



(51) International Patent Classification:

G06F 21/62 (2013.01)

(21) International Application Number:

PCT/CN2019/090836

(22) International Filing Date:

12 June 2019 (12.06.2019)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant: ALIBABA GROUP HOLDING LIMITED

[—/CN]; Fourth Floor, One Capital Place, P.O. Box 847, George Town, Grand Cayman (KY).

(72) Inventors: DING, Bolin; 17180 NE 123RD ST, Redmond,

WA 98052 (US). ZHOU, Jingren; Xixidieyuan 19-801, Yu Hang District, Hangzhou, Zhejiang (CN). HONG, Cheng;

Wangjing Block A, Alibaba Center No. 9th Wangjing East Garden 4th Area, Chaoyang District, Beijing (CN). HUANG, Zhicong; Room 2303, Building 5, Xuancailiaxuan, Chaoyang District, Beijing (CN). XU, Min; 1400 E 55th Pl. Apt 705S, Chicago, IL 60637 (US). WANG, Tianhao; 305 N. University Street, West Lafayette, Indiana, 47906 (US).

(74) Agent: BEIJING TSINGYUANHUI INTELLECTUAL PROPERTY LAW FIRM; Room 101, 1st Floor, Building 1, No. C-18, Zhichun Road, Haidian District, Beijing 100190 (CN).

(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, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP,

(54) Title: DATA SHARING AND DATA ANALYTICS IMPLEMENTING LOCAL DIFFERENTIAL PRIVACY

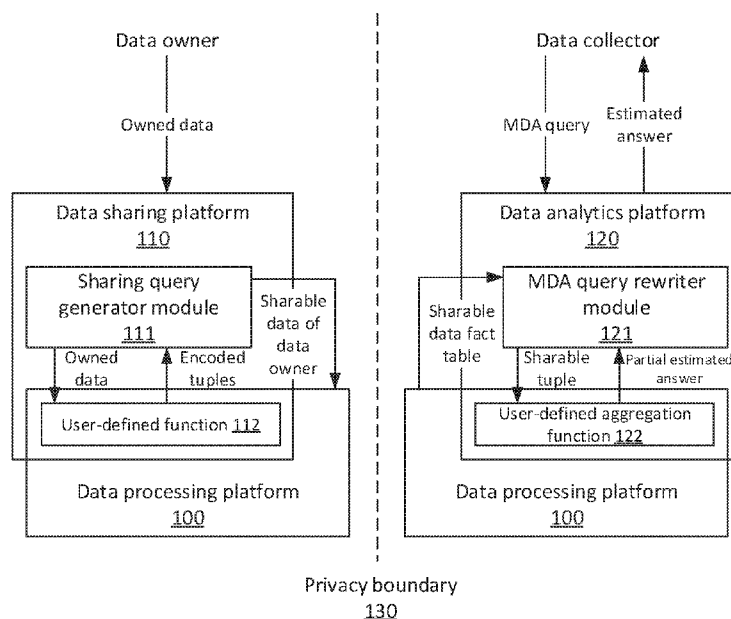


FIG. 1

(57) Abstract: Methods and systems are provided for implementing a data sharing platform providing services to a data processing platform and a data analytics platform providing services to a data processing platform, including a data sharing platform receiving owned data submitted by a data owner to a data processing platform; a sharing query generator module of the data sharing platform writing a generated query; a data analytics platform receiving a request for the owned data from a data collector; and the data sharing platform providing sharable data to a data analytics platform. These methods and systems allow services to be provided guaranteeing LDP in a self-enforcing manner between data owners and non-trusted data collectors, and the services to be scaled to computing resources and data throughput of the data processing platform itself, taking advantage of distributed computing, parallel computing, improved availability of physical or virtual computing resources, and such benefits that the data processing platform may provide.



KR, KW, 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).

Published:

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

DATA SHARING AND DATA ANALYTICS IMPLEMENTING LOCAL DIFFERENTIAL PRIVACY

BACKGROUND

[0001] In data analytics, individual data owners may submit data to a data processing platform, the submitted data including at least some sensitive data, such as data which identifies an individual data owner. For example, a website or a mobile application by an entity operating a service such as a social media network, an online retailer, a video streaming website, a photo-sharing website, a dating website, and the like may allow data owners using the services to submit data including sensitive data to the data processing platform. The data may be stored in a database of the data processing platform. As a condition for data owners' submission of data over the sharing platform, the data processing platform may need to guarantee to maintain some degree of privacy or security over at least the sensitive data collected, where the guarantee may not be subject to enforcement by statutes, regulations, contractual terms, or other legal means.

[0002] A data collector may submit queries that cause the provided data to be aggregated and returned to the data collectors. A data collector may be, for example, the entity operating one of the above services. A data collector may be non-trusted; for example, the data collector may not be legally subject to a guarantee regarding privacy or security, and terms of the guarantee may not be enforceable over the data collector. It is desirable for users to prevent the data sharing platform and data analytics platform from returning sensitive data from the database to a non-trusted data collector in a self-enforcing manner, or, even if returned data is anonymized and aggregated, from returning data from which sensitive data may be derived to the non-trusted data collector.

[0003] Techniques such as anonymization and aggregation of data may be insufficient to protect privacy of sensitive data in the face of known techniques for identifying individuals using anonymized and aggregated data, and thus new techniques are required to ensure data privacy against non-trusted data collectors.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] The detailed description is set forth with reference to the accompanying figures. In the figures, the left-most digit(s) of a reference number identifies the figure in which the reference number first appears. The use of the same reference numbers in different figures indicates similar or identical items or features.

[0005] FIG. 1 illustrates a data processing platform according to example embodiments of the present disclosure.

[0006] FIGS. 2A and 2B illustrate graphical user interfaces according to example embodiments of the present disclosure that receive data submitted by data owners.

[0007] FIG. 3 illustrates a graphical user interface according to example embodiments of the present disclosure that processes a multi-dimensional analytical (MDA) query input by a data collector.

[0008] FIG. 4 illustrates a flowchart of a data sharing method according to example embodiments of the present disclosure.

[0009] FIG. 5 illustrates a flowchart of a data analytics method according to example embodiments of the present disclosure.

[0010] FIG. 6 illustrates a flowchart of processing data sharing and data analytics at a data sharing platform and a data analytics platform hosted as services for a data processing platform according to example embodiments of the present disclosure.

[0011] FIG. 7 illustrates a flowchart of processing data sharing and data analytics at a data processing platform according to example embodiments of the present disclosure.

[0012] FIG. 8 illustrates an example system for implementing the processes and methods described above for implementing a data sharing platform and data analytics platform.

DETAILED DESCRIPTION

[0013] Systems and methods discussed herein are directed to implementing local differential privacy (“LDP”) for data sharing and for data analytics, and more specifically a sharing query generator for a data sharing platform and an MDA query rewriter for a data analytics platform.

[0014] Implementing LDP may be understood by persons skilled in the art as implementing one or more systems or methods that cause data returned in response to queries of a database by a data collector to satisfy the parameter ϵ with regard to sensitive data in the database. Persons skilled in the art will understand the parameter ϵ as an arbitrarily set parameter that notates a probability that differences in data returned in response to a query by a data collector correlate to differences in sensitive data between particular individuals in the database, and that ϵ may be chosen to notate an extent to which an implemented system or method should decrease the probability that differences in the data returned correlate to differences in sensitive data between particular individuals in the database. Persons skilled in the art will further understand that the concept of LDP does not necessarily suggest, and is not limiting as to, how a system or method should be implemented to satisfy a parameter ϵ , except that systems and methods according to LDP may be implemented so as to at least prevent owned data from entering the possession of a data collector unless it has been altered in some way. The data collector may, for example, be a non-trusted data collector, though for the purpose of implementing LDP all data collectors may be assumed to be non-trusted.

[0015] Thus, according to example embodiments of the present disclosure, LDP may be implemented at a data sharing platform provided to a data processing platform and at a data analytics platform provided to a data processing platform. LDP may be implemented as a sharing query generator module and a user-defined function (UDF) of a data sharing platform, the sharing query generator module and the UDF being operable to encode owned data stored in a database of the data processing platform to generate LDP-encoded data. Post-processing may further be implemented as an MDA query rewriter module and a user-defined aggregation function (UDAF) of a data analytics platform, the MDA query rewriter module and the UDAF being operative to receive a first query input by a data collector at the data analytics platform and compose a second query different from the first query.

[0016] FIG. 1 illustrates a data processing platform 100 according to example embodiments of the present disclosure. A data processing platform 100 may be one or more applications running on a computing device and/or one or more services hosted

on a network provided for a database. A data processing platform 100 according to the present disclosure may refer to one or more such platforms, each capable of performing functionalities referenced to in the context of the disclosure. The database may be stored on a computer or web server, distributed across multiple physically networked computers or web servers, distributed across computers or networks over a physical or virtual cluster, or otherwise stored by other computing architectures providing storage as known by persons skilled in the art. Applications or services provided by the data processing platform 100 may include an application or web server hosted over an Internet port providing a user interface enabling data owners to submit data to the data processing platform, a file server providing data distribution, backups, and redundancy, and such services known to persons skilled in the art to provide common functionality to database servers. Such services are not illustrated herein.

[0017] Applications or services provided by the data processing platform 100 may or may not expose data to data collectors, whether data stored by the data processing platform 100 or other data. For example, data may be exposed to data collectors by an application or web server providing a web-hosted graphical user interface, command line interface, SQL interface, application programming interface (API), or other web interfaces suitable for querying data upon being operated by a data collector connecting to an Internet port of the web server by operating a computing device.

[0018] According to example embodiments of the present disclosure, owned data includes at least some sensitive data of data owners. The owned data may be structured by a database schema, which may be a relational database schema such as a table having columns and rows, a non-relational database schema, or other database schemas known to persons skilled in the art. In the database schema, data of a data owner may be structured as a record, represented by an element such as a tuple in a relational database schema; sensitive data of a data owner may be structured as an attribute of a record, represented by an element such as a column in a relational database schema. Attributes may be sensitive or non-sensitive based on their content and context. For example, individualized attributes such as age, income, and residential data of a data owner may be sensitive attributes, whereas computer operating system of a data owner, amount of

time actively logged in to a website by a data owner, or price of a purchase made by a data owner may be non-sensitive attributes. Alternatively, price of a purchase made by a data owner may be a sensitive attribute if the prices are very high, or if the purchases are of a personal nature. Attributes may be designated as sensitive attributes or non-sensitive attributes by data owners, or by a standard data schema defined at the data processing platform 100.

[0019] The database may store owned data received from data owners. Applications or services provided by a data processing platform 100 may receive data submitted by data owners. For example, data owners may submit owned data to a data processing platform 100 by operating a computing device to access a web-hosted graphical user interface or other web interfaces provided by a web server and suitable for importing data, viewing data, and submitting data by connecting to an Internet port of the web server. According to example embodiments of the present disclosure, the computing device may be a data processing platform 100, or the computing device may access a data processing platform 100 hosted by a server. The owned data may be data associated with existing activities of the data owner as a user of a service operated by an entity operating the data processing platform, such services including a social media network, an online retailer, a video streaming website, a photo-sharing website, a dating website and the like.

[0020] Table 1 is an example of a relational database schema for some owned data. The columns Age, Salary, and State are designated as sensitive attributes. The columns OS (Operating System), ActiveTime, and Purchase are designated as non-sensitive attributes. The rows t_1 – t_4 are individual tuples of the owned data.

	Age	Salary	State	OS	ActiveTime	Purchase
t_1	30	\$50,000	NY	Windows	1.6 h	\$120
t_2	60	\$80,000	WA	iOS	1.2 h	\$100
t_3	40	\$70,000	NY	Windows	1.0 h	\$100
t_4	40	\$70,000	NY	iOS	1.8 h	\$100

[0021] According to example embodiments of the present disclosure, owned data may be exposed to data collectors by queries that include multidimensional analysis (MDA) queries. An MDA query may be a database query in any suitable programming language, including query languages such as SQL, where the query includes an aggregate function performed over a measure attribute and having a predicate restricting output of the query to specified ranges of multiple other attributes. For example, in SQL, an aggregate function may be AVG, SUM, COUNT, and such functions that return one value summarizing, respectively, a mean value of an attribute of multiple tuples, a sum value of an attribute of multiple tuples, and a count of the number of multiple tuples. In SQL, a predicate may be a WHERE clause modifying an aggregate function that limits the tuples summarized by the aggregate function by multiple constraints over other attributes, leaving only those tuples having an attribute value equal to a specified value in the case of a categorical attribute, or only those tuples having an attribute value belonging to a specified range in the case of a ranged attribute. A categorical attribute or a ranged attribute may be a sensitive attribute.

[0022] Thus, an MDA query of the format described herein may be written as follows:

$$\text{SELECT } F(M) \text{ FROM } T \text{ WHERE } C$$

[0023] Herein, $F()$ is an aggregate function such as $\text{AVG}()$, $\text{SUM}()$, $\text{COUNT}()$, and the like; M is a measure attribute being aggregated over by $F()$; T is a table of a database; and C is a predicate specifying either a point constraint, where values of a particular attribute for tuples selected from T should equal a specified value, or a range constraint, where values of a particular attribute for tuples selected from T should belong to a specified range of values. For example, an MDA query for a table of Table 1 above may have the following format:

$$\begin{aligned} Q_SUM = \text{SELECT SUM(Purchase)} \text{ FROM } T \text{ WHERE Age} \in [30, 40] \text{ AND Salary} \\ \in [\$50,000, \$150,000] \end{aligned}$$

[0024] This query, if processed, would aggregate dollar amounts of all purchases made by users between the ages of 30 and 40, inclusive, having annual salary between \$50,000 and \$150,000, inclusive, and return the sum of the dollar amounts of all

purchases. Persons skilled in the art will appreciate that MDA queries such as those of the format described herein may be utilized to identify individuals using anonymized and aggregated data, and details of such utilization shall not be reiterated herein.

[0025] Thus, according to example embodiments of the present disclosure, the data processing platform 100 should not expose owned data in the database to data collectors in response to an MDA query having an aggregate function modified by a predicate over one or more sensitive attributes, and LDP being implemented at the data processing platform 100 should result in exposing data other than owned data in response to such an MDA query, such as owned data in an altered form.

[0026] According to example embodiments of the present disclosure, a data sharing platform 110 may be one or more services hosted on a network provided for a data processing platform 100. A data sharing platform 110 may be a hosted service which generates sharable data from owned data received from data owners, which in particular is sharable with data collectors querying the database. Sharable data may be at least in part not the same as owned data stored in the database; in particular, owned data and sharable data may differ in sensitive attributes of tuples. For example, sharable data may be a copy of owned data wherein sensitive attributes have been altered, such as by encoding. The data sharing platform 110 may generate a copy of sharable data, and may be a service of the data processing platform 100 exposing sharable data to data collectors, or may provide the sharable data to another service of the data processing platform 100 exposing sharable data to data collectors, such as a data analytics platform 120 as described herein.

[0027] The data sharing platform 110 may be one or more services hosted on a network provided for the data processing platform 100. The data sharing platform 110 may be hosted on a computer or web server, hosted distributed across multiple physically networked computers or web servers, hosted distributed across computers or networks over a physical or virtual cluster, or otherwise hosted by other network architectures providing hosting as known by persons skilled in the art. The hosting of the data sharing platform 110 may or may not be in common with the hosting of the data processing platform 100. The one or more services of the data sharing platform

may be stored in physical or virtual memory provided by the hosting of the data sharing platform 110, and may be executable by one or more physical or virtual processors provided by the hosting of the data analytics platform 110 to cause the one or more physical or virtual processors to perform functions of the data analytics platform 110 as described herein.

[0028] According to example embodiments of the present disclosure, the data sharing platform 110 having LDP implemented may further include a sharing query generator module 111 which may write a generated query which calls a user-defined function (UDF) 112 implemented at the data processing platform 100, where the UDF 112 is programmed according to an API of the data processing platform 100 to cause the data processing platform 100 to execute an algorithm A which takes a parameter ϵ according to LDP as a parameter; in other words, the UDF 112 is an implementation of an ϵ -LDP algorithm A .

[0029] A sharing query generator module 111 may call a UDF 112 with a parameter ϵ to encode each tuple of owned data specified by a UDF 112 based on a database schema of the owned data, resulting in the sharing query generator module 111 satisfying ϵ -LDP with regard to each respective individual tuple.

[0030] Because the UDF 112 is programmed using an API of the data processing platform 100, the UDF 112 may be executable by one or more physical or virtual processors provided by the hosting of the data processing platform 100 to cause the one or more physical or virtual processors to execute the UDF 112 as described herein. Execution of the UDF 112 may be enhanced by hosting on computer clusters or cloud computing servers to provide distributed computing, parallel computing, improved availability of physical or virtual computing resources, and such benefits.

[0031] Each tuple may be encoded by the sharing query generator module 111, wherein the sharing query generator module 111 writes a query that maps a tuple in owned data to an encoded tuple. This encoded tuple becomes part of sharable data and is exposed in place of the tuple in owned data in response to MDA queries to the data processing platform 100 that would aggregate over the tuple in owned data. Consequently, an MDA query by a data collector to the data processing platform 100

may cause a fact table containing tuples making up sharable data to be provided to the data collector in place of a fact table containing tuples making up owned data.

[0032] The generated query may be written in any suitable programming language as described with regard to MDA queries above, including any same programming language used by data collectors to submit queries to the data processing platform 100 such as a query language, or any other programming language. The generated query is written for a single tuple and selects solely that tuple as its output, and a generated query may be written individually for each tuple containing sensitive attributes among the owned data. The generated query may call the UDF 112 with ϵ and each sensitive attribute of a selected tuple of a selected tuple as parameters; the generated query may not call the UDF 112 with regard to non-sensitive attributes. Moreover, ϵ may be different for tuples received from different data owners.

[0033] Thus, the writing of generated queries may be dependent upon which attributes of a selected tuple are sensitive and which are non-sensitive. The generated query need not correspond to any query submitted by a data collector. The sharing query generator module 111 may write a generated query for a particular tuple in owned data independent of writing a generated query for any other tuple, and regardless of whether any query has been submitted by a data collector or not.

[0034] Algorithm A may be defined having a degree of randomness, resulting in outputs for two tuples to be different even when input values are the same over the same sensitive attributes for those two tuples. Moreover, algorithm A may be defined such that operations performed upon sensitive attributes that are categorical attributes differ from operations performed upon sensitive attributes that are ranged attributes.

[0035] For example, based on the database schema presented by Table 1 above, a sharing query generator module 111 may write a generated query for a tuple in owned data as follows:

```
SELECT LDP_Sharing_UDF(2.0, Age, Salary, State), OS, ActiveTime, Purchase  
FROM T
```

[0036] Herein, LDP_Sharing_UDF() is a call to the function name of a UDF 112, 2.0 is a value of the parameter ϵ , and T is a selected tuple in the owned data. The value

of ϵ and labels of the sensitive attributes Age, Salary, State are passed to the UDF 112 as parameters. The other attributes of the tuple T are not passed to the UDF 112. The UDF 112 may be defined such that operations performed upon Age and Salary, which are numerical and thus ranged attributes, differ from operations performed upon State, which is a categorical attribute.

[0037] It should be understood that an algorithm A of an ϵ -LDP UDF 112 applied to sensitive attributes of a tuple in owned data may cause operations to be performed upon the sensitive attributes guaranteeing that, if an encoded tuple mapped to the tuple in owned data is returned in response to MDA queries that would aggregate over the tuple in owned data, ϵ -LDP is satisfied with regard to the tuple. However, the algorithm A of an ϵ -LDP UDF 112 applied to sensitive attributes of a tuple in owned data may not guarantee that ϵ -LDP is satisfied with regard to any other tuple of the owned data. Further details of operations performed by an algorithm A need not be described herein for persons skilled in the art to obtain a full understanding of example embodiments of the present disclosure.

[0038] A data owner, while submitting data to the data processing platform 100, may allow the data sharing platform 110 to access owned data at the data processing platform 100 to cause the data sharing platform 110 to encode tuples of owned data. In the course of this operation, the UDF 112, irrespective of its nature as “user-defined,” may not be programmed or configured by the data owner; and may not be inspectable by the data owner, and the data owner may operate the data sharing platform 110 without knowledge of the content of the UDF 112 or the algorithm A. The algorithm A may be provided pre-defined as part of the one or more services provided by the data sharing platform 110 to the data processing platform 100, and the “user-defined” nature of the UDF 112 may merely refer to the UDF 112 being defined by a party using an API provided by the data processing platform. According to some example embodiments of the present disclosure, the parameter ϵ passed to the UDF 112 may be set based on input by the data owner, as illustrated by FIG. 2A, setting an extent of privacy guarantees desired for owned data.

[0039] According to example embodiments of the present disclosure, a data analytics platform 120 may be one or more services hosted on a network provided for a data processing platform 100. A data analytics platform 120 may be a hosted service which exposes sharable data to data collectors querying the database. For example, the data analytics platform 120 may expose owned data to data collectors by a web server providing a web-hosted graphical user interface, command line interface, SQL interface, application programming interface (API), or other web interfaces suitable for querying data upon being operated by a data collector connecting to an Internet port of the web server by operating a computing device.

[0040] Sharable data may be provided to the data analytics platform 120 by the data sharing platform 110, where shared data may be generated by the data sharing platform 100 as described above. For example, sharable data may be composed of encoded tuples mapped to each tuple of the owned data, encoded by the sharing query generator module 111 calling a UDF 112 with a parameter ϵ . The data analytics platform 120 may receive an MDA query input by a data collector in an interface as described above, and execute the input on the sharable data.

[0041] However, even if each tuple of the sharable data guarantees ϵ -LDP with each data of the owned data to which it is mapped to, post-processing may be further implemented at the data analytics platform 120 to alter an answer to the MDA query input by the data collector so that an estimated answer rather than an exact answer is returned. The data analytics platform 120 returning an estimated answer may not contribute to guaranteeing ϵ -LDP, but may provide another level of obfuscation of the returned data to further provide privacy.

[0042] The data analytics platform 120 may be one or more services hosted on a network provided for the data processing platform 100. The data analytics platform 120 may be hosted on a computer or web server, hosted distributed across multiple physically networked computers or web servers, hosted distributed across computers or networks over a physical or virtual cluster, or otherwise hosted by other network architectures providing hosting as known by persons skilled in the art. The hosting of the data analytics platform 120 may or may not be in common with the hosting of the

data processing platform 100 and/or the hosting of the data sharing platform 110. The one or more services of the data analytics platform may be stored in physical or virtual memory provided by the hosting of the data analytics platform 120, and may be executable by one or more physical or virtual processors provided by the hosting of the data analytics platform 120 to cause the one or more physical or virtual processors to perform functions of the data analytics platform 120 as described herein. Performance of such functions may be enhanced by hosting on computer clusters or cloud computing servers to provide distributed computing, parallel computing, improved availability of physical or virtual computing resources, and such benefits.

[0043] According to example embodiments of the present disclosure, the data analytics platform 120 may further include an MDA query rewriter module 121 which may rewrite an MDA query submitted by a data collector into a rewritten query that calls a user-defined aggregation function (UDAF) 122 implemented at the data processing platform 100, where the UDAF 122 is programmed according to an API of the data processing platform 100. The UDAF 122 may be programmed to implement an estimation algorithm $\bar{P}$ which takes the original query q and the encoded, sharable data $A(T)$ as parameters; the programming of such an algorithm according to an API is generally known to persons skilled in the art. Any number of different queries q from data collectors may be processed by the data analytics platform 120 for the same $A(T)$ generated by the data sharing platform 110.

[0044] The rewritten query may be written in any suitable programming language as described with regard to MDA queries above, including any same programming language used by data collectors to submit queries to the data processing platform 100 such as a query language, or any other programming language.

[0045] The data analytics platform 120 then causes the rewritten query calling the UDAF 122 to be executed by a data processing platform 100. A data processing platform 100 may or may not be a same data processing platform 100 in the context of the data sharing platform 110. In the case that the data processing platform 100 in the context of the data sharing platform 110 is one or more applications running on a computing device, the data sharing platform 100 in the context of the data analytics

platform 120 may be one or more services hosted by a server. In the case that the data processing platform 100 in the context of the data sharing platform 110 is one or more services hosted by one or more servers, the data processing platform 100 in the context of the data analytics platform 120 may be also among the one or more services hosted by the same one or more servers, or may be one or more services hosted by other servers. Because the UDAF 122 is programmed using an API of the data processing platform 100, the rewritten query may be executable by one or more physical or virtual processors provided by the hosting of the data processing platform 100 to cause the one or more physical or virtual processors to execute the UDAF 122 as described herein. Execution of the UDAF 122 may be enhanced by hosting on computer clusters or cloud computing servers to provide distributed computing, parallel computing, improved availability of physical or virtual computing resources, and such benefits.

[0046] In accordance with function call formats provided by an API of the data processing platform 100, which may support passing only individual tuples of $A(T)$ as function parameters rather than the entire table $A(T)$, the estimation algorithm $\bar{P}$ may be decomposed, where rather than the estimation algorithm $\bar{P}$ taking q and $A(T)$ (which may be referred to herein as $\bar{P}(q, A(T))$) as parameters directly, one iteration of $\bar{P}$ may be run for the same q and each different tuple of $A(T)$ (where any single tuple of $A(T)$ may be referred to herein as t_{ldp}), and an individual answer of each iteration of $\bar{P}$ ($\bar{P}(q, t_{ldp})$) may be merged to give a sum which is the answer of $\bar{P}(q, A(T))$ as a whole. An API call which performs each iteration of $\bar{P}$ may be a parallelizable API call, enabling iterations of $\bar{P}$ to be executed parallel to each other by the data processing platform 100, speeding up computation time required for the answer of each iteration of $\bar{P}$ to be merged.

[0047] Implementation of decomposition of the estimation algorithm $\bar{P}$ may be accomplished by, for example, creating a buffer data structure in memory, providing an iterating function that executes $\bar{P}$ for a query q and each tuple t_{ldp} and writes a partial answer to the buffer, and providing a merging function that reads the buffer and combines the partial answers to derive the answer of $\bar{P}(q, A(T))$. Other manners of decomposing the algorithm may be known to persons skilled in the art and shall fall

under the scope of the present disclosure as long as an answer to $\bar{P}(q, A(T))$ is derived by function calls in accordance with an API format where individual tuples t_{ldp} are passed to function calls rather than $A(T)$ as a whole. Alternatively, if the data processing platform 100 provides an API where $A(T)$ as a whole may be passed to function calls, $\bar{P}(q, A(T))$ may be programmed as executing a non-decomposed algorithm, although in this case parallelizing the computation of $\bar{P}(q, A(T))$ by decomposition or other algorithmic design may still be advantageous.

[0048] For example, based on the database schema presented by Table 1 above, an MDA query rewriter module 121 may rewrite an MDA query for a tuple in sharable data as follows:

SELECT LDP_Analytics_UDAF(ldp_tuple, Purchase, Q_SUM_Str) FROM ldp_T

[0049] Herein, LDP_Sharing_UDAF() is a call to the function name of a UDAF 122; ldp_tuple is each t_{ldp} ; Purchase is the particular measure attribute being aggregated over by the MDA query input by the data collector; Q_SUM_Str is the MDA query input by the data collector, which may be parsed by the UDAF 122; and ldp_T is $A(T)$. As ϵ -LDP is already guaranteed by the LDP implementation for $A(T)$, the sensitivity or non-sensitivity of attributes of each tuple T, and whether an attribute is categorical or ranged, need not be parameterized for the UDAF 122.

[0050] It should be understood that since the MDA query rewriting module 120 deals only with sharable data having ϵ -LDP guaranteed, the estimated answer may not alter whether ϵ -LDP is guaranteed with regard to the sharable data, but may merely further decrease the probability that differences in the data returned correlate to differences in sensitive data between particular individuals in the database. However, it remains desirable to notify a data collector of statistics regarding the accuracy of estimated answers, which may improve reliability of the data analytics platform 120 without compromising ϵ -LDP guarantees.

[0051] For each possible aggregate function called by the original MDA query, a confidence interval may be computed for an estimated answer in accordance with statistical methods. A confidence interval for an estimated answer to a query containing an aggregate function over a measure attribute may provide a range of values for that

measure attribute wherein the exact answer, which is not shown to a data collector, has, for example, a 90% chance of being located. In the case that the aggregate function is a COUNT function, the confidence interval may be derived from variance or (α, β) -accuracy. In the case that the aggregate function is a SUM function, the confidence interval may be derived from variance. In the case that the aggregate function is an AVG function (SUM / COUNT), the confidence interval may be derived from the SUM confidence interval divided by the COUNT confidence interval.

[0052] Moreover, an estimated answer to a query output by the estimation algorithm $\bar{P}$ may introduce some expected error to the answer over an exact answer to the same query; however, the expected error may be bounded to some extent. For example, expected error for the estimation algorithm $\bar{P}$ may be bounded to mean squared error, measured by the expected value of the square of a difference between an estimated answer and a corresponding exact answer.

[0053] Further details of operations performed by a UDAF 122 need not be described herein for persons skilled in the art to obtain a full understanding of example embodiments of the present disclosure.

[0054] A data collector, while requesting data from the data processing platform 100, may submit an MDA query using an interface provided by the data analytics platform 120, which is executed over sharable data provided by the data sharing platform 110 rather than owned data at the data processing platform 100. This causes the data analytics platform 120 to rewrite the MDA query and the data processing platform 100 to execute a UDAF 122 and return an estimated answer to the query in place of an exact answer. In the course of this operation, the UDAF 122, irrespective of its nature as “user-defined,” may not be programmed or configured by the data collector; and may not be inspectable by the data collector, and the data collector may operate the data analytics platform 120 without knowledge of the content of the UDAF 122. The UDAF 122 may be provided pre-defined as part of the one or more services provided by the data analytics platform 120 to the data processing platform 100, and the “user-defined” nature of the UDAF 122 may merely refer to the UDAF 122 being defined by a party using an API provided by the data processing platform.

[0055] FIG. 1 further illustrates a privacy boundary 130 conceptually organizing the data sharing platform 110 and the data analytics platform 120. On a private side of the privacy boundary 130 (illustrated in FIG. 1 as all elements left of the privacy boundary 130), data, including owned data and sharable data, is not exposed to data collectors; on a non-private side of the privacy boundary 130 (illustrated in FIG. 1 as all elements right of the privacy boundary), data is exposed to data collectors 130. The data sharing platform 110 is located left of the privacy boundary 130, and the data analytics platform is located right of the privacy boundary 130.

[0056] The privacy boundary 130 may be a conceptual boundary that need not correspond to any boundaries between hardware or software configurations, system hardware or software architecture, or physical location between computing devices running a data processing platform 100 and/or computers and/or servers hosting a data processing platform 100 and/or the database, and need not correspond to any boundaries between networks wherein one or more data processing platforms 100 are running or hosted and other networks such as local area networks, wide area networks, the Internet, and the like. For example, a data processing platform 100 executes the UDF 112 on the private side of the privacy boundary, and a data processing platform 100 executes the UDAF 122 on the non-private side of the privacy boundary. The privacy boundary 130 may correspond to sharable data, in the form of, for example, a fact table, being provided by the data sharing platform 110 to the data analytics platform 120 allowing processing of an MDA query over the sharable data resulting in aggregated data being returned to data collectors.

[0057] Although a data processing platform 100 is illustrated on both sides of the privacy boundary 130, a data processing platform 100 should not be understood as being exposed in its entirety on the non-private side of the privacy boundary 130. Rather, only those parts of a data processing platform 100 which perform execution of the UDAF 122, such as computational resources such as processors and memory and an API, need be exposed on the non-private side of the privacy boundary 130. Moreover, data processing platforms 100 on each side of the privacy boundary 130 may be a same data processing platform 100 or may be multiple data processing platforms 100, as

described above in the contexts of a data sharing platform 110 and a data analytics platform 120.

[0058] FIGS. 2A and 2B illustrate graphical user interfaces according to example embodiments of the present disclosure that receive data submitted by data owners. As illustrated by FIG. 2A, upon a data owner operating a computing device to access the data processing platform 100, the data sharing platform 110 may communicate with the computing device of the data owner to cause the computing device to display an encoding interface 210 which the data owner may further operate using the computing device. The encoding interface 210 is a graphical user interface displaying owned data imported into the encoding interface 210, such as data of a data owner, in a tabular view 211. Sensitive attributes 212 (for example, Age, Salary, and State, according to the database schema of Table 1) may be visually highlighted in any suitable manner that distinguishes them from non-sensitive attributes 213. Furthermore, the encoding interface 210 provides an input control 214 which accepts input by a data owner of a privacy parameter representing a desired extent of a privacy guarantee. For example, according to the example embodiment illustrated by FIG. 2A, the input control 214 may accept an input of a numerical value of a parameter ϵ directly. According to other example embodiments of the present disclosure, the input control 214 may accept another form of ranged input that scale to different values of a parameter ϵ , such as options or a dial control corresponding to high or low degrees of privacy guarantees. Furthermore, the encoding interface 210 provides an encoding control 215 which a data owner may operate to encode the data viewed in a tabular view 211, upon satisfaction of the data owner that the data viewed reflects data desired to be encoded prior to sharing with data collectors and the input at the input control 214 reflects a desired extent of privacy guarantees to be implemented through the encoding.

[0059] As illustrated by FIG. 2B, upon the data owner operating the computing device to operate the encoding control 215, the data sharing platform 110 may communicate with the computing device to cause the computing device to display a submitting interface 220. The submitting interface 220 displays encoded, sharable data to be submitted, in a tabular view 221 similar to the tabular view 211 of the encoding

interface 210. Sensitive attributes (for example, Age, Salary, and State, according to the database schema of Table 1) may be visually omitted from the tabular view 221 and replaced with an encoded column 222 showing, for each tuple, a representation of results of encoding sensitive attributes of that tuple. Non-sensitive attributes 223 are shown unchanged from the tabular view 211. Furthermore, the submitting interface 220 provides a submission control 224 which a data owner may operate to submit the data viewed in tabular form 221, upon satisfaction of the data owner that the sharable data viewed reflects data desired to be shared with data collectors.

[0060] FIG. 3 illustrates a graphical user interface according to example embodiments of the present disclosure that processes an MDA query input by a data collector. Upon a data collector operating a computing device to access the data processing platform 100 to request sharable data, the data analytics platform 120 may communicate with the computing device of the data owner to cause the computing device to display a querying interface 310 which the data owner may further operate using the computing device. The querying interface 310 is a graphical user interface displaying all sharable data requested by the data collector, which may include data of multiple data owners, in a tabular view 311. The tabular view 311 may be similar to that of the tabular view 221 of the submitting interface 220, where sensitive attributes are replaced with an encoded column 222 (this view not being illustrated in FIG. 3, due to redundancy with FIG. 2B).

[0061] Furthermore, the querying interface 310 provides a query input control 312 which accepts input by a data collector of queries including MDA queries as described above, including an aggregate function performed over a measure attribute and having a predicate restricting output of the query to specified ranges of multiple other attributes. Attributes referenced in queries may be in accordance with the database scheme for the owned data, including attributes present in the owned data but encoded in the sharable data and not expressly shown in the tabular views 221 and 311.

[0062] Upon the data collector submitting an MDA query input into the query input control 312 and operating a query issue control 313, the MDA query rewriter module 121 rewrites the MDA query into a query that calls a UDAF 122 implemented at the

data processing platform 100, where the UDAF 122 is programmed according to an API of the data processing platform 100. The data analytics platform 120 then causes the rewritten query calling the UDAF 122 to be executed by the data processing platform 100.

[0063] The querying interface 310 then displays an estimated answer returned by the data processing platform 100 and a confidence interval computed for the estimated answer as described above.

[0064] The querying interface 310 also updates the tabular view 311 to cause the tabular view 311 to display a contribution column 313, which shows contribution of each tuple of the sharable data $\bar{P}(q, t_{lap})$ to the summed estimated answer of $\bar{P}(q, A(T))$ as a whole. Contribution of two tuples to be different even when those two tuples have the same respective values for the measure attribute the same sensitive attributes, further decreasing the probability that differences in the data returned correlate to differences in sensitive data between particular individuals in the database.

[0065] FIG. 4 illustrates a flowchart of a data sharing method 400 according to example embodiments of the present disclosure.

[0066] At step 402, a data owner shares owned data with a data processing platform. The data owner may operate a computing device to access a web-hosted graphical user interface or other web interfaces provided by a web server and suitable for importing data, viewing data, and submitting data by connecting to an Internet port of the web server. The owned data may be data associated with existing activities of the data owner as a user of a service operated by an entity operating the data processing platform, such services including a social media network, an online retailer, a video streaming website, a photo-sharing website, a dating website and the like.

[0067] At step 404, the data owner operates an encoding interface provided by a data sharing platform to cause the data processing platform to encode the owned data and generate sharable data. The encoding interface may be a graphical user interface displaying, on the computing device of the data owner, owned data imported into the encoding interface in a tabular interview as described above. The data owner may, furthermore, view the owned data in the encoding interface and input a privacy

parameter representing a desired extent of a privacy guarantee in the encoding interface prior to causing the data sharing platform to encode the owned data.

[0068] At step 406, the data owner operates a submitting interface provided by the data sharing platform to cause the data sharing platform to submit the sharable data to the data processing platform. The submitting interface may be a graphical user interface displaying, on the computing device of the data owner, sharable data generated by the data sharing platform in a tabular interview as described above. The data owner may, furthermore, view the sharable data in the submitting interface to the satisfaction of the data owner that the sharable data viewed reflects data desired to be shared with data collectors prior to causing the data sharing platform to submit the sharable data to the data processing platform.

[0069] FIG. 5 illustrates a flowchart of a data analytics method 500 according to example embodiments of the present disclosure.

[0070] At step 502, a data collector requests data of one or more data owners from a data processing platform. The data owner may operate a computing device to access the data processing platform to request sharable data. The owned data may be data associated with existing activities of one or more data owner as a user of a service operated by an entity operating the data processing platform, such services including a social media network, an online retailer, a video streaming website, a photo-sharing website, a dating website and the like.

[0071] At step 504, the data collector operates a querying interface provided by a data analytics platform to cause the data processing platform to input an MDA query into a query input control. As described above, an MDA query includes an aggregate function performed over a measure attribute and having a predicate restricting output of the query to specified ranges of multiple other attributes. The querying interface may be a graphical user interface displaying, on the computing device of the data owner, sharable data generated by encoding owned data in a tabular interview as described above. The data collector may, furthermore, view the sharable data in the encoding interface.

[0072] The querying interface then displays an estimated answer returned by the data processing platform and a confidence interval computed for the estimated answer as described above. The querying interface also updates the tabular view to cause the tabular view to display a contribution column. The data collector may review this information and may input another query if desired.

[0073] FIG. 6 illustrates a flowchart of a data sharing and data analytics processing method 600 at a data sharing platform and a data analytics platform hosted as services for a data processing platform according to example embodiments of the present disclosure.

[0074] At step 602, a data sharing platform receives owned data submitted by a data owner to a data processing platform.

[0075] At step 604, a sharing query generator module of the data sharing platform writes a generated query which calls a user-defined function executable by the data processing platform to cause the data processing platform to generate an encoded tuple from each tuple of the owned data, and maps each encoded tuple returned by the user-defined function to the respective tuple of the owned data.

[0076] At step 606, a data analytics platform receives a request for the owned data from a data collector.

[0077] At step 608, the data sharing platform provides sharable data to a data analytics platform, the sharable data being composed of each encoded tuple mapped to a tuple of the owned data.

[0078] At step 610, the data analytics platform receives the sharable data from the data sharing platform.

[0079] At step 612, the data analytics platform receives an MDA query from the data collector having an aggregate function over the owned data.

[0080] At step 614, an MDA query rewriter module of the data analytics platform rewrites the MDA query into a rewritten query which calls a user-defined aggregation function executable by the data processing platform to cause the data processing platform to generate an estimated answer to the rewritten query over the sharable data.

[0081] At step 616, the data analytics platform receives the estimated answer from the data processing platform.

[0082] At step 618, the data analytics platform outputs the estimated answer.

[0083] Each of the above steps may be performed as described above with reference to the other Figures of the present disclosure and may further include optional steps as described above, which shall not be reiterated herein.

[0084] FIG. 7 illustrates a flowchart of a data sharing and data analytics processing method 700 at a data processing platform according to example embodiments of the present disclosure.

[0085] At step 702, a data processing platform receives owned data from a data sharing platform.

[0086] At step 704, the data processing platform executes a user-defined function causing the data processing platform to generate an encoded tuple from each tuple of the owned data.

[0087] At step 706, the data processing platform returns each encoded tuple to the data sharing platform.

[0088] At step 708, the data processing platform receives sharable data from a data analytics platform.

[0089] At step 710, the data processing platform receives a rewritten query from the data analytics platform.

[0090] At step 712, the data processing platform executes a user-defined aggregation function causing the data processing platform to generate an estimated answer to the rewritten query over the sharable data.

[0091] At step 714, the data processing platform returns the estimated answer to the data sharing platform.

[0092] FIG. 8 illustrates an example system 800 for implementing the processes and methods described above for implementing a data sharing platform and data analytics platform.

[0093] The techniques and mechanisms described herein may be implemented by multiple instances of the system 800, as well as by any other computing device, system,

and/or environment. The system 800 may be a distributed system composed of multiple physically networked computers or web servers, a physical or virtual cluster, a computing cloud, or other networked computing architectures providing physical or virtual computing resources as known by persons skilled in the art. The system 800 shown in FIG. 8 is only one example of a system and is not intended to suggest any limitation as to the scope of use or functionality of any computing device utilized to perform the processes and/or procedures described above. Other well-known computing devices, systems, environments and/or configurations that may be suitable for use with the embodiments include, but are not limited to, personal computers, server computers, hand-held or laptop devices, multiprocessor systems, microprocessor-based systems, set top boxes, game consoles, programmable consumer electronics, network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, implementations using field programmable gate arrays (“FPGAs”) and application specific integrated circuits (“ASICs”), and/or the like.

[0094] The system 800 may include one or more processors 802 and system memory 804 communicatively coupled to the processor(s) 802. The processor(s) 802 and system memory 804 may be physical or may be virtualized and/or distributed. The processor(s) 802 may execute one or more modules and/or processes to cause the processor(s) 802 to perform a variety of functions. In embodiments, the processor(s) 802 may include a central processing unit (CPU), a graphics processing unit (GPU), both CPU and GPU, or other processing units or components known in the art. Additionally, each of the processor(s) 802 may possess its own local memory, which also may store program modules, program data, and/or one or more operating systems.

[0095] Depending on the exact configuration and type of the system 800, the system memory 804 may be volatile, such as RAM, non-volatile, such as ROM, flash memory, miniature hard drive, memory card, and the like, or some combination thereof. The system memory 804 may include one or more computer-executable modules 806 that are executable by the processor(s) 802. The modules 806 may be hosted on a network

as services for a data processing platform, which may be implemented on a separate system from the system 800.

[0096] The modules 806 may include, but are not limited to, an owned data storage module 808, a sharing query generator module 810, an encoding interface module 812, a submitting interface module 814, a sharable data storage module 816, an MDA query rewriter module 818, and a querying interface module 820. The owned data storage module 808, the sharing query generator module 810, the encoding interface module 812, and the submitting interface module 814 may make up services provided by a data sharing platform, a logical organization of services providing related functions. The sharable data storage module 816, the MDA query rewriter module 818, and the querying interface module 820 may make up services provided by a data analytics platform, a logical organization of services providing related functions. The services of the data sharing platform and the services of the data analytics platform may be hosted on different instances of the system 800.

[0097] The owned data storage module 808 may be configured to store owned data received from a data owner.

[0098] The sharing query generator module 810 may be configured to write a generated query which calls a user-defined function 811 executable by the data processing platform to cause the data processing platform to generate an encoded tuple from each tuple of the owned data, and map each encoded tuple returned by the user-defined function 811 to the respective tuple of the owned data.

[0099] The encoding interface module 812 may be configured to display a web-hosted encoding interface operable on a computing device in network communication with the hosted data sharing platform services. The encoding interface module 812 may be further configured to display owned data imported into the encoding interface; accept input by a data owner of a privacy parameter representing a desired extent of a privacy guarantee; and receive an instruction to encode the displayed data. The encoding interface module 812 may be further configured to provide the privacy parameter to the sharing query generator module 810 as parameters for the user-defined function 811 and cause the sharing query generator module 810 to write a generated query.

[00100] The submitting interface module 814 may be configured to display a web-hosted encoding interface operable on a computing device in network communication with the hosted data sharing platform services. The submitting interface module 814 may be further configured to display encoded data received from the data processing platform; and receive an instruction to submit the displayed data. The submitting interface module 814 may be further configured to cause the encoded data to be submitted to the data processing platform as sharable data.

[00101] The sharable data storage module 816 may be configured to store sharable data received from the data sharing platform.

[00102] The MDA query rewriter module 818 may be configured to rewrite the MDA query into a rewritten query which calls a user-defined aggregation function 819 executable by the data processing platform to cause the data processing platform to generate an estimated answer to the rewritten query over the sharable data.

[00103] The querying interface module 820 may be configured to display a web-hosted encoding interface operable on a computing device in network communication with the hosted data analytics platform services. The querying interface module 820 may be further configured to display sharable data on a querying interface; accept input by a data collector of an MDA query over owned data; and display an estimated answer to a rewritten query over the sharable data. The encoding interface module 820 may be further configured to provide the MDA query to the MDA query rewriter module 818 and cause the MDA query rewriter generator module 818 to rewrite the MDA query.

[00104] The system 800 may additionally include an input/output (I/O) interface 840 and a communication module 850 allowing the system 800 to communicate with other systems and devices over a network, such as the data processing platform, a computing device of a data owner, and a computing device of a data collector. The network may include the Internet, wired media such as a wired network or direct-wired connections, and wireless media such as acoustic, radio frequency (RF), infrared, and other wireless media.

[00105] Some or all operations of the methods described above can be performed by execution of computer-readable instructions stored on a computer-readable storage

medium, as defined below. The term “computer-readable instructions” as used in the description and claims, include routines, applications, application modules, program modules, programs, components, data structures, algorithms, and the like. Computer-readable instructions can be implemented on various system configurations, including single-processor or multiprocessor systems, minicomputers, mainframe computers, personal computers, hand-held computing devices, microprocessor-based, programmable consumer electronics, combinations thereof, and the like.

[00106] The computer-readable storage media may include volatile memory (such as random-access memory (RAM)) and/or non-volatile memory (such as read-only memory (ROM), flash memory, etc.). The computer-readable storage media may also include additional removable storage and/or non-removable storage including, but not limited to, flash memory, magnetic storage, optical storage, and/or tape storage that may provide non-volatile storage of computer-readable instructions, data structures, program modules, and the like.

[00107] A non-transient computer-readable storage medium is an example of computer-readable media. Computer-readable media includes at least two types of computer-readable media, namely computer-readable storage media and communications media. Computer-readable storage media includes volatile and non-volatile, removable and non-removable media implemented in any process or technology for storage of information such as computer-readable instructions, data structures, program modules, or other data. Computer-readable storage media includes, but is not limited to, phase change memory (PRAM), static random-access memory (SRAM), dynamic random-access memory (DRAM), other types of random-access memory (RAM), read-only memory (ROM), electrically erasable programmable read-only memory (EEPROM), flash memory or other memory technology, compact disk read-only memory (CD-ROM), digital versatile disks (DVD) or other optical storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other non-transmission medium that can be used to store information for access by a computing device. In contrast, communication media may embody computer-readable instructions, data structures, program modules, or other data in a

modulated data signal, such as a carrier wave, or other transmission mechanism. As defined herein, computer-readable storage media do not include communication media.

[00108] The computer-readable instructions stored on one or more non-transitory computer-readable storage media that, when executed by one or more processors, may perform operations described above with reference to FIGS. 1-8. Generally, computer-readable instructions include routines, programs, objects, components, data structures, and the like that perform particular functions or implement particular abstract data types. The order in which the operations are described is not intended to be construed as a limitation, and any number of the described operations can be combined in any order and/or in parallel to implement the processes.

[00109] By the abovementioned technical solutions, the present disclosure provides a data sharing platform providing services to a data processing platform and a data analytics platform providing services to a data processing platform, providing ϵ -LDP guarantees over privacy of owned data received from data owners by decreasing the probability that differences in the data returned correlate to differences in sensitive data between particular individuals in the database. The one or more services of the data sharing platform and the data analytics platform include functions executable by the data processing platform, allowing services to be provided guaranteeing LDP in a self-enforcing manner between data owners and non-trusted data collectors, and the services to be scaled to computing resources and data throughput of the data processing platform itself, as well as taking advantage of distributed computing, parallel computing, improved availability of physical or virtual computing resources, and such benefits that the data processing platform may provide. Moreover, functions executable by the data processing platform may execute decomposed algorithms, speeding up computation time required to derive answers to queries.

EXAMPLE CLAUSES

[00110] A. A method comprising: a data owner sharing owned data to a data processing platform; the data owner operating an encoding interface provided by a data sharing platform to cause the data processing platform to encode the owned data and generate sharable data; and the data owner operating a submitting interface provided by

the data sharing platform to cause the data sharing platform to submit the sharable data to the data processing platform.

[00111] B. The method as paragraph A recites, further comprising: the data owner viewing the owned data in the encoding interface and inputting a privacy parameter representing a desired extent of a privacy guarantee in the encoding interface prior to causing the data sharing platform to encode the owned data.

[00112] C. The method as paragraph A recites, further comprising: the data owner viewing the sharable data in the submitting interface prior to causing the data sharing platform to submit the sharable data to the data processing platform.

[00113] D. A method comprising: a data collector requesting data of one or more data owners from a data processing platform; and the data collector operating a querying interface provided by a data analytics platform to cause the data processing platform to input an MDA query into a query input control.

[00114] E. A method comprising: a data sharing platform receiving owned data submitted by a data owner to a data processing platform; a sharing query generator module of the data sharing platform writing a generated query; a data analytics platform receiving a request for the owned data from a data collector; and the data sharing platform providing sharable data to a data analytics platform.

[00115] F. The method as paragraph E recites, wherein the generated query calls a user-defined function executable by the data processing platform to cause the data processing platform to generate an encoded tuple from each tuple of the owned data.

[00116] G. The method as paragraph F recites, wherein the user-defined function has a parameter ϵ in accordance with local differential privacy ("LDP"), and each encoded tuple satisfies ϵ -LDP with regard to the respective tuple of the owned data based on at least one attribute of the respective tuple being a sensitive attribute.

[00117] H. The method as paragraph G recites, wherein operations of the user-defined function performed upon sensitive attributes that are categorical attributes differ from operations of the user-defined function performed upon sensitive attributes that are ranged attributes.

[00118] I. The method as paragraph F recites, further comprising the data sharing platform mapping each encoded tuple returned by the user-defined function to the respective tuple of the owned data.

[00119] J. The method as paragraph I recites, wherein the sharable data comprises each encoded tuple mapped to a tuple of the owned data.

[00120] K. The method as paragraph F recites, wherein the data processing platform is configured to execute the user-defined function on a cloud computing server.

[00121] L. The method as paragraph E recites, further comprising the data analytics platform receiving the sharable data from the data sharing platform; the data analytics platform receiving an MDA query from the data collector having an aggregate function over the owned data; an MDA query rewriter module of the data analytics platform rewriting the MDA query into a rewritten query; and the data analytics platform receiving the estimated answer from the data processing platform.

[00122] M. The method as paragraph L recites, wherein the rewritten query calls a user-defined aggregation function executable by the data processing platform to cause the data processing platform to generate an estimated answer to the rewritten query over the sharable data.

[00123] N. The method as paragraph M recites, wherein the user-defined aggregation function implements an estimation algorithm having the MDA query and the sharable data as parameters.

[00124] O. The method as paragraph N recites, wherein the estimation algorithm is decomposed to be executed by the user-defined function running in an iteration for each tuple of the sharable data.

[00125] P. The method as paragraph O recites, wherein the estimated answer is a sum of an answer of each iteration of the estimated algorithm.

[00126] Q. The method as paragraph O recites, wherein each iteration of the estimation algorithm is executable at least in part in parallel with at least one other iteration of the estimation algorithm.

[00127] R. The method as paragraph N recites, wherein the estimation algorithm has an expected error value bounded to a mean squared error value.

[00128] S. The method as paragraph N recites, wherein the data processing platform is configured to execute the user-defined aggregate function on a cloud computing server.

[00129] T. The method as paragraph L recites, further comprising the data analytics platform computing confidence interval for the estimated answer.

[00130] U. A method comprising: a data processing platform receiving owned data from a data sharing platform; the data processing platform executing a user-defined function causing the data processing platform to generate an encoded tuple from each tuple of the owned data; and the data processing platform returning each encoded tuple to the data sharing platform.

[00131] V. The method as paragraph U recites, wherein the user-defined function has a parameter ϵ in accordance with local differential privacy ("LDP"), and each encoded tuple satisfies ϵ -LDP with regard to the respective tuple of the owned data based on at least one attribute of the respective tuple being a sensitive attribute.

[00132] W. The method as paragraph V recites, wherein operations of the user-defined function performed upon sensitive attributes that are categorical attributes differ from operations of the user-defined function performed upon sensitive attributes that are ranged attributes.

[00133] X. The method as paragraph U recites, wherein the data processing platform executes the user-defined function on a cloud computing server.

[00134] Y. The method as paragraph U recites, further comprising the data processing platform receiving sharable data from a data analytics platform; the data processing platform executing a user-defined aggregation function causing the data processing platform to generate an estimated answer to the rewritten query over the sharable data; and the data processing platform returning the estimated answer to the data sharing platform.

[00135] Z. The method as paragraph Y recites, wherein the user-defined aggregation function implements an estimation algorithm having the MDA query and the sharable data as parameters.

[00136] AA. The method as paragraph Z recites, wherein the estimation algorithm is decomposed to be executed by the user-defined aggregation function running in an iteration for each tuple of the sharable data.

[00137] BB. The method as paragraph AA recites, wherein the estimated answer is a sum of an answer of each iteration of the estimated algorithm.

[00138] CC. The method as paragraph AA recites, wherein each iteration of the estimation algorithm is executable at least in part in parallel with at least one other iteration of the estimation algorithm.

[00139] DD. The method as paragraph Y recites, wherein the data processing platform executes the user-defined function on a cloud computing server.

[00140] EE. A system comprising: one or more processors; and memory communicatively coupled to the one or more processors, the memory storing computer-executable modules executable by the one or more processors that, when executed by the one or more processors, perform associated operations, the computer-executable modules comprising: a sharing query generator module configured to write a generated query which calls a user-defined function executable by a data processing platform to cause the data processing platform to generate an encoded tuple from each tuple of owned data, and map each encoded tuple returned by the user-defined function to the respective tuple of the owned data; and an MDA query rewriter module configured to rewrite the MDA query into a rewritten query which calls a user-defined aggregation function executable by the data processing platform to cause the data processing platform to generate an estimated answer to the rewritten query over sharable data, the sharable data comprising encoded tuple each mapped to a tuple of the owned data.

[0141] Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described. Rather, the specific features and acts are disclosed as exemplary forms of implementing the claims.

CLAIMS

WHAT IS CLAIMED IS:

1. A method comprising:
receiving, by a data sharing platform, owned data submitted by a data owner to a data processing platform;
writing, by a sharing query generator module of the data sharing platform, a generated query;
receiving, by a data analytics platform, a request for the owned data from a data collector; and
providing, by the data sharing platform, sharable data to a data analytics platform.
2. The method of claim 1, wherein the generated query calls a user-defined function executable by the data processing platform to cause the data processing platform to generate an encoded tuple from each tuple of the owned data.
3. The method of claim 2, wherein the user-defined function has a parameter ϵ in accordance with local differential privacy ("LDP"), and each encoded tuple satisfies ϵ -LDP with regard to the respective tuple of the owned data based on at least one attribute of the respective tuple being a sensitive attribute.
4. The method of claim 3, wherein operations of the user-defined function performed upon sensitive attributes that are categorical attributes differ from operations of the user-defined function performed upon sensitive attributes that are ranged attributes.
5. The method of claim 2, further comprising mapping, by the data sharing platform, each encoded tuple returned by the user-defined function to the respective tuple of the owned data.

6. The method of claim 5, wherein the sharable data comprises each encoded tuple mapped to a tuple of the owned data.

7. The method of claim 1, further comprising:
receiving, by the data analytics platform, the sharable data from the data sharing platform;
receiving, by the data analytics platform, an MDA query from the data collector having an aggregate function over the owned data;
rewriting, by an MDA query rewriter module of the data analytics platform, the MDA query into a rewritten query; and
receiving, by the data analytics platform, the estimated answer from the data processing platform.

8. The method of claim 7, wherein the rewritten query calls a user-defined aggregation function executable by the data processing platform to cause the data processing platform to generate an estimated answer to the rewritten query over the sharable data.

9. The method of claim 8, wherein the user-defined aggregation function implements an estimation algorithm having the MDA query and the sharable data as parameters.

10. The method of claim 9, wherein the estimated answer is a sum of an answer of each iteration of the estimated algorithm.

11. The method of claim 8, wherein each iteration of the estimation algorithm is executable at least in part in parallel with at least one other iteration of the estimation algorithm.

12. A method comprising:

receiving, by a data processing platform, owned data from a data sharing platform;

executing, by the data processing platform, a user-defined function causing the data processing platform to generate an encoded tuple from each tuple of the owned data; and

returning, by the data processing platform, each encoded tuple to the data sharing platform.

13. The method of claim 12, wherein the user-defined function has a parameter ϵ in accordance with local differential privacy (“LDP”), and each encoded tuple satisfies ϵ -LDP with regard to the respective tuple of the owned data based on at least one attribute of the respective tuple being a sensitive attribute.

14. The system of claim 13, wherein operations of the user-defined function performed upon sensitive attributes that are categorical attributes differ from operations of the user-defined function performed upon sensitive attributes that are ranged attributes.

15. The method of claim 12, further comprising:

receiving, by the data processing platform, sharable data from a data analytics platform;

executing, by the data processing platform, a user-defined aggregation function causing the data processing platform to generate an estimated answer to the rewritten query over the sharable data; and

returning, by the data processing platform, the estimated answer to the data sharing platform.

16. The system of claim 15, wherein the user-defined aggregation function implements an estimation algorithm having the MDA query and the sharable data as parameters.

17. The method of claim 16, wherein the estimation algorithm is decomposed to be executed by the user-defined aggregation function running in an iteration for each tuple of the sharable data.

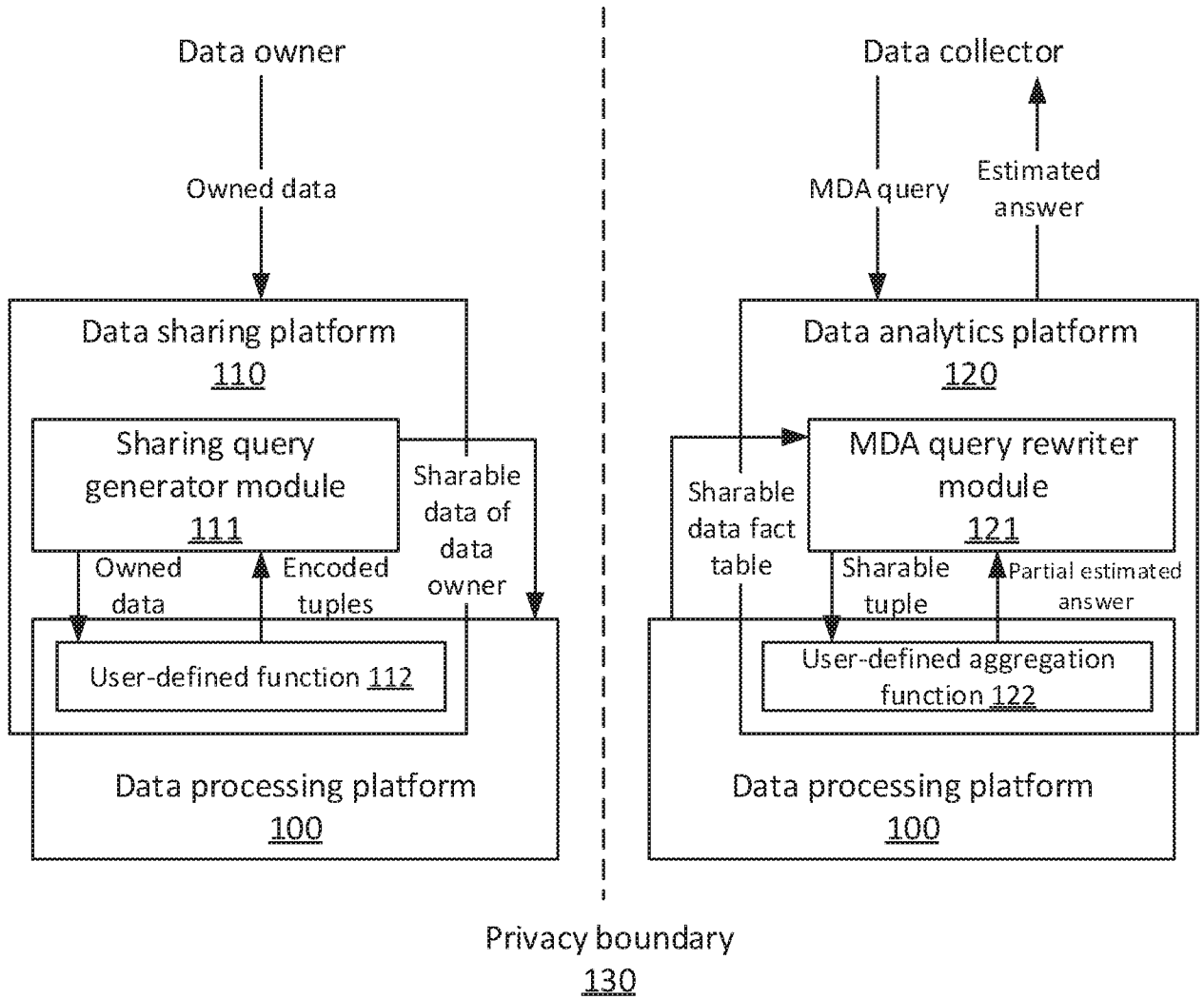
18. The method of claim 17, wherein the estimated answer is a sum of an answer of each iteration of the estimated algorithm.

19. The method of claim 17, wherein each iteration of the estimation algorithm is executable at least in part in parallel with at least one other iteration of the estimation algorithm.

20. A system comprising:
one or more processors; and
memory communicatively coupled to the one or more processors, the memory storing computer-executable modules executable by the one or more processors that, when executed by the one or more processors, perform associated operations, the computer-executable modules comprising:

a sharing query generator module configured to write a generated query which calls a user-defined function executable by a data processing platform to cause the data processing platform to generate an encoded tuple from each tuple of owned data, and map each encoded tuple returned by the user-defined function to the respective tuple of the owned data; and

an MDA query rewriter module configured to rewrite the MDA query into a rewritten query which calls a user-defined aggregation function executable by the data processing platform to cause the data processing platform to generate an estimated answer to the rewritten query over sharable data, the sharable data comprising encoded tuple each mapped to a tuple of the owned data.

**FIG. 1**

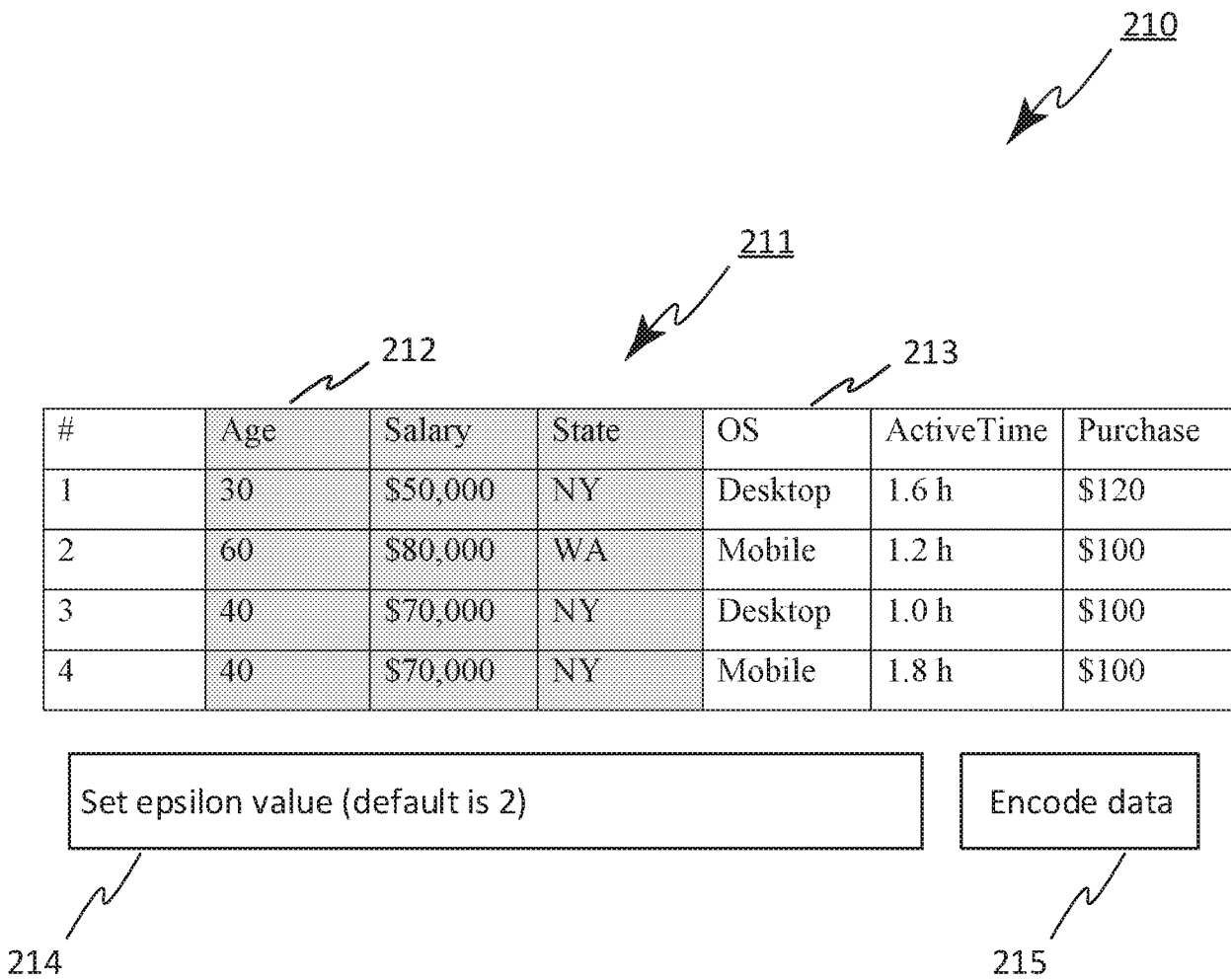


FIG. 2A

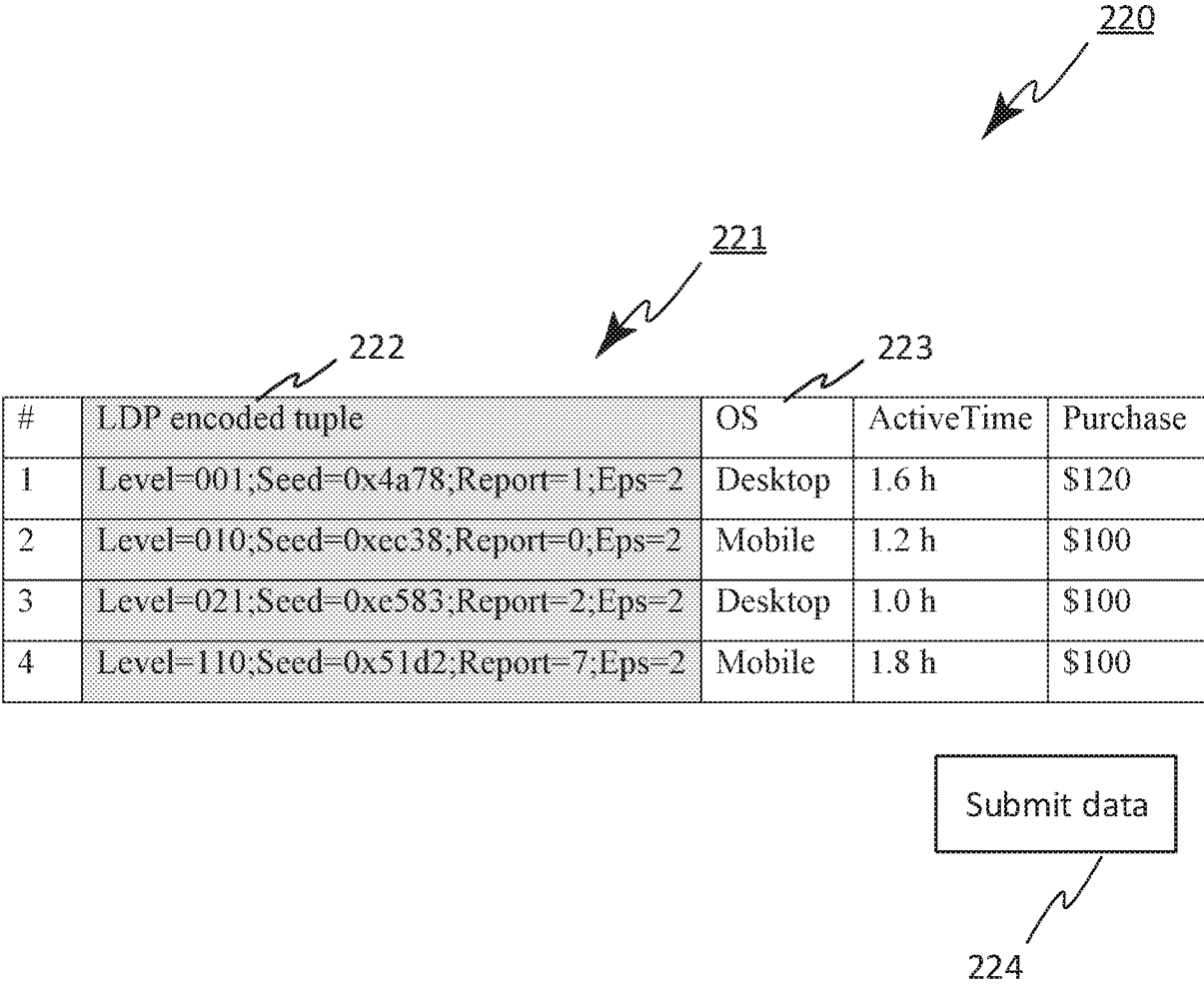


FIG. 2B

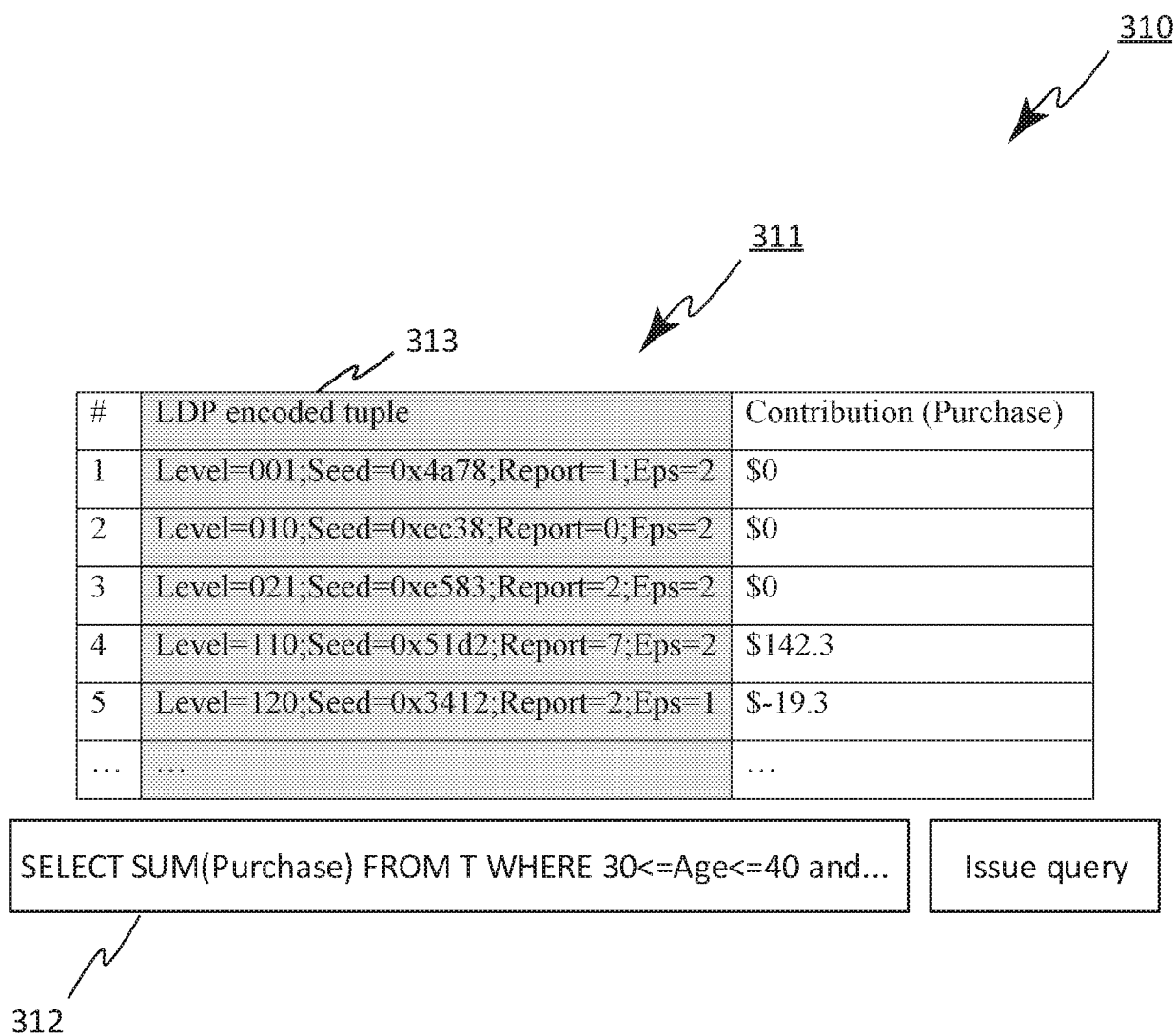


FIG. 3

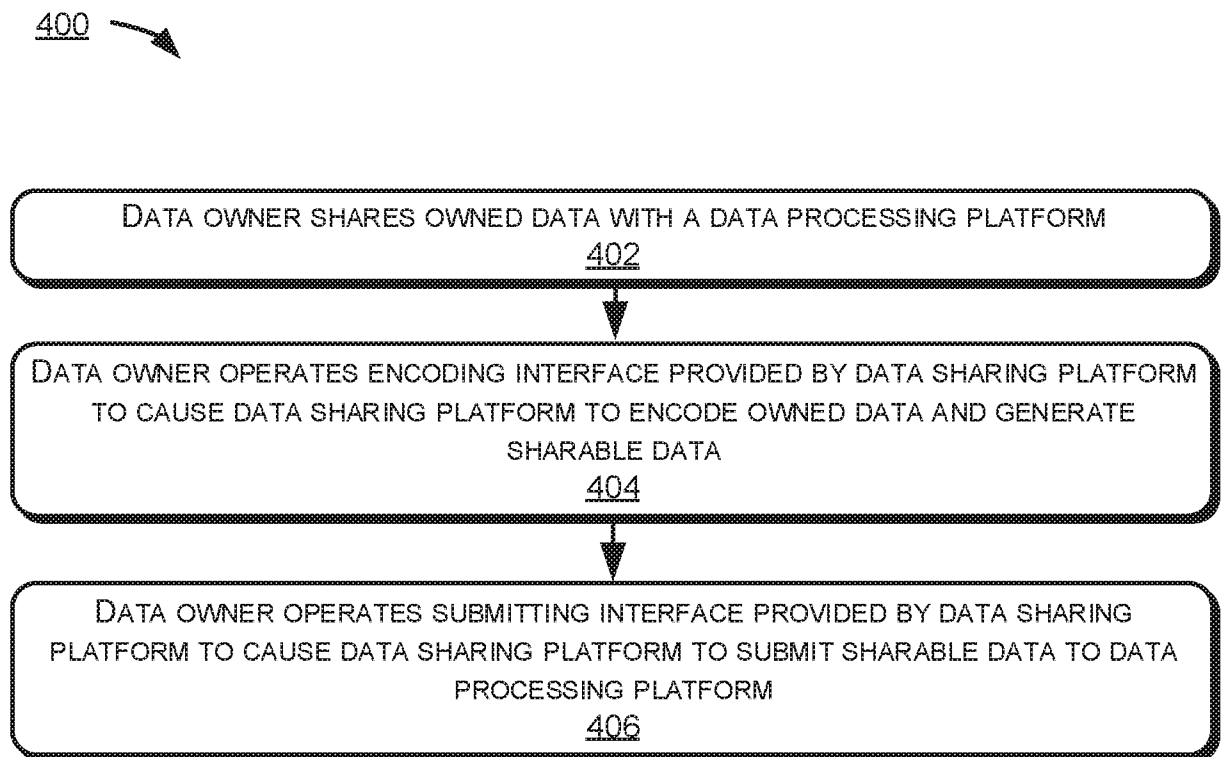
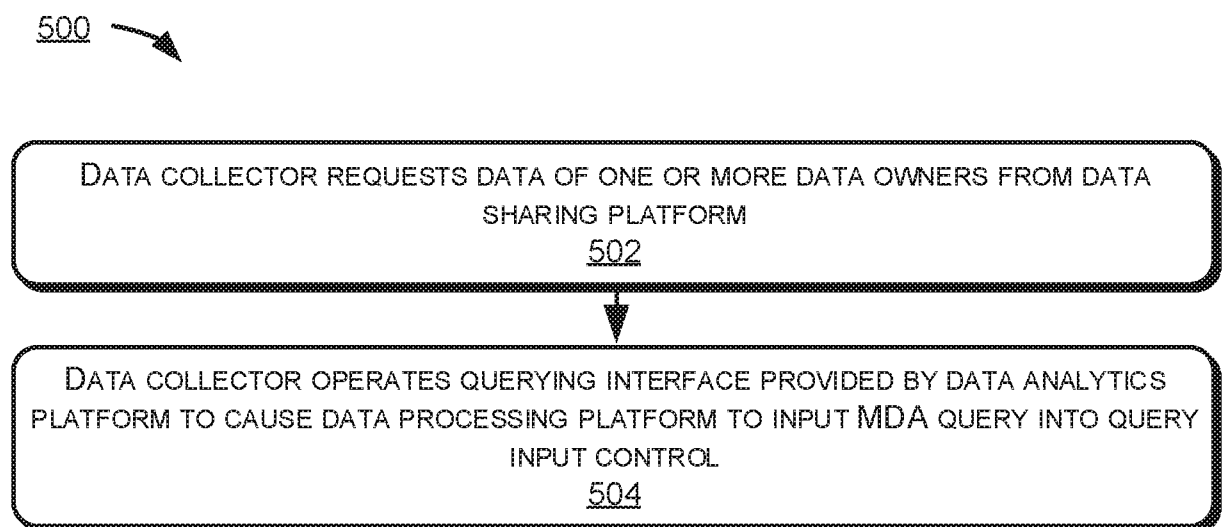


FIG. 4

**FIG. 5**

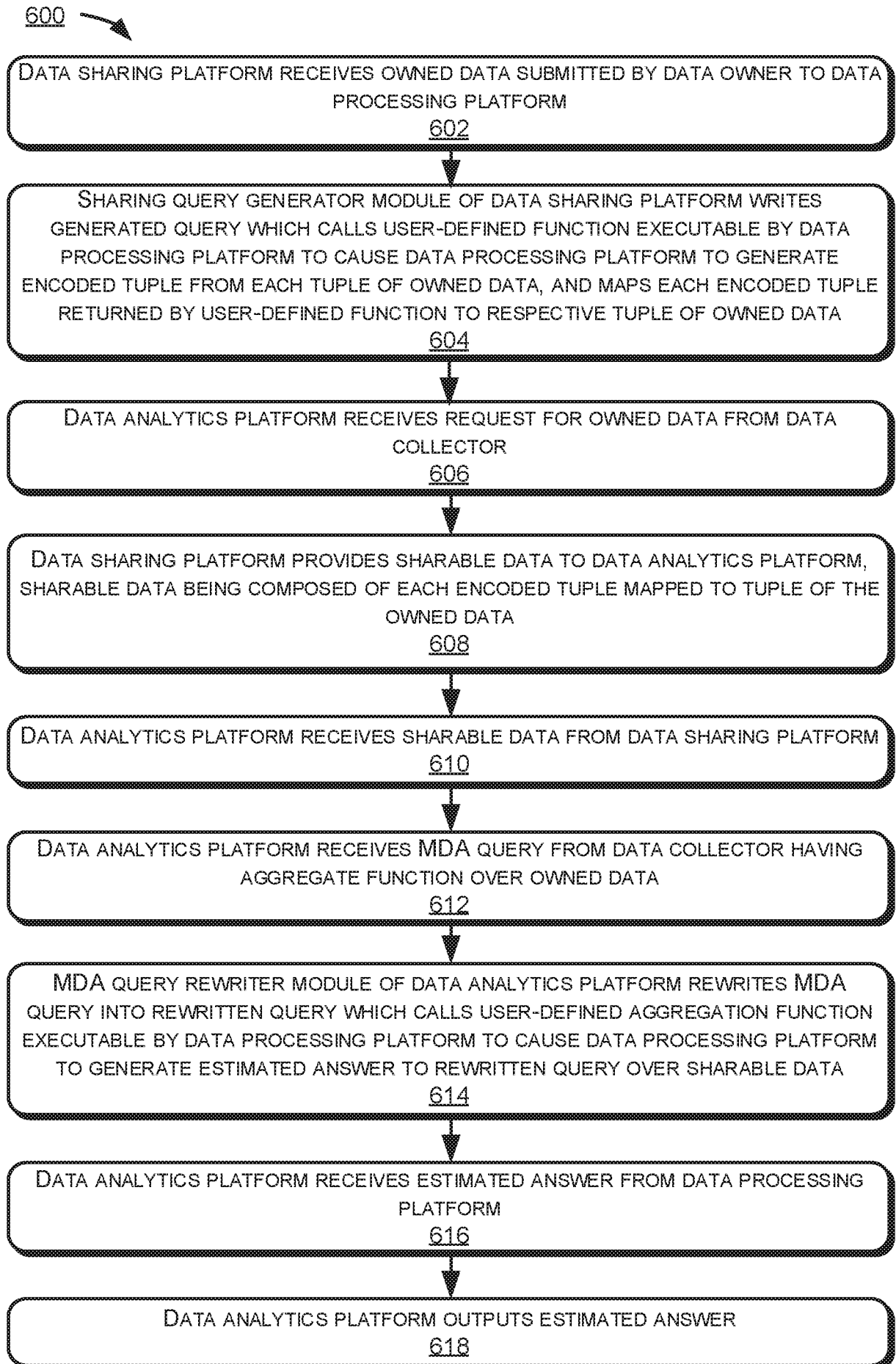


FIG. 6

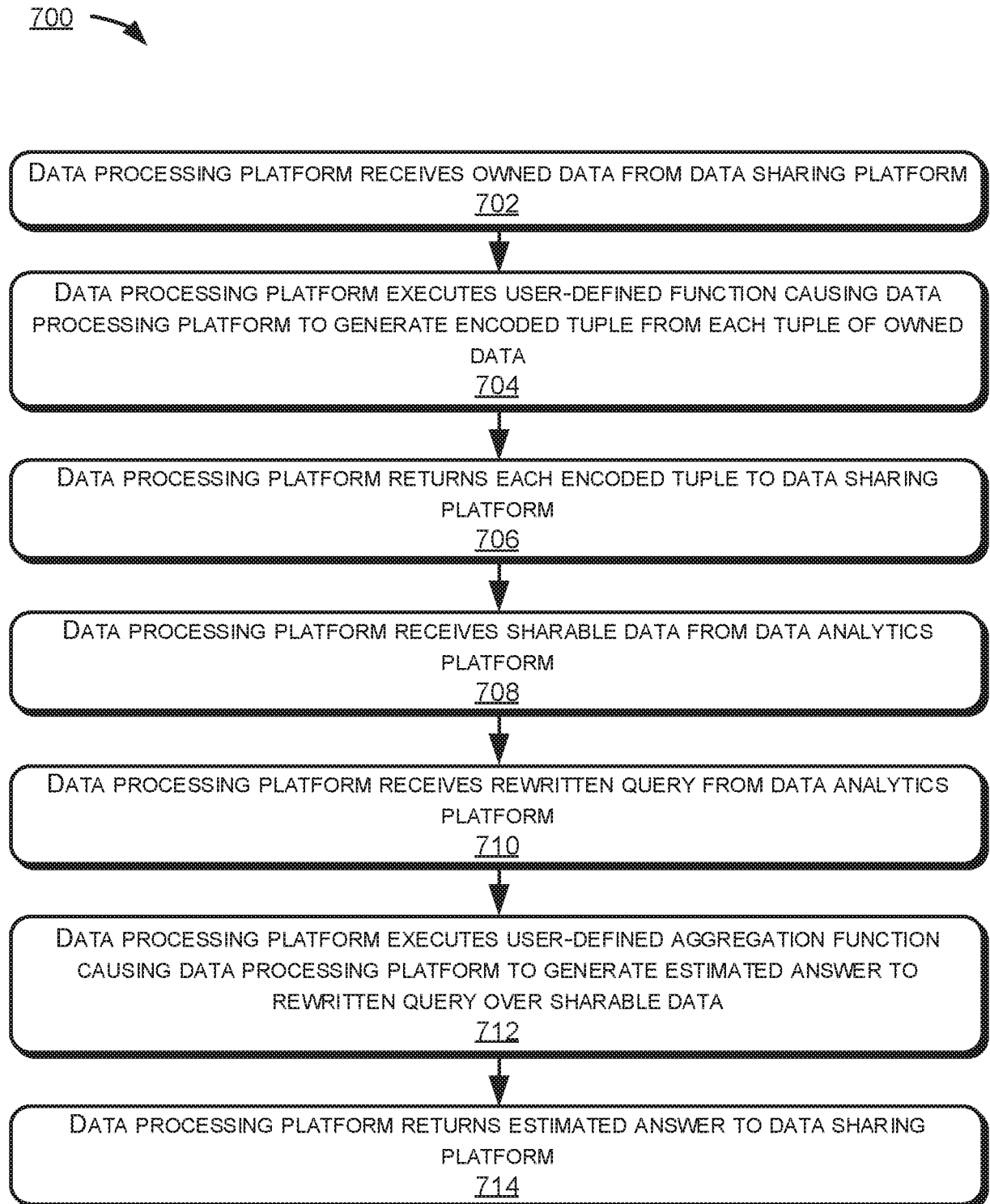


FIG. 7

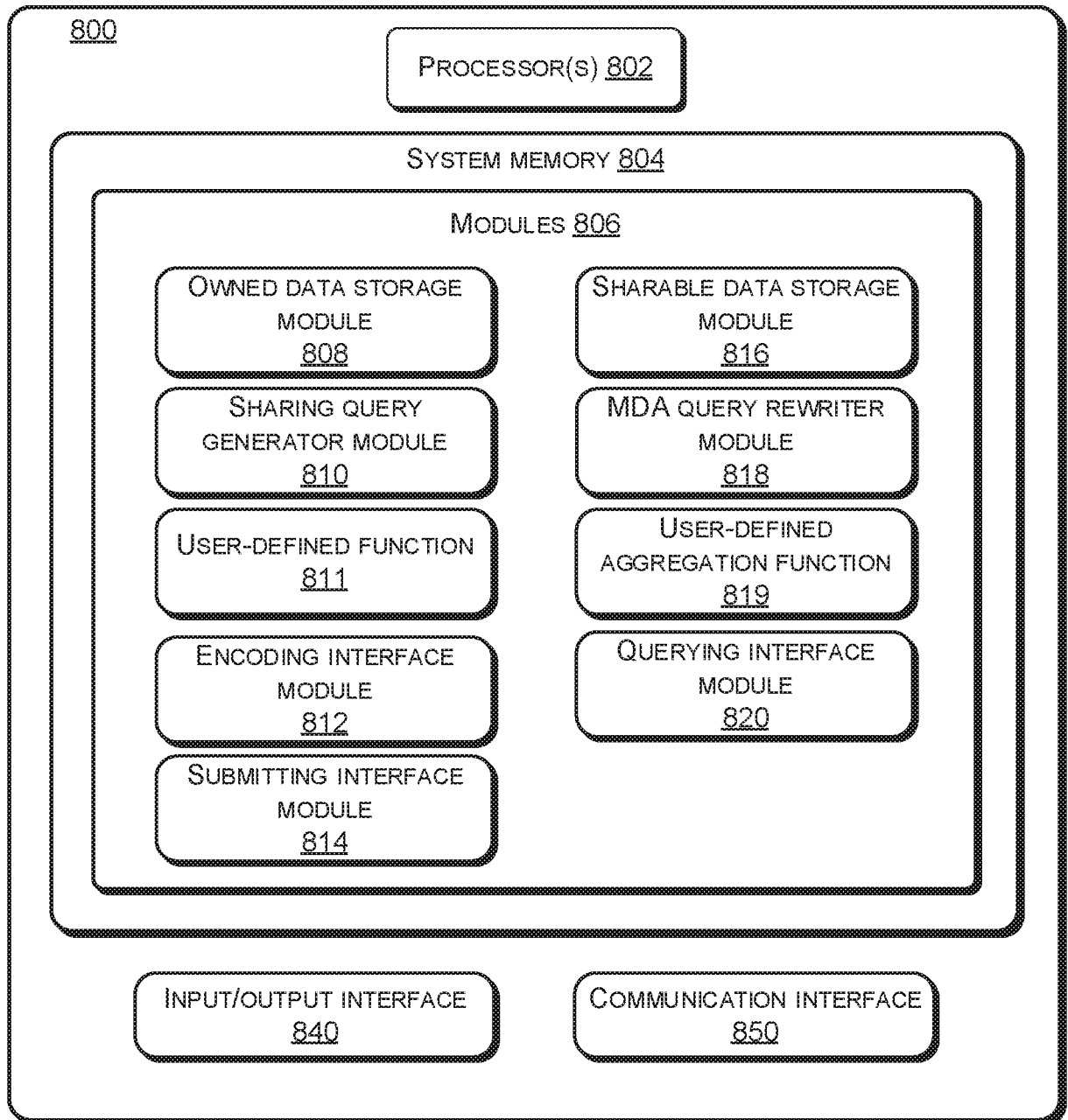


FIG. 8

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2019/090836

A. CLASSIFICATION OF SUBJECT MATTER

G06F 21/62(2013.01)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

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

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

CNPAT, CNKI, WPI, EPODOC: sensitive data, query, upload, owner, differential privacy, LDP, MDA, share, encrypt, code, rewrite, estimation algorithm, decompose, iteration

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 105608388 A (WUHAN UNIVERSITY) 25 May 2016 (2016-05-25) abstract, and description, paragraphs [0063]-[0087]	1-6, 12-14, 20
Y	CN 105608388 A (WUHAN UNIVERSITY) 25 May 2016 (2016-05-25) abstract, and description, paragraphs [0063]-[0087]	7-11, 15-19
Y	WO 2017187207 A1 (PRIVITAR LIMITED) 02 November 2017 (2017-11-02) description, pages 13-14, 20, 30	7-11, 15-19
A	CN 105653981 A (CHINA ELECTRONIC TECHNOLOGY CYBER SECURITY CO., LTD.) 08 June 2016 (2016-06-08) the whole document	1-20
A	CN 107547525 A (FUDAN UNIVERSITY) 05 January 2018 (2018-01-05) the whole document	1-20
A	CN 107113183 A (LITOIU, Marin et al.) 29 August 2017 (2017-08-29) the whole document	1-20



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

26 February 2020

Date of mailing of the international search report

13 March 2020

Name and mailing address of the ISA/CN

National Intellectual Property Administration, PRC
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088
China

Authorized officer

WANG, Xiaoyan

Facsimile No. (86-10)62019451

Telephone No. 86-(10)-53961384

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2019/090836

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	105608388	A	25 May 2016	CN	105608388	B	22 February 2017
WO	2017187207	A1	02 November 2017	EP	3449414	A1	06 March 2019
				CN	109716345	A	03 May 2019
				SG	11201809476	A1	29 November 2018
CN	105653981	A	08 June 2016	CN	105653981	B	30 November 2018
CN	107547525	A	05 January 2018	None			
CN	107113183	A	29 August 2017	WO	2016074094	A1	19 May 2016
				CA	2931041	A1	19 May 2016
				EP	3219051	A1	20 September 2017
				US	2018293283	A1	11 October 2018
				CA	2931041	C	28 March 2017
				EP	3219051	A4	23 May 2018