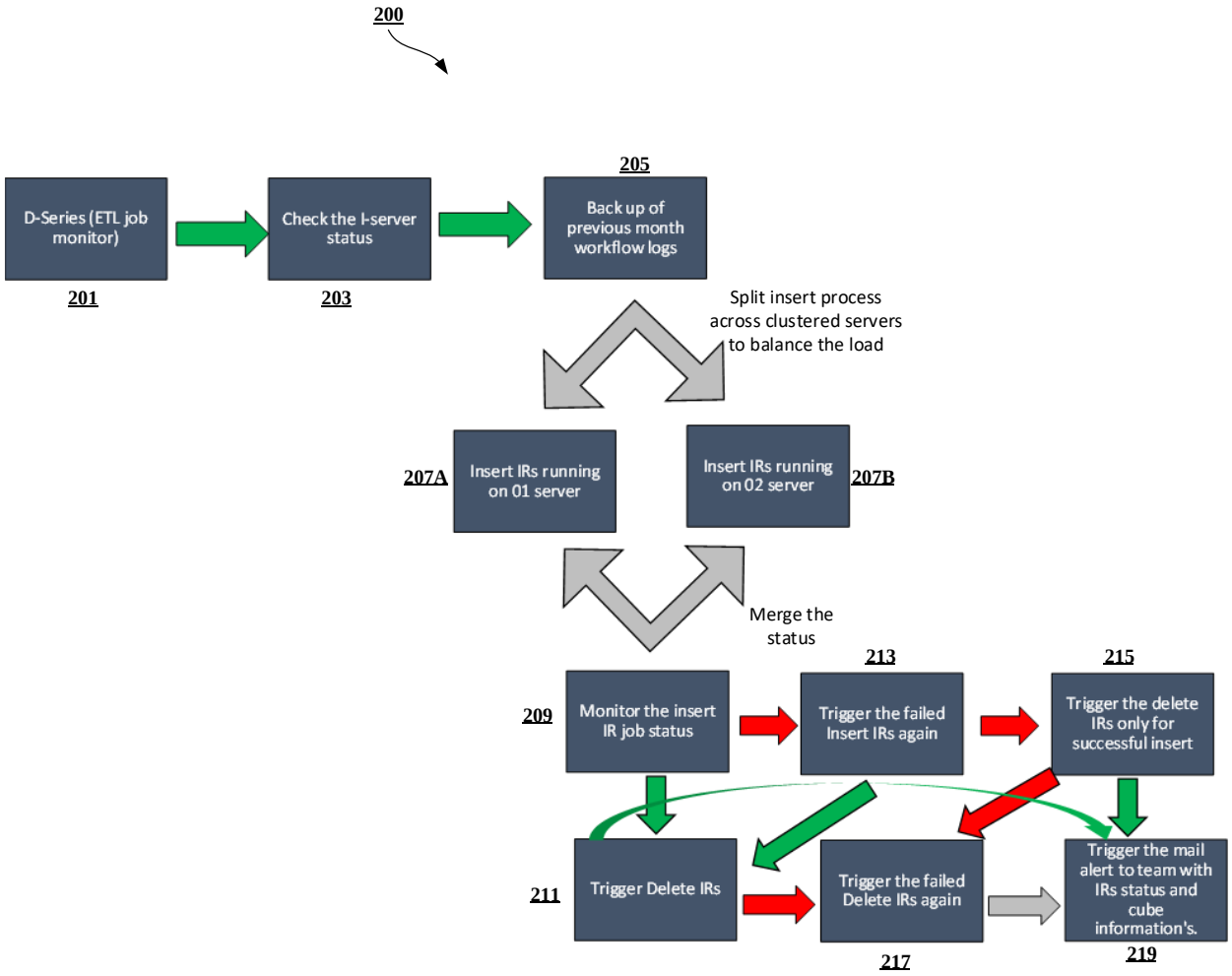
200
D-Series (ETL job monitor)
201
Check the I-server status
203
205
Back up of previous month workflow logs
Split insert process across clustered servers to balance the load
207A
Insert IRs running on 01 server
Insert IRs running on 02 server
207B
Merge the status
209
Monitor the insert IR job status
213
Trigger the failed Insert IRs again
215
Trigger the delete IRs only for successful insert
211
Trigger Delete IRs
Trigger the failed Delete IRs again
217
Trigger the mail alert to team with IRs status and cube information's.
219

FIG. 2

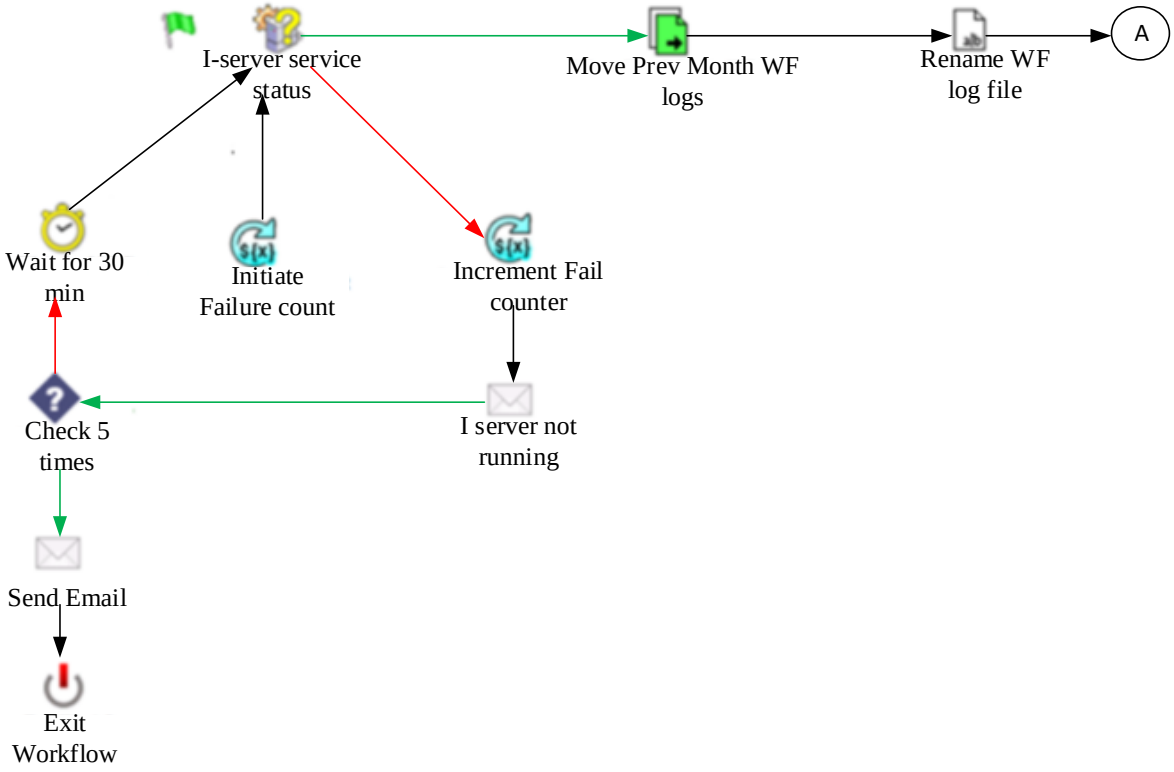

FIG. 3 (continued)

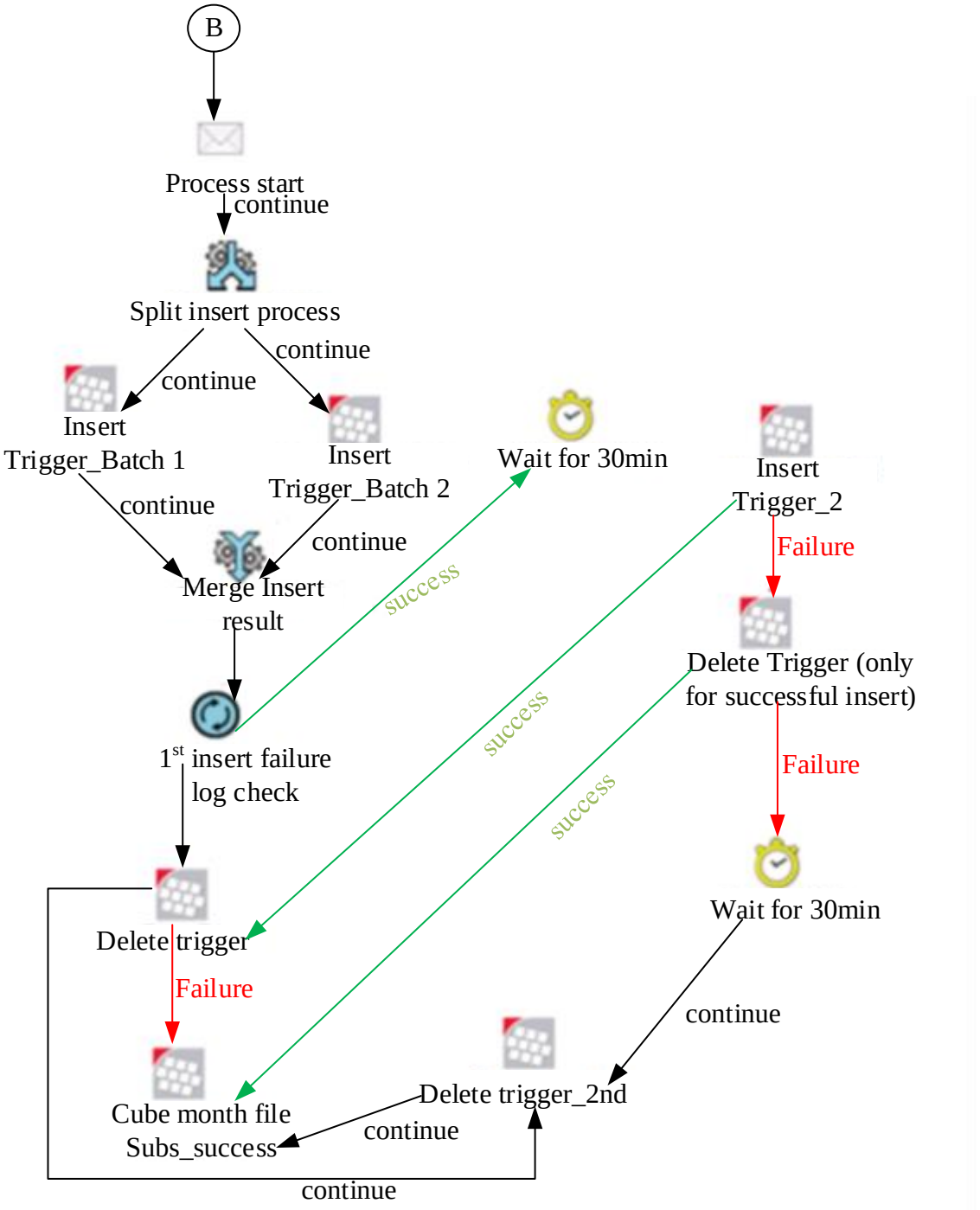
B
Process start
continue
Split insert process
continue
continue
Insert
Trigger_Batch 1
Insert
Trigger_Batch 2
Wait for 30min
Insert
Trigger_2
continue
continue
Merge Insert
result
success
Failure
Delete Trigger (only
for successful insert)
1st insert failure
log check
success
success
Failure
Wait for 30min
Delete trigger
Failure
continue
Delete trigger_2nd
Cube month file
Subs_success
continue
continue

FIG. 3

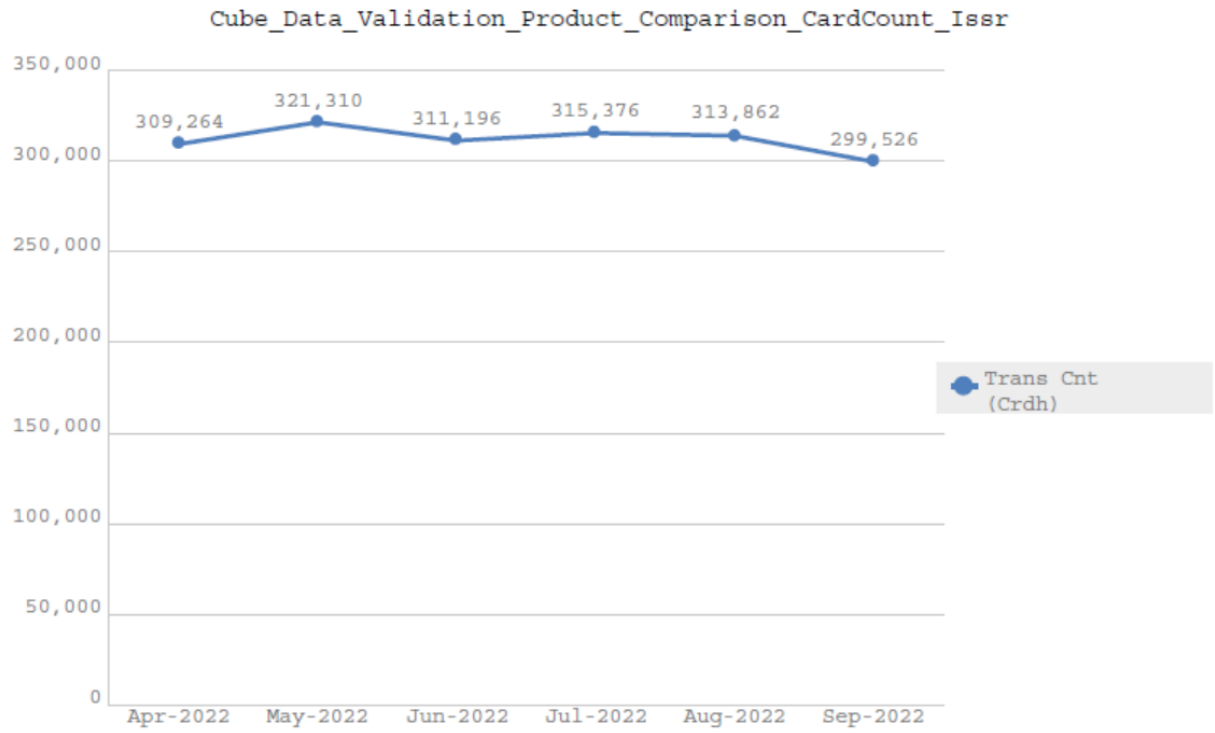
Cube_Data_Validation_Product_Comparison_CardCount_Issr
350,000
300,000
250,000
200,000
150,000
100,000
50,000
0
309,264
321,310
311,196
315,376
313,862
299,526
Apr-2022
May-2022
Jun-2022
Jul-2022
Aug-2022
Sep-2022
Trans Cnt (Crdh)

FIG. 4A

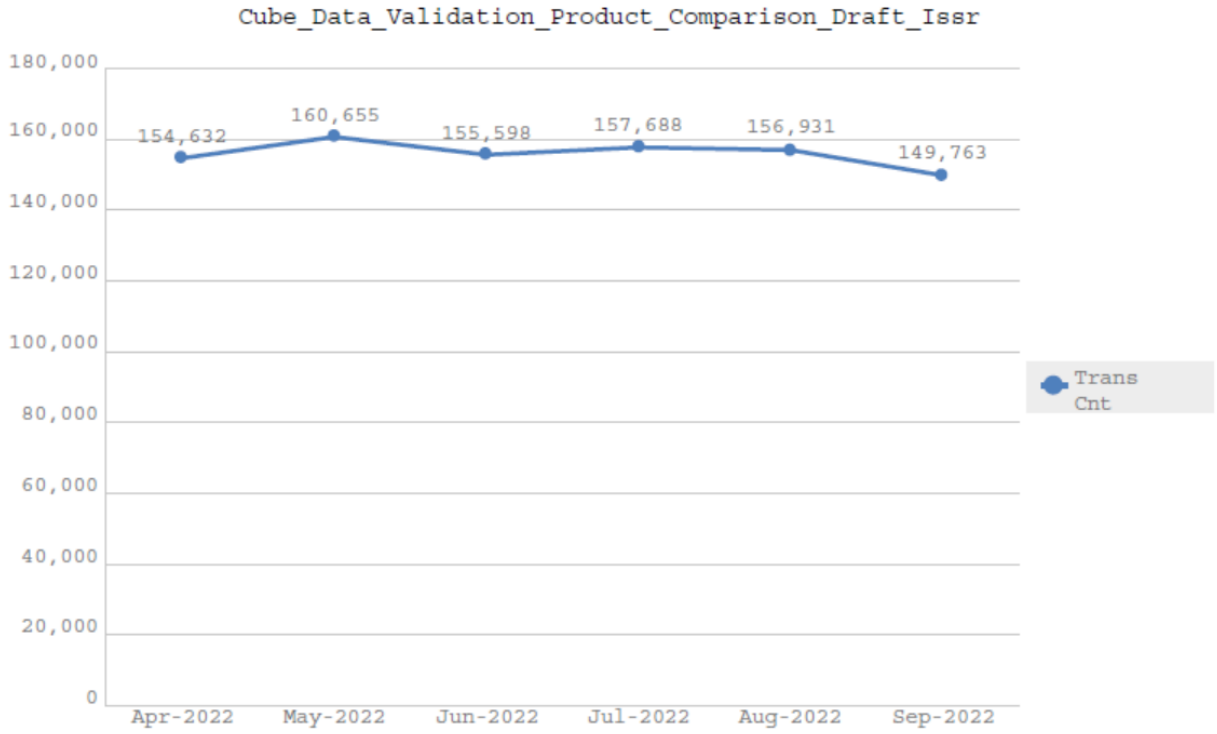
Cube_Data_Validation_Product_Comparison_Draft_Issr
180,000
160,000
140,000
120,000
100,000
80,000
60,000
40,000
20,000
0
154,632
160,655
155,598
157,688
156,931
149,763
Apr-2022
May-2022
Jun-2022
Jul-2022
Aug-2022
Sep-2022
Trans Cnt

FIG. 4B

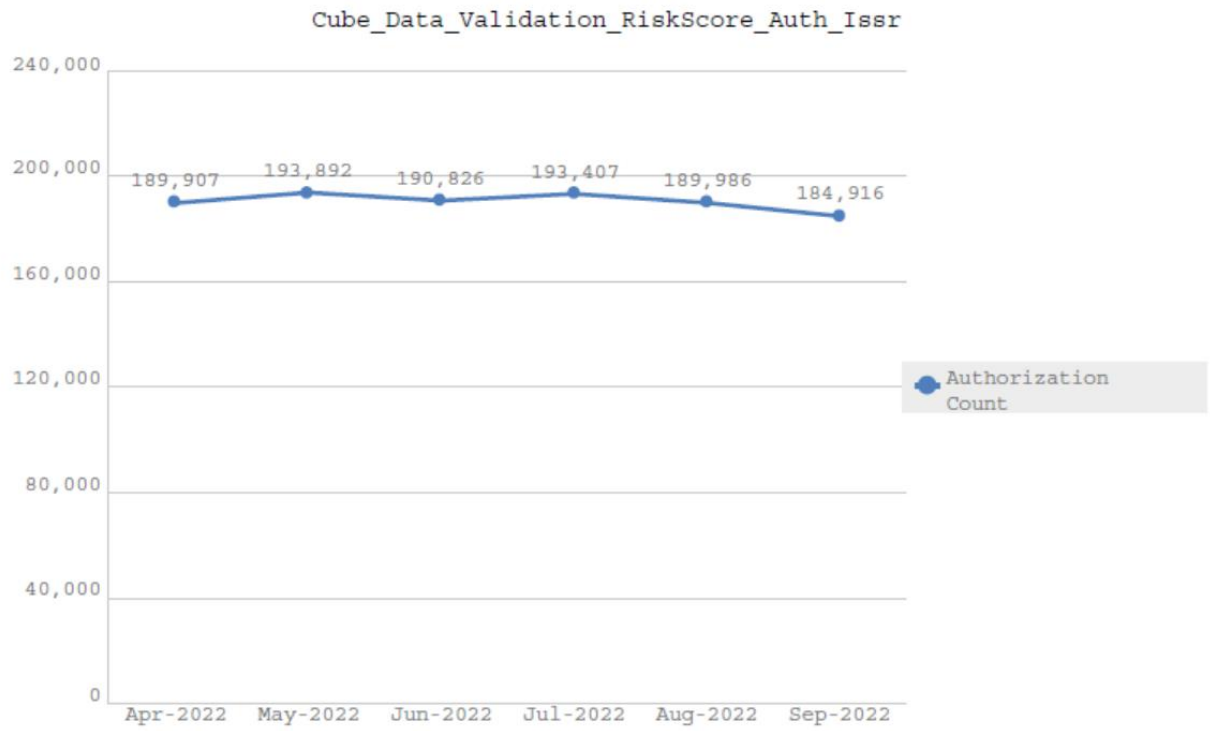
Cube_Data_Validation_RiskScore_Auth_Issr
240,000
200,000
160,000
120,000
80,000
40,000
0
189,907
193,892
190,826
193,407
189,986
184,916
Apr-2022
May-2022
Jun-2022
Jul-2022
Aug-2022
Sep-2022
Authorization Count

FIG. 4C

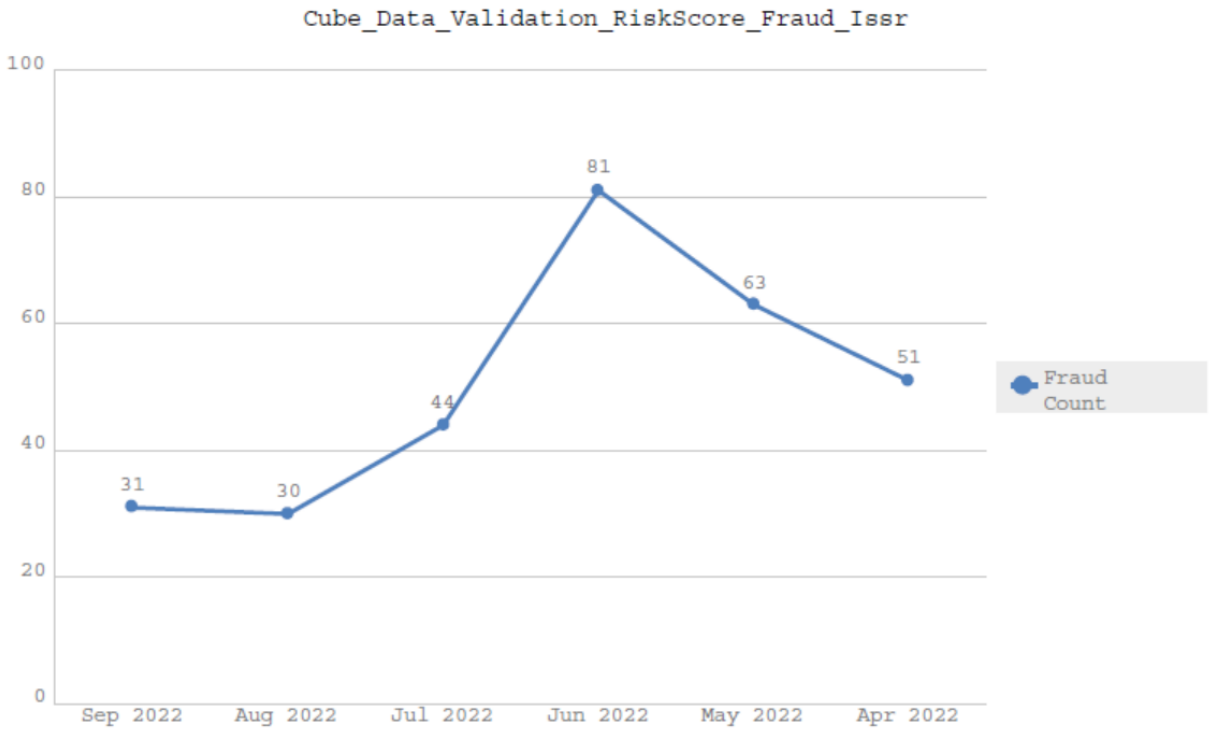
Cube_Data_Validation_RiskScore_Fraud_Issr
100
80
60
40
20
0
31
30
44
81
63
51
Sep 2022
Aug 2022
Jul 2022
Jun 2022
May 2022
Apr 2022
Fraud Count

FIG. 4D

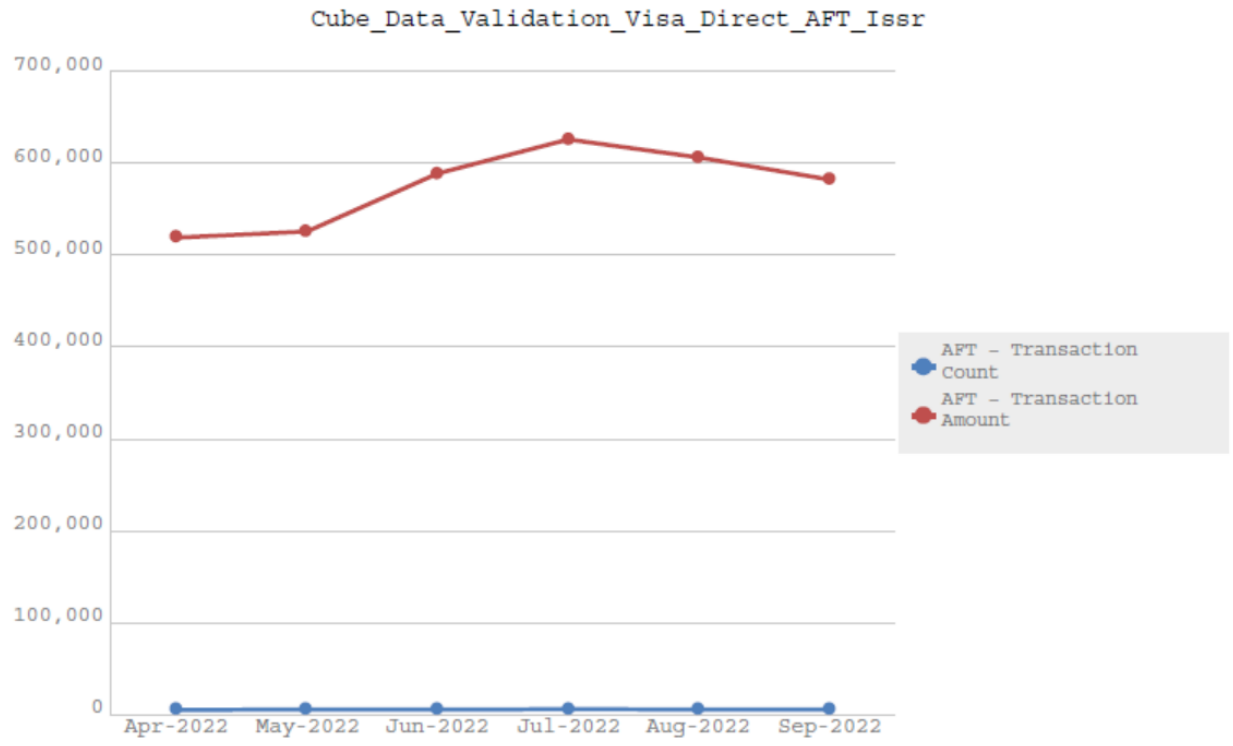
Cube_Data_Validation_Visa_Direct_AFT_Issr
700,000
600,000
500,000
400,000
300,000
200,000
100,000
0
Apr-2022
May-2022
Jun-2022
Jul-2022
Aug-2022
Sep-2022
AFT - Transaction Count
AFT - Transaction Amount

FIG. 4E

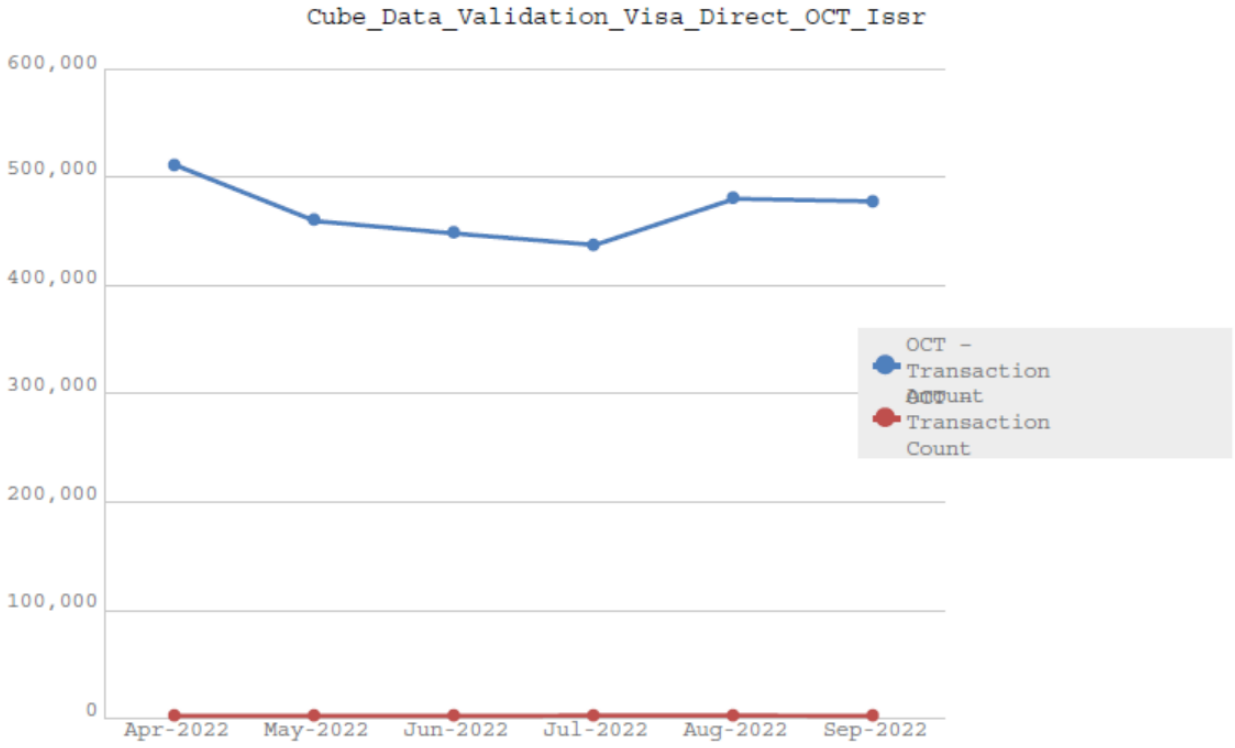
Cube_Data_Validation_Visa_Direct_OCT_Issr
600,000
500,000
400,000
300,000
200,000
100,000
0
Apr-2022
May-2022
Jun-2022
Jul-2022
Aug-2022
Sep-2022
OCT - Transaction Count
OCT - Transaction Count

FIG. 4F

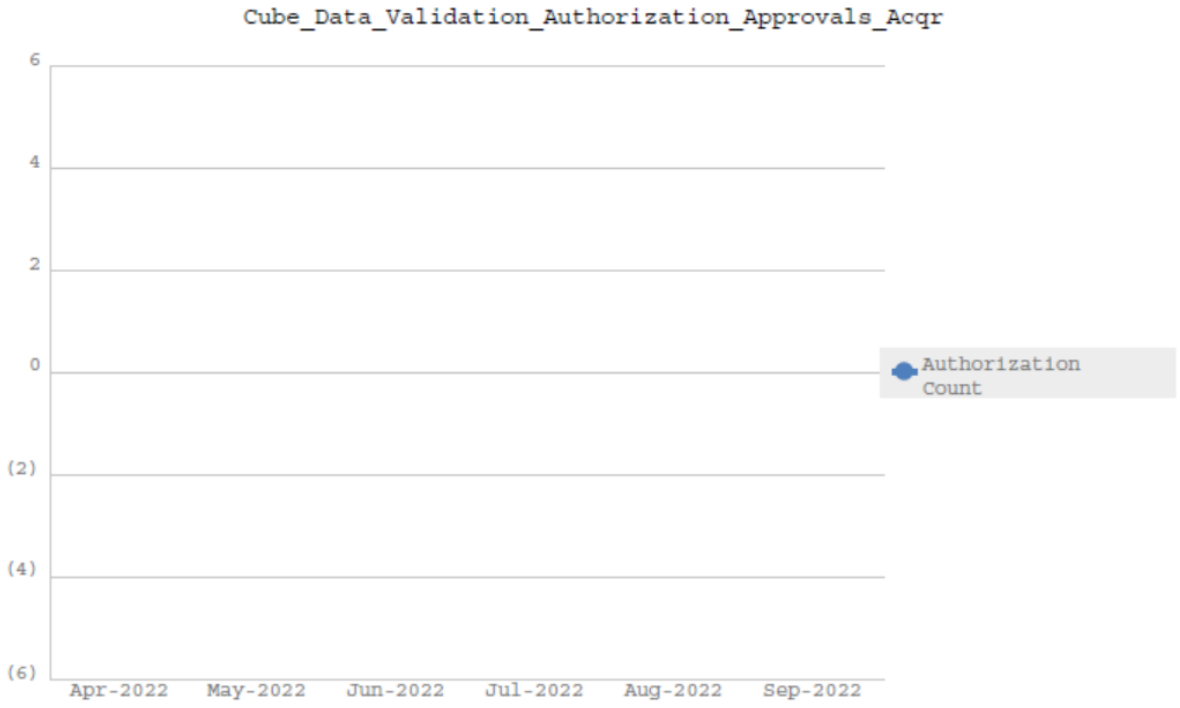
Cube_Data_Validation_Authorization_Approvals_Acqr
6
4
2
0
(2)
(4)
(6)
Apr-2022
May-2022
Jun-2022
Jul-2022
Aug-2022
Sep-2022
Authorization Count

FIG. 4G

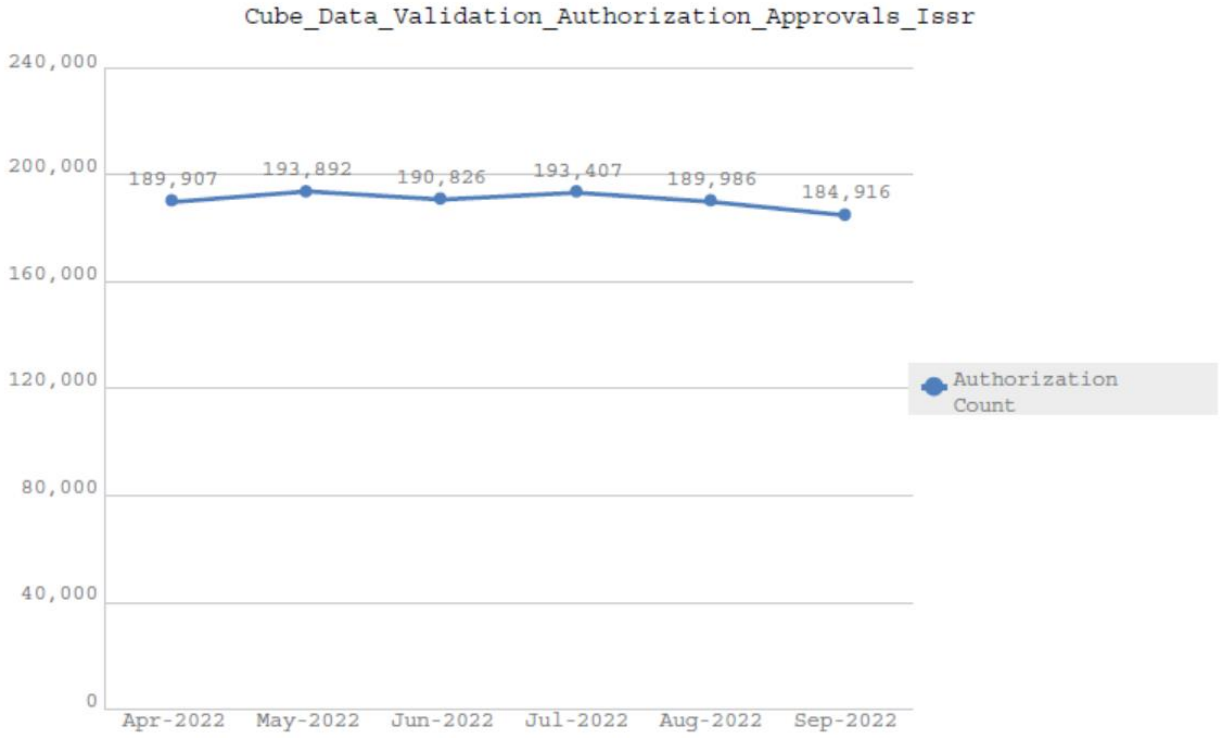
Cube_Data_Validation_Authorization_Approvals_Issr
240,000
200,000
160,000
120,000
80,000
40,000
0
189,907
193,892
190,826
193,407
189,986
184,916
Apr-2022
May-2022
Jun-2022
Jul-2022
Aug-2022
Sep-2022
Authorization Count

FIG. 4H

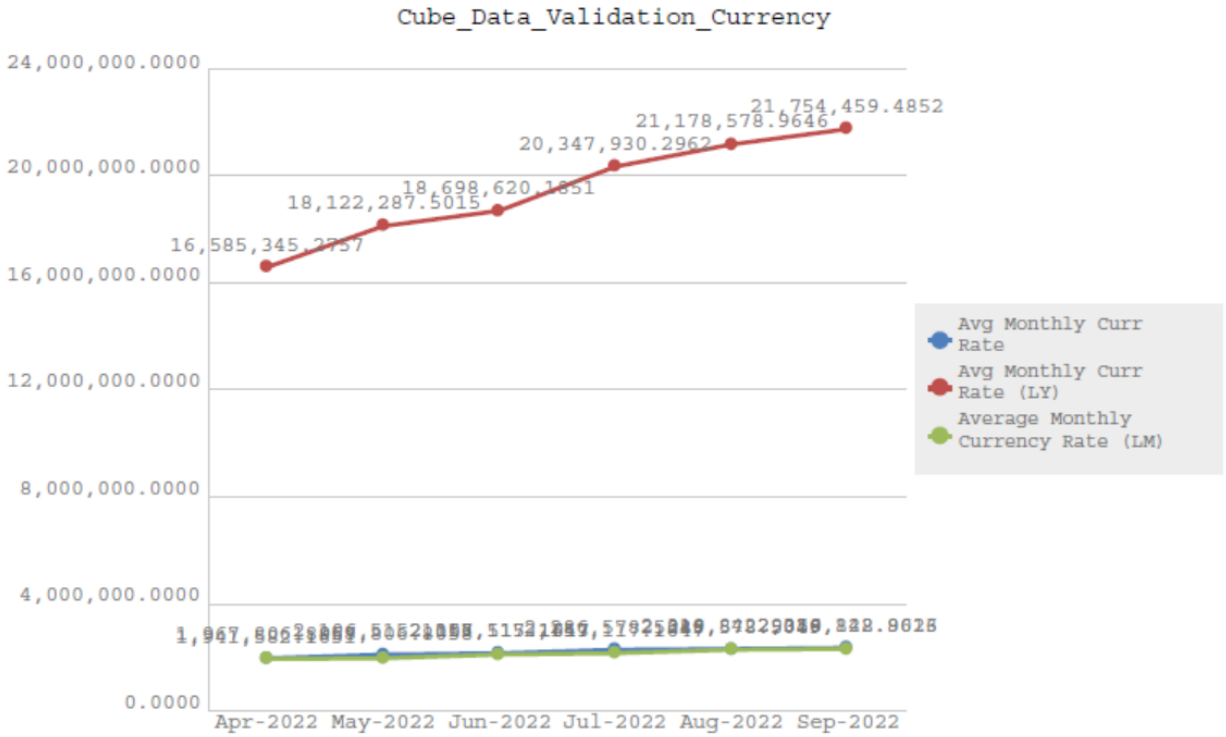
Cube_Data_Validation_Currency
24,000,000.0000
20,000,000.0000
16,000,000.0000
12,000,000.0000
8,000,000.0000
4,000,000.0000
0.0000
16,585,345.2757
18,122,287.5015
18,698,620.1851
20,347,930.2962
21,178,578.9646
21,754,459.4852
Apr-2022
May-2022
Jun-2022
Jul-2022
Aug-2022
Sep-2022
Avg Monthly Curr Rate
Avg Monthly Curr Rate (LY)
Average Monthly Currency Rate (LM)

FIG. 4I