August 2023

SMART ARRAY

XUEYING SHEN
Visa

ALEXANDER FILATOV
Visa

OKKO GRIPPANDO
Visa

Title: SMART ARRAY

VISA

SHEN, XUEYING

FILATOV, ALEXANDER

GRIPPANDO, OKKO

FIELD OF THE INVENTION

[0001] The present subject matter is generally related to aggregated data metrics and policies for determining whether the data metrics are sharable with an external entity. Specifically, the subject matter relates to aggregated transaction data metrics. Even more specifically, the subject matter relates to policies for determining whether the data metrics are shareable with an external entity based on privacy concerns.

BACKGROUND OF THE INVENTION

[0002] Entities often generate metrics based on collected and aggregated data. These data-derived metrics can provide valuable insight not only to the entity collecting the data but also to other entities (*e.g.*, external entities). For example, a transaction service provider may collect and aggregate transaction data related to transactions it facilitates across a payment network. Further, the transaction service provider may generate transaction data metrics based on the aggregated transaction data. An issuer issuing transactions across the payment network may gain insight from transaction data metrics, for example, because the metrics enable the issuer to evaluate its performance against its peers. Thus, the transaction service provider may wish to share transaction data metrics with external entities, such as issuers who are clients of the transaction service provider.

[0003] However, various privacy concerns often arise when sharing metrics with an external entity. Referring again to the example above, a transaction service provider may generate a metric representing an average transaction amount for a particular type of transaction issued by a set of issuers. The transaction service provider may wish to provide the metric to a particular issuer from the set of issuers. However, depending on the nature of the transaction data from which the metric was derived, the particular issuer, knowing its own transaction data and the metric representative of all of the set of issuers' transaction data, may be able to deduce information related to the individual transaction data of the other issuers. Thus, by sharing the metric, the transaction service

provider may risk exposing private information of the other issuers. Accordingly, entities such as transaction service providers often implement various privacy-related policies to determine whether or not a metric is sharable with an external entity.

[0004] Although the transaction service provider may determine that a metric is not sharable with an external entity (*e.g.*, an issuer) because of privacy concerns, the transaction service provider may still wish to provide some insight related to the metric to the external entity. Various approaches exist for modifying the metric and/or the data underlying the metric so that the metric is externally sharable. However, these approaches are often problematic.

[0005] As one example, the transaction service provider may dilute the underlying transaction data by incorporating additional data. However, the additional data may be only tangentially relevant to the intended metric. Accordingly, a metric derived from diluted data may be based on a comparison of potentially irrelevant transactions and therefore may provide little value to the external entity.

[0006] As another example, the transaction service provider may alter the underlying transaction data such that it complies with its privacy-related policies. However, as a result, both the metric and the underlying data are factually incorrect and therefore may provide little value to the external entity.

[0007] As yet another example, the transaction service provider may generate an array (*e.g.*, a range including a maximum value and a minimum value) that encompasses the metric and also attempts to obfuscate the metric. However, generating an array with too narrow of a range may not adequately obscure the metric and therefore may fail to address the above-mentioned privacy concerns. Conversely, generating an array with too wide of a range is unlikely to provide valuable insight to the external entity. Moreover, regardless of the width of the array, if the array is centered around the metric, it may be easy to deduce both the metric and the underlying

data from the array.

[0008] For example, FIG. 1 is a graph 100 illustrating examples of traditional arrays. The noncompliant value 102 shown in the graph 100 represents a metric that is not externally sharable because the underlying data from which the metric was derived violates a privacy-related policy. The array 104 is an example of an array that is tightly centered around the noncompliant value 102 and does not adequately obscure the non-compliant value. The array 106 is an example of a wide array that obscures the noncompliant value 102 but also fails to provide insight related to the data underlying the non-compliant value.

[0009] In light of the above-mentioned concerns related to sharing metrics externally and the shortcomings of traditional arrays, there is a need for devices, systems, and methods that enable an entity (*e.g.*, a transaction service provider) to provide to an external entity insight related to a metric that would otherwise not be sharable with the external entity.

SUMMARY

[0010] In one aspect, the present disclosure provides a computer-implemented method. The method includes retrieving, by a peer analysis module, a peer set. The peer set includes peer values and weight percentages corresponding to each of the peer values. The method further includes calculating, by the peer analysis module, a true metric for the peer set based on the peer values and the weight percentages. The method further includes determining, by a compliance module, a number of peers represented in the peer set and a highest weight percentage of the weight percentages. The method further includes determining, by the compliance module, that the true metric is not sharable with an external entity based on at least one of the highest weight percentage or the number of peers. The method further includes generating, by an array module, an array representing the true metric. The array is sharable with the external entity. Generating the array includes calculating a weighted standard deviation of the peer set

based on the peer values and the weight percentages. Generating the array further includes generating a first random number and generating a second random number. Generating the array further includes calculating an upper bound of the array based on the true metric, the weighted standard deviation, and the first random number; and calculating a lower bound of the array based on the true metric, the weighted standard deviation, and the second random number.

[0011] In one aspect, the present disclosure provides a smart array system. The smart array system includes a peer analysis module, a compliance module, and an array module. The peer analysis module is configured to retrieve a peer set from a metrics database and calculate a true metric for the peer set. The peer set includes peer values and weight percentages corresponding to each of the peer values. The compliance module is configured to determine that the peer set does not comply with at least one privacy policy rule. The array module is configured to generate an array based on the compliance module determining that the peer set does not comply with the at least one privacy policy rule. The array represents the true metric. Further, the array includes an upper bound and a lower bound. The upper bound is calculated based on the true metric, a weighted standard deviation of the peer set, and a first random number. The lower bound is calculated based on the true metric, the weighted standard deviation of the peer set, and a second random number.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] Embodiments will be described, by way of example only, with reference to the attached figures, wherein:

[0013] **FIG. 1** is a graph illustrating examples of traditional arrays, according to at least one aspect of the present disclosure.

[0014] **FIG. 2** is a graph illustrating examples of smart arrays, according to at least one aspect of the present disclosure.

[0015] **FIG. 3** is diagram illustrating a smart array system, according to at least one aspect of the present disclosure.

[0016] **FIG. 4** illustrates a logic flow diagram of a method for generating a smart array, according to at least one aspect of the present disclosure

[0017] **FIG. 5** illustrates a logic flow diagram of a method for generating a smart array, according to at least one aspect of the present disclosure

[0018] **FIG. 6** illustrates an example array that may be generated by the smart array system of FIG. 3, according to at least one aspect of the present disclosure.

[0019] **FIG. 7** illustrates a logic flow diagram of a method for generating a first random and a second random number used respectively to calculate an upper bound and a lower bound of an array, according to at least one aspect of the disclosure.

[0020] **FIG. 8** is a table illustrating example arrays generated based on the first random number and the second random number generated according to the method of FIG. 7, according to at least one aspect of the present disclosure.

[0021] **FIG. 9** is a table of example edge case rules for generating a first random number and a second random number used respectively to calculate an upper bound and a lower bound of an array, according to at least one aspect of the disclosure.

[0022] **FIG. 10** is a table showing example peer sets and example arrays generated for the peer sets by the smart array system of FIG. 3, according to at least one aspect of the present disclosure.

[0023] **FIG. 11** visually illustrates the example arrays of the FIG. 10, according to at least one aspect of the present disclosure.

[0024] **FIG. 12** is a block diagram of a computer apparatus with data processing

subsystems or components, according to at least one aspect of the present disclosure.

[0025] **FIG. 13** is a diagrammatic representation of an example system that includes a host machine within which a set of instructions to perform any one or more of the methodologies discussed herein may be executed, according to at least one aspect of the present disclosure.

DETAILED DESCRIPTION

[0026] The following disclosure provides exemplary systems, devices, and methods for conducting a financial transaction and related activities, such as the aggregation and analysis of financial transaction data. Although reference may be made to such financial transactions in the examples provided below, aspects are not so limited. That is, the systems, methods, and apparatuses may be utilized for any suitable purpose.

[0027] Before discussing specific embodiments, aspects, or examples, some descriptions of terms used herein are provided below.

[0028] As used herein, the term "comprising" is not intended to be limiting, but may be a transitional term synonymous with "including," "containing," or "characterized by." The term "comprising" may thereby be inclusive or open-ended and does not exclude additional, unrecited elements or method steps when used in a claim. For instance, in describing a method, "comprising" indicates that the claim is open-ended and allows for additional steps. In describing a device, "comprising" may mean that a named element(s) may be essential for an embodiment or aspect, but other elements may be added and still form a construct within the scope of a claim. In contrast, the transitional phrase "consisting of" excludes any element, step, or ingredient not specified in a claim. This is consistent with the use of the term throughout the specification.

[0029] As used herein, the term "computing device" or "computer 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 device, a desktop computer, and/or the like. As an example, a mobile device may include a cellular phone (e.g., a smartphone or standard cellular phone), a portable computer, a wearable device (e.g., watches, glasses, lenses, clothing, and/or the like), a personal digital assistant (PDA), and/or other like devices. The computing device may not be a mobile device, such as a desktop computer. Furthermore, the term "computer" may refer to any computing device that includes the necessary components to send, receive, process, and/or output data, and normally includes a display device, a processor, a memory, an input device, a network interface, and/or the like.

[0030] A "consumer" may include an individual, customer, or a user that may be associated with one or more personal accounts and/or consumer devices. The consumer may also be referred to as a cardholder, account holder, or user.

[0031] Reference to "a device," "a server," "a processor," and/or the like, as used herein, may refer to a previously-recited device, server, or processor that is recited as performing a previous step or function, a different server or processor, and/or a combination of servers and/or processors. For example, as used in the specification and the claims, a first server or a first processor that is recited as performing a first step or a first function may refer to the same or different server or the same or different processor recited as performing a second step or a second function.

[0032] As used herein, the term "external entity" may refer to any entity that is not operating or otherwise controlling the implementation of the smart array system described herein. For example, as described further below, the smart array system 330 (FIG. 3) may be implemented by a computer apparatus 3000 (FIG. 12) and/or a computing system 4000 (FIG. 13) owned by, controlled by, and/or with corresponding resources provisioned to a transaction service provider. Accordingly, an external entity may be an issuer, a merchant, a consumer, and/or a user associated with transactions

facilitated by the transaction service provider.

[0033] The terms "issuer institution," "portable financial device issuer," "issuer," or "issuer bank" may refer to one or more entities that provide one or more accounts (*e.g.*, a credit account, a debit account, a credit card account, a debit card account, and/or the like) to a user (e.g., customer, consumer, and/or the like) for conducting transactions (*e.g.*, payment transactions), such as initiating credit and/or debit payments. For example, an issuer may provide an account identifier, such as a personal account number (PAN), to a user that uniquely identifies one or more accounts associated with the user. The account identifier may be used by the user to conduct a payment transaction. The account identifier may be embodied on a portable financial device, such as a physical financial instrument, e.g., a payment card, and/or may be electronic and used for electronic payments. In some non-limiting aspects of the present disclosure, an issuer may be associated with a bank identification number (BIN) that uniquely identifies the issuer. As used herein "issuer system" or "issuer institution system" may refer to one or more systems operated by or operated on behalf of an issuer. For example, an issuer system may refer to a server executing one or more software applications associated with the issuer. In some non-limiting aspects of the present disclosure, an issuer system may include one or more servers (*e.g.*, one or more authorization servers) for authorizing a payment transaction.

[0034] An "issuer" can include a payment account issuer. The payment account (which may be associated with one or more payment devices) may refer to any suitable payment account (e.g., credit card account, a checking account, a savings account, a merchant account assigned to a consumer, or a prepaid account), an employment account, an identification account, an enrollment account (e.g., a student account), etc.

[0035] As used herein, the term "merchant" may refer to one or more individuals or entities (e.g., operators of retail businesses that provide goods and/or services, and/or access to goods and/or services, to a user (e.g., a customer, a consumer, a customer of

the merchant, and/or the like) based on a transaction (e.g., a payment transaction)). As used herein "merchant system" may refer to one or more computer systems operated by or on behalf of a merchant, such as a server computer executing one or more software applications.

[0036] As used herein, the term "metric" may refer any type of information used to evaluate, analyze, and/or assess data. For example, a metric may be a numerical value representative of aggregated data, such as aggregated transaction data.

[0037] A "payment network" may refer to an electronic payment system used to accept, transmit, or process transactions made by payment devices for money, goods, or services. The payment network may transfer information and funds among issuers, acquirers, merchants, and payment device users. One illustrative non-limiting example of a payment network is VisaNet, which is operated by Visa, Inc.

[0038] A "payment processing network" may refer to a system that receives accumulated transaction information from the gateway processing service, typically at a fixed time each day, and performs a settlement process. Settlement may involve posting the transactions to the accounts associated with the payment devices used for the transactions and calculating the net debit or credit position of each user of the payment devices. An exemplary payment processing network is Interlink®.

[0039] As used herein, the term "peer" may refer to any entity that is being compared to another entity in a data set. For example, a data set may include transaction data corresponding to particular set of issuers. The issuers represented in the data set may be referred to as "peers."

[0040] As used herein, the term "peer set" may refer to any data set that includes data corresponding to a set of "peers."

[0041] As used herein, the term "peer value" may refer to information in a "peer

set" corresponding to a particular "peer."

[0042] As used herein, the term "server" may include one or more computing devices which can be individual, stand-alone machines located at the same or different locations, may be owned or operated by the same or different entities, and may further be one or more clusters of distributed computers or "virtual" machines housed within a datacenter. It should be understood and appreciated by a person of skill in the art that functions performed by one "server" can be spread across multiple disparate computing devices for various reasons. As used herein, a "server" is intended to refer to all such scenarios and should not be construed or limited to one specific configuration. Further, a server as described herein may, but need not, reside at (or be operated by) a merchant, a payment network, a financial institution, a healthcare provider, a social media provider, a government agency, or agents of any of the aforementioned entities. The term "server" may also refer to or include one or more processors or computers, storage devices, or similar computer arrangements that are operated by or facilitate communication and processing for multiple parties in a network environment, such as the Internet, although it will be appreciated that communication may be facilitated over one or more public or private network environments and that various other arrangements are possible. Further, multiple computers, e.g., servers, or other computerized devices, e.g., point-of-sale devices, directly or indirectly communicating in the network environment may constitute a "system," such as a merchant's point-of-sale system. Reference to "a server" or "a processor," as used herein, may refer to a previously-recited server and/or processor that is recited as performing a previous step or function, a different server and/or processor, and/or a combination of servers and/or processors. For example, as used in the specification and the claims, a first server and/or a first processor that is recited as performing a first step or function may refer to the same or different server and/or a processor recited as performing a second step or function.

[0043] A "server computer" may typically be a powerful computer or cluster of computers. For example, the server computer can be a large mainframe, a minicomputer cluster, or a group of servers functioning as a unit. The server computer may be associated with an entity such as a payment processing network, a wallet provider, a merchant, an authentication cloud, an acquirer or an issuer. In one example, the server computer may be a database server coupled to a Web server. The server computer may be coupled to a database and may include any hardware, software, other logic, or combination of the preceding for servicing the requests from one or more client computers. The server computer may comprise one or more computational apparatuses and may use any of a variety of computing structures, arrangements, and compilations for servicing the requests from one or more client computers. In some embodiments or aspects, the server computer may provide and/or support payment network cloud service.

[0044] As used herein, the term "system" may refer to one or more computing devices or combinations of computing devices (e.g., processors, servers, client devices, software applications, components of such, and/or the like).

[0045] A "transaction amount" may be the price assessed to the consumer for the transaction. The transaction amount condition may be a threshold value (e.g., all transactions for an amount exceeding $100) or a range (e.g., all transactions in the range of $25-$50). For example, a user may wish to use a first routing priority list for a transaction for an amount in the range of $0.01-$100 and a second routing priority list for a transaction for an amount exceeding $100.

[0046] The term "transaction data" may include any data associated with one or more transactions. In some embodiments or aspects, the transaction data may merely include an account identifier (e.g., a PAN) or payment token. Alternatively, in other embodiments or aspects, the transaction data may include any information generated, stored, or associated with an issuer, a merchant, a consumer, an account, or any other

related information to a transaction. For example, transaction data may include data in an authorization request message that is generated in response to a payment transaction being initiated by a consumer with a merchant. Alternatively, transaction data may include information associated with one or more transactions that have been previously processed and the transaction information has been stored on a merchant database or other merchant computer. The transaction data may include an account identifier associated with the payment instrument used to initiate the transaction, consumer personal information, products or services purchased, or any other information that may be relevant or suitable for transaction processing. The transaction information may include a payment token or other tokenized or masked account identifier substitute that may be used to complete a transaction and protect the underlying account information of the consumer. Additionally, transaction data may include transaction amounts, transaction volumes, merchant information, issuer information, user information, segment information, geographic information, etc.

[0047] As used herein, the term "transaction service provider" may refer to an entity that receives transaction authorization requests from merchants or other entities and provides guarantees of payment, in some cases through an agreement between the transaction service provider and an issuer. For example, a transaction service provider may include a payment network, such as Visa®, MasterCard®, American Express®, or any other entity that processes transactions. As used herein "transaction service provider system" may refer to one or more systems operated by or operated on behalf of a transaction service provider, such as a transaction service provider system executing one or more software applications associated with the transaction service provider. In some non-limiting embodiments or aspects, a transaction processing system may include one or more server computers with one or more processors and, in some non-limiting embodiments or aspects, may be operated by or on behalf of a transaction service provider.

[0048] A "user" may include an individual. In some embodiments or aspects, a user may be associated with one or more personal accounts and/or mobile devices. The user may also be referred to as a cardholder, account holder, or consumer.

[0049] "User information" may include any information that is associated with a user. For example, the user information may include a device identifier of a device that the user owns or operates and/or account credentials of an account that the user holds. A device identifier may include a unique identifier assigned to a user device that can later be used to verify the user device. In some embodiments or aspects, the device identifier may include a device fingerprint. The device fingerprint may an aggregation of device attributes. The device fingerprint may be generated by a software development kit (SDK) provided on the user device using, for example, a unique identifier assigned by the operating system, an International Mobile Station Equipment Identity (IMEI) number, operating system (OS) version, plug-in version, and the like.

[0050] The present disclosure provides devices, systems, and methods for generating smart arrays. In some aspects, a smart array represents a metric (*e.g.*, a true metric), wherein the true metric is not sharable with the external entity, and wherein the smart array is sharable with the external entity. Thus, the devices, systems, and methods disclosed herein can enable a first entity (*e.g.*, a transaction service provider) to provide data insights to a second entity (*e.g.*, an external entity such as an issuer) that could not otherwise be provided by sharing a true metric.

[0051] For example, a true metric may be calculated for a peer set (*e.g.*, transaction data for a particular set of issuers). The peer set can include peer values (*e.g.*, transaction amounts) and weight percentages corresponding to each of the peer values (*e.g.*, the relative weight and/or contribution of each of the peer values to the larger peer set). Based on the peer values and/or the weight percentages, it may be determined that the true metric is not sharable with the external entity, for example, because by knowing the true metric the external entity may be able to deduce the individual peer

values, thus exposing private data of the peers. The array (*e.g.*, a smart array) can be generated by calculating a weighted standard deviation of the peer set, generating a first random number, and generating a second random number. An upper bound of the array can be generated based on the true metric, the weighted standard deviation, and the first random number. A lower bound of the array can be generated based on the true metric, the sweighted standard deviation, and the second random number.

[0052] According to the example above, the width of the smart array depends of the variance of the peer set, where wider smart arrays are generated for peer sets having a larger variance. It may be easier for an external entity to estimate individual peer values in a peer set with a large variance. Thus, in some aspects, the devices, systems, and methods disclosed herein can generate smart arrays that protect the privacy of individual peers by ensuring that the width of the smart arrays are dependent on the variance of the underlying data. Moreover, as discussed in detail below with respect to FIGS. 3, 7 and 9, various edge case rules may be applied when generating a smart array. In some aspects, the edge case rules are applied when the number of peers in the peer set is especially low (*e.g.* less than 4 peers) and/or when the highest weight percentage of the peer set is especially high (*e.g.*, greater than 70%, greater than 80%, greater than 90%). Applying the edge case rules can ensure that the resulting smart array is sufficiently wide to obfuscate individual peer values in cases where the number of peers in the peers set and/or the highest weight percentage of the peer set make the peer set especially vulnerable to potential privacy concerns.

[0053] Also according to the example above, the upper and lower bounds of the smart array are based on randomly generated numbers. Accordingly, the smart array is not necessary centered around the true metric. Thus, an external entity cannot deduce the true metric from the smart array by simply calculating the midpoint of the array. Thus, in some aspects, the devices, systems, and methods disclosed herein can generate smart arrays that protect the privacy of individual peers by ensuring that the true metric

cannot be identified by the external entity based on calculating the midpoint of the array.

[0054] FIG. 2 is a graph 200 illustrating examples of smart arrays that may be generated by the devices, systems, and methods disclosed herein. The noncompliant value 202 shown in the graph 200 represents a true metric that is not externally sharable because the underlying data from which the true metric was derived violates a privacy-related policy. Each of the smart arrays 204, 206, 208 represent the same noncompliant value 202 but are derived from different peer sets. The width of the smart arrays 204, 206, 208 vary depending on the characteristics of the underlying peer set (*e.g.*, the risk of an external entity deducing individual peer values from the peer set based on weight percentages of the peer set, the number of peers in the peer set, the variance of the peer set, etc.). Generally, as the risk that the external entity may deduce individual peer values increases, the width of the array increases. As also shown by the graph 200, not all of the smart arrays 204, 206, 208 are centered around the non-compliant value 202.

[0055] The devices, system, and methods described herein can provide numerous technical benefits. For example, by calculating (i) an upper bound of the array based on a true metric, a weighted standard deviation, and a first random number and (ii) a lower bound of the array based on the true metric, the weighted standard deviation, and a second random number, the devices, system, and methods disclosed herein can generate an externally sharable smart array representing a true metric that is not externally sharable. Thus, devices, system, and methods described herein can address privacy concerns related to sharing the true metric while also enabling data-related insight to be shared with the external entity.

[0056] Furthermore, the devices, system, and methods described herein integrate the calculation of the (i) an upper bound of the array based on a true metric, a weighted standard deviation, and a first random number and (ii) a lower bound of the array based on the true metric, the weighted standard deviation, and a second random number into

a practical application by generating an externally sharable smart array representing a true metric that is not externally sharable. For example, by using randomly generated numbers to calculate the upper and lower bounds, the smart array is externally sharable because the true metric is adequately obfuscated. Moreover, the first and second random number may be generated in an edge case range for peer sets that include a number of peers (*e.g.* less than 4 peers) and/or a highest weight percentage (*e.g.*, greater than 70%, greater than 80%, greater than 90%) that cause the peer set to be especially vulnerable to potential privacy concerns. Thus, this solution addresses the above-mentioned technical problems associated with metrics that are not sharable because they are insufficiently aggregated and the above-mentioned technical problems associated with traditional arrays that either fail to adequately obfuscate the underlying data or fail to provide useful insight related to the underlying data. Yet further, in some aspects, this calculation method is specifically limited in application to generating smart arrays that are externally sharable and that represent a true metric that is not externally sharable.

[0057] FIG. 3 is diagram 300 illustrating a smart array system 330, according to at least one aspect of the present disclosure. The smart array system 330 can include various modules such as a peer analysis module 332, a compliance module 334, an array module 336, and a policy rules module 338. Although the modules of the smart array system 330 are described below as separately performing various functions, any of the modules can be configured to perform any combination of the functions described herein. Likewise, multiple modules may be combined into a single module to perform any combination of the functions described herein and/or a single module may be split into multiple submodules with each of the submodules performing any of the functions described herein.

[0058] In various aspects, the smart array system 330 and any of the modules included therein may be implemented with a computer apparatus 3000 comprising data

processing subsystems or components shown in FIG. 12 and/or a computing system 4000 comprising a host machine 4002 as shown in FIG. 13. For example, the computer apparatus 3000 and/or the computing system 4000 may include a set of machine instructions that are executable to perform any one or more of the methodologies discussed herein such as, for example, methods 400 (FIG. 4), 500 (FIG. 5), 700 (FIG. 7). In some aspects, a computer apparatus 3000, a computing system 4000, and/or resources thereof implementing the smart array system 330 may be owned by, controlled by, and/or provisioned to a transaction service provider.

[0059] With reference back to FIG. 3, the smart array system 330 is configured to access or otherwise communicate with one or more than one metrics database 310 via a network 320. The network 320 can include any variety of wired, long-range wireless, and/or short-range wireless networks. For example, the network 320 can include an internal network, a Local Area Network (LAN), Wi-Fi, a cellular network, a private network, the Internet, a cloud computing network, and/or a combination of these or other types of networks.

[0060] The metrics database 310 stores various types of transaction data, such as transaction data related to transactions executed across a payment network. The data stored in the metrics database 310 may include transaction data indexed based on transaction amounts, transaction volumes, merchant information, issuer information, user information, segment information, geographic information, etc. Further, the metrics database 310 may include metrics calculated based the indexed transaction data. For example, the metrics database 310 may include an average transaction amount metric for a particular transaction type issued by a particular issuer. As another example, the metrics database 310 may include an average transaction amount metric for a particular issuer and a particular geographic region. As another example, the metrics database 310 may include an average transaction amount metric for a particular issuer and a particular segment.

[0061] The transaction data stored in metrics database 310 may be grouped together based on one or more than one particular peer set. Each peer set includes transaction data for a particular set of peers (*e.g.*, a particular set of issuers, a particular set of merchants, a particular set of users, etc.). Further, the transaction data included in each peer set is correlated with the peers of that set. Thus, a peer set may include peer values (*e.g.*, transaction amounts) for each peer in the peer set and weight percentages corresponding to each of the peer values (*e.g.*, weight percentages calculated based on each peer's volume of transactions contributing to that peer's peer value). For example, a particular peer set may include transaction data grouped based on a particular type of transaction issued by a particular set of issuers. Thus, in this example, the peer set may include each issuer's average transaction amount metric for the particular transaction type and weight percentages corresponding to each issuer's volume of transactions contributing to its average transaction amount metric.

[0062] Still referring to FIG. 3, the peer analysis module 332 is configured to retrieve transaction data from the metrics database 310 via the network 320. For example, the peer analysis module 332 can be configured retrieve a particular peer set from the metrics database 310. As described above, the peer set can include peer values and weight percentages corresponding to each of the peer values.

[0063] Turning now briefly to FIG. 10 described further below together with FIG. 3, FIG. 10 provides a table 1000 with example peer sets 1002A-D that may be retrieved from the metrics database 310 by the peer analysis module 332. For a particular set of peers (*e.g.*, issuers 1004), each peer set 1002A-D includes peer values (*e.g.*, metrics 1006) and weight percentages corresponding to each of the peer values (*e.g.*, peer weights 1008). In these examples, each one of the metrics 1006 may be calculated based on the corresponding issuer's 1004 average transaction amount metric for a particular type transaction that the peer set 1002A-D is based on. Each one of the peer weights 1008 may be calculated based on the volume transactions that the

corresponding issuer 1004 issued compared to the total volume of transactions issued by the entire peer set 1002A-D.

[0064] Referring again to FIG. 3, in some aspects, the peer analysis module 332 is configured to calculate a "true metric" for a peer set retrieved from the metrics database 310. The true metric can be calculated based on the peer values and the weight percentages included in the peer set. For example, the true metric may be a weighted average of the peer values included in the peer set.

[0065] Referring again to the example peer sets 1002A-D in FIG. 10, the true metric for each of the peer sets 1002A-D is the noncompliant value 1010 (explained further below). In these examples, the noncompliant value 1010 for each peer set 1002A-D is calculated as the sum of each individual metric 1006 (peer value) multiplied by its corresponding peer weight 1008 (weight percentage).

[0066] As noted above, a transaction service provider may own and/or control a computer apparatus 3000 (FIG. 12), a computing system 4000 (FIG. 13), and/or resources thereof for implementing the smart array system 330. The transaction service provider may wish to communicate the true metric calculated by the peer analysis module 332 to an external entity (*e.g.*, one or more than one of the peers of the peer set). For example, each of the peers may be an issuer that, based on the true metric, can gain insight related to its performance compared to the larger peer set. However, depending on the number of peers in the peer set and/or the weight percentages of the peer set, a peer with knowledge of its own peer value and the true metric may be able to determine (*e.g.*, back calculate) the peer values of other peers. For example, there may be only a small number of peers (*e.g.*, less than 6 peers) in a peer set and/or one of the peers' transaction volume may make up a significant portion (*e.g.*, more than half) of the total transaction volume of the peer set. By knowing the true metric for this peer set, any one of the peers may be able to estimate the peer values of at least some of the other peers. Thus, a transaction service provider may risk disclosing private

information of the peers by sharing the true metric. Accordingly, the smart array system 330 can include a policy rules module 338 and/or a compliance module 334 configured to protect the privacy of the peers.

[0067] The compliance module 334 is configured to determine whether or not the true metric determined by the peer analysis module 332 is sharable with an external entity (*e.g.*, one or more of the peers of the corresponding peer set), for example, based on privacy risks related to sharing the true metric. The compliance module 334 may determine whether or not the true metric is sharable with an external entity based on characteristics of the peer set from which the true metric was calculated. For example, a peer set may have certain characteristics that allow an external entity with which the corresponding true metric is shared to estimate or otherwise determine individual peer values within the peer set, thus compromising the privacy of the peer(s). The compliance module may apply one or more than one policy rule to determine whether the characteristics of the peer set make the peer set susceptible to this type of privacy risk.

[0068] In some aspects, the one or more than one policy rule applied by the compliance module 334 is based on the number of peers in a peer set. For example, sharing a true metric with a first peer of a peer set that only includes a small number of peers (such as less than 7 peers, less than 6 peers, less than 5 peers, less than 4 peers, etc.) may allow the first peer to estimate peer values associated with other peers. Accordingly, the compliance module 334 can be configured to determine the number of peers represented in a peer set. Further, the compliance module 334 can be configured to determine that a true metric is not sharable with an external entity when the peer set from which the true metric was calculated represents a number of peers that is less than a predetermined threshold, such as less than 7 peers, less than 6, peers, less than 5 peers, less than 4 peers, etc. When a true metric is not sharable with an external entity, the peer set from which the true metric was calculated is sometimes

referred to herein as a "noncompliant peer set." Further, a true metric that is not sharable with an external entity is sometimes referred to herein as a "noncompliant value."

[0069] In some aspects, the one or more than one policy rule applied by the compliance module 334 is based on the weight percentages corresponding to peer values in a peer set. For example, sharing a true metric with a first peer of a peer set that includes a highly weighted peer value (such as weight percentage of greater than 30%, greater than 40%, greater than 50%, greater than 55%, greater than 60%, greater than 65%, greater than 70%, greater than 75%, greater than 80%, or greater than 85%) may allow the first peer to estimate peer values associated with other peers. Accordingly, the compliance module 334 can be configured to determine the highest weight percentage of the weight percentages included in the peer set. Further, the compliance module 334 can be configured to determine that a true metric is not sharable with an external entity when the peer set from which the true metric was calculated has a highest weight percentage greater than a predetermined threshold, such as greater than 30%, greater than 40%, greater than 50%, greater than 55%, greater than 60%, greater than 65%, greater than 70%, greater than 75%, greater than 80%, or greater than 85%.

[0070] In some aspects, the compliance module 334 is configured to receive the one or more than one policy rule from the policy rules module 338. For example, the policy rules module 338 may be configured to store policy rules configured to protect peers' privacy, such as rules based on the number of peers in a peer set and/or a rules based on the highest weight percentage corresponding to a peer value in a peer set, as described above.

[0071] Referring again to the example peer sets 1002A-D shown in FIG. 10 together with FIG. 3, as noted above, the true metric for each of the peer sets 1002A-D is the noncompliant value 1010. For example, the true metric for peer sets 1002A-D

may be determined by the compliance module 334 to be a noncompliant value 1010 because the highest weight percentage (*e.g.*, peer weight 1008) of the corresponding peer set 1002A-D is above a predetermined threshold (*e.g.*, 50%) and/or because the number of peers (*e.g.*, issuers 1004) the corresponding peer set 1002A-D is below a predetermined threshold (*e.g.*, 6 peers).

[0072] Referring again to FIG. 3, as noted above, a transaction service provider implementing the smart array system 330 may wish to communicate the true metric calculated by the peer analysis module 332 to an external entity (*e.g.*, one or more than one of the peers of the peer set) because it may provide various performance-related insights to the external entity. However, as also noted above, the compliance module 334 may determine that the true metric is not sharable with the external entity. Accordingly, the smart array system includes an array module 336 to generate an array that is sharable with the external entity, and can provide similar insights to the external entity, but cannot be used by the external entity to estimate (*e.g.*, back calculate) individual peer values of the corresponding peer set. Thus, the smart array system 330, and particularly the array module 336, can provide the technical advantage of enabling a transaction service provider to provide insights to its customers (*e.g.*, external entities such as issuers) by generating a sharable array representative of an otherwise non-sharable true metric. Further, as described in more detail below, the array module 336 can be configured to generate the array with upper and lower bounds calculated based on randomly generated numbers, thereby making the range and the offset of the array relative to the true metric unpredictable to the external entity with which the array is shared. This unpredictability can make it such that the external entity cannot estimate individual peer values based on the array. Moreover, as discussed further with respect to FIG. 9, when the characteristics of the peer set make it more likely that the external entity could estimate individual peer values, the array module 336 can to generate the random numbers used to calculate the upper and lower bounds of the array to be in an edge case range. The edge case range can be selected to ensure that the resulting upper

and lower bounds are sufficiently wide to obfuscate the true metric. Accordingly, the smart array system 330 and the array module 336 can generate an array that protects the privacy of the corresponding peers while ensuring that the array is representative of the true metric (and therefore can still provide insight the external entity it is shared with).

[0073] Referring to FIG. 3, the array module 336 is configured to generate an array representing the true metric that is also sharable with an external entity. In some aspects, the array module 336 generates the array based on the compliance module 334 determining that the true metric is not sharable with the external entity. To generate the array, the array module 336 is configured to calculate a weighted standard deviation (σ) of the peer set based on the peer values included in the peer set and the weight percentages corresponding to each of the peer values. Further, the array module 336 is configured to generate a first random number (x1) and a second random number (x2). The array module 336 calculates an upper bound of the array based on the weighted standard deviation (σ), the first random number (x1), and the true metric determined by the peer analysis module 332. The array module 336 calculates a lower bound of the array based on the weighted standard deviation (σ), the second random number (x2), and the true metric determined by the peer analysis module 332.

[0074] Turning now to FIG. 6, there is illustrated an example array 600 that may be generated by the array module 336. Referring primarily to FIG. 6 together with FIG. 3, the array module 336 calculates the upper bound 604 by adding the product of the first random number (x1) 610 and the weighted standard deviation (σ) 608 to the true metric 602. The array module 336 calculates the lower bound 606 by subtracting the product of the second random (x2) 620 and the weighted standard deviation (σ) 608 to the true metric 602. As shown in FIG. 6, the array 600 may be generated such that the true metric 602 is not always (*e.g.*, infrequently) located at the midpoint of the array 600. Accordingly, an external entity that the array 600 is shared with is not able to

estimate the true metric 602 by simply calculating the midpoint of the array 600. Yet, the array 600 is still representative of the true metric 602 and can provide insight to the external entity related to its performance relative to a larger peer set.

[0075] Referring again primarily to FIG. 3, the array module 336 can be configured to generate the first random (x1) and the second random number (x2) within specific ranges based on characteristics of the peer set. For example, as the number of peers in the peer set decreases and/or as the highest weight percentage of the peer set increases, the likelihood that an external entity with knowledge of a true metric will be able to estimate individual peer values generally increases. Accordingly, as the number of peers in the peer set decreases and/or as the highest weight percentage of the peer set increases, the array module 336 can be configured to generate the first random (x1) and/or the second random number (x2) within a range of numbers with increasingly higher values.

[0076] For example, the array module 336 can be configured to determine whether or not the peer set satisfies an edge case rule. If the array module 336 determines that the peer set does not satisfy an edge case rule, then the array module 336 may generate the first random number (x1) and the second random number (x2) based on the method 700 shown in FIG. 7. Referring back to FIG. 3, if the array module 336 determines that the peer set does satisfy an edge case rule, then the array module may generate the first random (x1) and the second random number (x2) based on one of the edge case rules illustrated in table 900 of FIG. 9.

[0077] Referring back to FIG. 3 together with FIG. 9, in some aspects, the array module 336 determines whether or not the peer set satisfies an edge case rule based on the number of peers in the peer set and the highest weight percentage of the peer set. For example, the array module 336 may determine that the peer set does not satisfy an edge case rule if the number of peers (*e.g.*, issuers) is greater than or equal to 4 (906 of FIG. 9). As another example, the array module 336 may determine that the peer set

does not satisfy an edge case rule if the highest weight percentage of the peer set (*e.g.*, the dominating issuer's weight) is not greater than 70% (908 of FIG. 9), not greater than 80% (904 of FIG. 9), and/or not greater than 90% (902 of FIG. 9).

[0078] Referring now to FIG. 7, there is illustrated a method 700 for generating the first random number 610 (x1) and the second random number 612 (x2) used respectively to calculate the upper and lower bounds of an array, for example, based on the array module 336 determining that a peer set does not satisfy an edge case rule, according to at least one aspect of the disclosure. Referring primarily to FIG. 7 together with FIGS. 12 and 13, the method 700 may be executed by the array module 336, which may be implemented by the computer apparatus 3000 (FIG. 12) and/or the computing system 4000 (FIG. 13). According to the method 700, the array module 336 executes 702 a first biased coin flip and executes 712 a second biased coil flip. If executing 702 the first bias coin flip results in heads, then the array module 336 generates 704 the first random number 610 (x1) in a first lower range (*e.g.*, 0.5-1). If executing 702 the first bias coin flip results in tails, then the array module 336 generates 706 the first random number 610 (x1) in a first upper range (*e.g.*, 1-1.5). If executing 712 the second bias coin flip results in heads, then the array module 336 generates 714 the second random number 612 (x2) in a second lower range (*e.g.*, 0.5-1). If executing 712 the second bias coin flip results in tails, then the array module 336 generates 716 the second random number 612 (x1) in a second upper range (*e.g.*, 1-1.5).

[0079] FIG. 8 is a table 800 illustrating example arrays 802A-D generated based a first random number (x1) and a second random number (x2) generated according to the method 700. Referring to FIG. 8 together with FIG. 7, the array 802A is an array where executing 702 the first bias coin flip resulted in heads and where executing 712 the second bias coin flip resulted in heads. Accordingly, the first random number 610 (x1) was generated 704 in the first lower range (*e.g.*, 0.5-1) and second random number 612 (x2) was generated 714 in the second lower range (*e.g.*, 0.5-1). Thus, the upper and

lower bounds of the resulting array 802A are more closely centered around and tighter to the true metric compared to the other arrays 802B-D.

[0080] Still referring to FIG. 8 together with FIG. 7, the array 802B is an array where executing 702 the first bias coin flip resulted in tails and where executing 712 the second bias coin flip resulted in heads. Accordingly, the first random number 610 (x1) was generated 706 in the first upper range (*e.g.*, 1-1.5) and second random number 612 (x2) was generated 714 in the second lower range (*e.g.*, 0.5-1). Thus, the upper bound of the resulting array 802B is relatively higher than arrays 800A and 800C and the true metric is skewed towards the lower bound of the array 802B.

[0081] Still referring to FIG. 8 together with FIG. 7, the array 802C is an array where executing 702 the first bias coin flip resulted in heads and where executing 712 the second bias coin flip resulted in tails. Accordingly, the first random number 610 (x1) was generated 704 in the first lower range (*e.g.*, 0.5-1) and second random number 612 (x2) was generated 716 in the second upper range (*e.g.*, 1-1.5). Thus, the lower bound of the resulting array 802C is relatively lower than arrays 800A and 800B and the true metric is skewed towards the upper bound of the array 802C.

[0082] Still referring to FIG. 8 together with FIG. 7, the array 802D is an array where executing 702 the first bias coin flip resulted in tails and where executing 712 the second bias coin flip resulted in tails. Accordingly, the first random number 610 (x1) was generated 716 in the first upper range (*e.g.*, 1-1.5) and second random number 612 (x2) was generated 716 in the second upper range (*e.g.*, 1-1.5). Thus, the upper bound of the resulting array 802D is relatively higher than arrays 800A and 800C and the lower bound of the resulting array 802D is relatively lower than arrays 800A and 800B. Thus, array 802D is the widest of arrays 800A-D.

[0083] As illustrated by FIGS. 7, 8 and 3, by executing the method 700, the array module 336 can generate arrays that are representative of the true metric and are also

not always centered around the true metric and do not have the same relative width. Thus, an external entity receiving multiple arrays is unlikely to recognize a pattern in how the arrays are generated and therefore cannot precisely estimate the true metric based on the array, thereby protecting the privacy of the peers. However, because the arrays are representative of the true metric, the external entity is still able to gain insight related to its performance relative to the peer set.

[0084] FIG. 9 is a table 900 of example edge case rules for generating the first random number (x1) and the second random number (x2) used respectively to calculate the upper and lower bounds of an array. With reference primarily to FIG. 9 together with FIGS. 3 and 7, edge case rules, such as the edge case rules shown in table 900, may be applied in cases where the characteristics of the peer set result an even higher likelihood that an external entity with knowledge of a true metric will be able to estimate individual peer values (compared peer sets that do not satisfy one or more than one of the edge case rules). Accordingly, where an edge case rule is satisfied, instead of generating the first random number (x1) and the second random number (x2) according to the method 700 of FIG. 7, the array module 336 (FIG. 3) can be configured to generate the first random number (x1) and the second random number (x2) to be in an edge case range. Further, to ensure that the true metric is sufficiently obscured, the edge case range may include increasing values as the highest weight percentage of the peer set increases and/or as the number of peers in the peer set decreases. Thus, by applying the edge case rules, the array module 336 can be configured generate arrays that are sharable with an external entity while still protecting the privacy of peers represented in especially small and/or heavily weighted peer sets.

[0085] Still referring primarily to FIG. 9 together with FIGS. 3 and 7, for example, referring to edge case rule 902, if the highest weight percentage of the peer set (*e.g.*, the dominating issuer's weight) is greater than 90%, then the array module 336 (FIG. 3) can be configured to generate the first random number (x1) and the second random

number (x2) in a range of 2-3. Referring to edge case rule 904, if the highest weight percentage of the peer set (*e.g.*, the dominating issuer's weight) is greater than 80%, then the array module 336 can be configured to generate the first random number (x1) and the second random number (x2) in a range of 1.5-2. Referring to edge case rule 906, if the number of peers (*e.g.*, number of issuers) in the peer set is less than 4, then the array module 336 can be configured to generate the first random number (x1) and the second random number (x2) in a range of 1.5-2. Referring to edge case rule 908, if the highest weight percentage of the peer set (*e.g.*, the dominating issuer's weight) is greater than 70%, then the array module 336 can be configured to generate the first random number (x1) and the second random number (x2) in a range of 1-1.5. Referring to edge case rule 910, various other edge case rules can be additional or alternatively applied to control the weighted standard deviation (σ) to ensure that the resulting array is sufficiently wide to obscure the true metric. As shown by FIGS. 7 and 9, the array module 336 can generally be configured to generate the first random number (x1) and the second random number (x2) in higher-valued ranges in edge cases (FIG. 9) compared to non-edge cases (FIG. 7). As a result, the arrays generated by the array module 366 for edge cases may generally be wider than for non-edge cases.

[0086] FIG. 10 is a table 1000 showing example peer sets 1002A-D and example arrays 1012 generated for the peer sets 1002A-D by the smart array system 330 (FIG. 3). FIG. 11 visually illustrates the example arrays 1012 relative to the true metric (noncompliant value 1010). As noted above, each peer set 1002A-D includes peer values 1006 (*e.g.*, metrics 1006) and weight percentages corresponding to each of the peer values (*e.g.*, peer weights 1008). To illustrate how the smart array system 330 can ensure peer privacy by generating arrays 1012 based on characteristics of the peer sets 1002A-D, each of the peer sets 1002A have been selected with a different number of peers (*e.g.* issuers 1004), different peer values 1006, and/or different peer weights 1008. The calculated true metric (*e.g.*, noncompliant value 1010) for each of the peer sets 1002A-D is approximately the same. However, the smart array system 330 can

generate different arrays 1012 based on the varying likelihood that an external entity receiving data related to the peer set 1002A-D would be able to estimate individual peer values 1006 within the peer set 1002A-D.

[0087] As shown in FIGS. 10 and 11, generally, as the inconsistency (*e.g.*, the standard deviation) of the peer sets 1002A-D increases, the width of the generated array 1022 also increases. Similarly, as the highest weight percentage (*e.g.*, peer weight 1008; the largest peer) increases, so does the width of the generated array 1012. For example, although the peer sets 1002A, 1002C have the same number of peers, the individual peer values 1006 in the peer set 1002A are more inconsistent than the peer values 1006 of peer set 1002C. With reference primarily to FIGS. 10 and 11 together with FIG. 3, accordingly, the array 1012 generated by the smart array system 330 (FIG. 3) for the peer set 1002A is wider than the array 1012 for the peer set 1002C. As another example, although the peer sets 1002C, 1002D are have a similar consistency (*e.g.*, similar standard deviation), there are fewer peers (*e.g.*, issuers 1004) in the peer set 1002D. Accordingly, the array 1012 generated by the smart array system 330 for the peer set 1002D is wider than the array 1012 for the peer set 1002C. Thus, generally, the smart array system 330 can generate relatively wider arrays 1012 to further obfuscate the true metric (noncompliant value 1010) and the peer values 1006 for peer sets 1002 with more inconsistency and/or with a fewer number of peers. Further, as shown in FIG. 11, the smart array system 330 can obfuscate the true metric (noncompliant value 1010) by ensuring that arrays 1012 are not necessarily centered around the true metric (noncompliant value 1010).

[0088] Turning now back to FIG. 4, there is illustrated a logic flow diagram of a method 400 for generating a smart array, according to at least one aspect of the present disclosure. With reference primarily to FIG. 4 together with FIGS. 3, 12, and 13, the method 400 may be practiced by the smart array system 330 described above with respect to FIG. 3 and/or any combination of the components of the smart array system

330. As shown in FIG. 3, the smart array system 330 can include a peer analysis module 332, a compliance module 334, and an array module 336. The computer apparatus 3000 (FIG. 12) and/or the computing system 4000 (FIG. 13) may implement the smart array system 330 and any of the modules thereof. Referring now primarily to FIG. 4 together with FIG. 3, according to the method 400, a peer analysis module 332 retrieves 402 a peer set. The peer set can include peer values and weight percentages corresponding to each of the peer values. The peer analysis module 332 may retrieve the peer set from a metrics database 310. Further, according to the method 400, the peer analysis module 332 calculates 404 a true metric for the peer set based on the peer values and the weight percentages.

[0089] Still referring primarily to FIG. 4 together with FIG. 3, according to the method 400, a compliance module 334 (FIG. 3) determines 406 the number of peers represented in the peer set and determines 408 the highest weight percentages of the weight percentages in the peer set. In some aspects, the compliance module 334 determines 410 that the true metric is not sharable with an external entity based on at least one of the highest weight percentage or the number of peers. Further, according to the method 400, the array module 336 (FIG. 3) generates 412 an array representing the true metric. The array is sharable with the external entity. In some aspects, the array module 336 generates 412 the array based on the compliance module 334 determining 410 that the true metric is not sharable with the external entity.

[0090] Referring to FIG. 4, in some aspects, according to the method 400, determining 410 that the true metric is not sharable with the external entity can include determining that the number of peers is less than a first predetermined threshold. The first predetermined threshold may be, for example, 8 peers, 7 peers, 6 peers, 5 peers, 4 peers, 3 peers, or 2 peers. In some aspects, determining 410 that the true metric is not sharable with the external entity can include determining that the highest weight percentage is greater than a second predetermined threshold. The second predetermined threshold may be, for example, 30%, 40%, 50%, 55%, 60%, 65%, 70%, 75%, 80%,

85%, or 90%.

[0091] Still referring primarily to FIG. 4 together with FIG. 3, in some aspects, the method 400 can include eliminating, by the peer analysis module 332 (FIG. 3), at least one of the peer values from the peer set prior to calculating 404 the true metric for the peer set. For example, the peer analysis module may eliminate peer values corresponding to weight percentages greater than 30%, such as greater than 40%, 50%, 55%, 60%, 65%, 70%, 75%, 80%, 85%, or greater than 90%. The peer analysis module 332 may eliminate peers values corresponding to various weight percentages such that the peer set is limited to peers values of peers conducting a similar volume of transactions compared with the external entity.

[0092] Still referring primarily to FIG. 4 together with FIG. 3, in some aspects, the method 400 can include retrieving and/or adding, by the peer analysis module 332 (FIG. 3) at least one additional peer value to the peer set. For example, upon determining 410 that the true metric is not sharable with the external entity, the peer analysis module 332 may identify a peer (*e.g.*, the most similar peer) that is not included in the current peer set and add the peer to the peer set. The peer analysis module 332 may continue adding additional peers until the calculated 404 true metric is sharable with the external entity.

[0093] Turning now to FIG. 5, there is illustrated a logic flow diagram of a method 500 for generating a smart array, according to at least one aspect of the present disclosure. Referring now to FIG. 5 together with FIGS. 3, 4, 12, and 13, in some aspects, the method 500 may be executed as part of the method 400 to generate 412 an array that represents a true metric and is sharable with an external entity, wherein the true metric is based on a peer set including peer values and weight percentages corresponding to each of the peer values. The method 500 may be practiced by the smart array system 330, and more particularly the array module 336, described above with respect to FIG. 3. The computer apparatus 3000 (FIG. 12) and/or the computing

system 4000 (FIG. 13) may implement the smart array system 330 and/or the array module 336. Referring now primarily to FIG. 5 together with FIG. 3, according to the method 500, the array module 336 calculates 502 a weighted standard deviation of the peer set based on the peer values and the weight percentages. Further, the array module 336 generates 504 a first random number and generates 506 a second random number. Yet further, the array module 336 calculates 508 an upper bound of the array based on the true metric, the weighted standard deviation, and the first random number. The array module 336 calculates 510 a lower bound of the array based on the true metric, the weighted standard deviation, and the second random number.

[0094] Referring to FIG. 5, in some aspects, according to the method 500, calculating 508 the upper bound of the array includes adding a first product of the weighted standard deviation and the first random number to the true metric. In some aspects, calculating 510 the lower bound of the array includes subtracting a second product of the weighted standard deviation and the second random number from the true metric.

[0095] Referring primarily to FIG. 5 together with FIG. 3, in some aspects, according to the method 500, the array module 336 (FIG. 3) determines the peer set does not satisfy an edge case rule. In this aspect, the array module 336 can execute a first bias coin flip and a second bias coin flip. Further, in this aspect, generating 504 the first random number can include generating the first random number in a first lower range if the first bias coin flip results in heads and generating the first random number in a first upper range if the first bias coil flip results in tails. Yet further, in this aspect, generating 506 the second random number can include generating the second random number in a second lower range if the second bias coin flip results in heads and generating the second random number in a second upper range if the second bias coil flip results in tails. In some aspects, the first lower range is 0.5 to 1.0, the first upper range is 1.0 to 1.5, the second lower range is 0.5 to 1.0, and the second upper range is 1.0 to 1.5.

[0096] Still referring primarily to FIG. 5 together with FIG. 3, in some aspects, according to the method 500, the array module 336 (FIG. 3) determines the peer set does satisfy an edge case rule. In this aspect, generating 504 the first random number and generating 506 the second random number can include generating the first and second random number to be in an edge case range. In some aspects, determining the peer set satisfies an edge case rule includes determining the highest weight percentage is greater than 70% and generating the first and second random numbers in an edge case range of 1.0 to 1.5. In some aspects, determining the peer set satisfies an edge case rule includes determining the highest weight percentage is greater than 80% and generating the first and second random numbers in an edge case range of 1.5 to 2.0. In some aspects, determining the peer set satisfies an edge case rule includes determining the highest weight percentage is greater than 90% and generating the first and second random numbers in an edge case range of 2.0 to 3.0. In some aspects, determining the peer set satisfies an edge case rule includes determining the number of peers is less than 4 and generating the first and second random numbers in an edge case range of 1.5 to 2.0.

[0097] Still referring primarily to FIG. 5 together with FIG. 3, in some aspects, according to the method 500, a user specifies at least one of an absolute upper bound or an absolute lower bound. Further, the array module 336 (FIG. 3) modifies the array based on at least one of the absolute upper bound or the absolute lower bound. For example, for ticket size metric, the array should not include negative values. Accordingly, a user can set an absolute lower bound of 0. If an array is generated that includes negative values, the array module 366 can be configured to calculate the difference of the calculated 510 lower bound and the absolute lower bound (*e.g.*, 0) and add the difference to the calculated 508 upper bound. Similarly, if an array is generated that exceeds the absolute upper bound, the array module 366 can be configured to calculate the difference of the calculated 508 upper bound and the absolute upper bound and subtract the difference from the calculated 510 lower bound. If both the absolute

upper bound and the absolute lower bound are exceeded by the array, then the array module 336 can be configured to generate an array with an upper bound equal to the absolute upper bound and a lower bound equal to the absolute lower bound.

[0098] Still referring primarily to FIG. 5 together with FIGS. 3 and 4, in In some aspects, according to the method 400 (FIG. 4) and/or 500, if a true metric is calculated 404 (FIG. 4) based on two or more metrics (*e.g.*, a true metric calculated based on two or more types of peer values, wherein each type of peer values correspond to a different set of weight percentages) then the smart array system 330 (FIG. 3) can be configured to select the set of weight percentages that includes the highest overall weight percentage. For example, a true metric for average ticket may be calculated based on total amount (a first type of peer value) divided by total transaction (a second type of peer value). The peer values for total amount correspond to one set of weight percentages and the peer values for total transaction correspond to a second set of weight percentages. The maximum weight percentage of the weight percentage set corresponding to total amount is 90% and the maximum weight percentage of the weight percentage set corresponding to total transaction 80%. The smart array system 330 can be configured to select the weight percentage set corresponding to total amount as the weight used to calculate 502 the weighted standard deviation.

[0099] FIG. 12 is a block diagram of a computer apparatus 3000 comprising data processing subsystems or components, according to at least one aspect of the present disclosure. The subsystems shown in FIG. 12 are interconnected via a system bus 3010. Additional subsystems such as a printer 3018, keyboard 3026, fixed disk 3028 (or other memory comprising computer readable media), monitor 3022, which is coupled to a display adapter 3020, and others are shown. Peripherals and input/output (I/O) devices, which couple to an I/O controller 3012 (which can be a processor or other suitable controller), can be connected to the computer system by any number of means known in the art, such as a serial port 3024. For example, the serial port 3024 or external

interface 3030 can be used to connect the computer apparatus to a wide area network such as the Internet, a mouse input device, or a scanner. The interconnection via system bus allows the central processor 3016 to communicate with each subsystem and to control the execution of instructions from system memory 3014 or the fixed disk 3028, as well as the exchange of information between subsystems. The system memory 3014 and/or the fixed disk 3028 may embody a computer readable medium.

[0100] FIG. 13 is a diagrammatic representation of an example computing system 4000 that includes a host machine 4002 within which a set of instructions to perform any one or more of the methodologies discussed herein may be executed, according to at least one aspect of the present disclosure. In various aspects, the host machine 4002 operates as a standalone device or may be connected (e.g., networked) to other machines. In a networked deployment, the host machine 4002 may operate in the capacity of a server or a client machine in a server-client network environment, or as a peer machine in a peer-to-peer (or distributed) network environment. The host machine 4002 may be a computer or computing device, a personal computer (PC), a tablet PC, a set-top box (STB), a personal digital assistant (PDA), a cellular telephone, a portable music player (e.g., a portable hard drive audio device such as an Moving Picture Experts Group Audio Layer 3 (MP3) player), a web appliance, a network router, switch or bridge, or any machine capable of executing a set of instructions (sequential or otherwise) that specify actions to be taken by that machine. Further, while only a single machine is illustrated, the term “machine” shall also be taken to include any collection of machines that individually or jointly execute a set (or multiple sets) of instructions to perform any one or more of the methodologies discussed herein.

[0101] The example system 4000 includes the host machine 4002, running a host operating system (OS) 4004 on a processor or multiple processor(s)/processor core(s) 4006 (e.g., a central processing unit (CPU), a graphics processing unit (GPU), or both), and various memory nodes 4008. The host OS 4004 may include a hypervisor 4010 which is able to control the functions and/or communicate with a virtual machine

("VM") 4012 running on machine readable media. The VM 4012 also may include a virtual CPU or vCPU 4014. The memory nodes 4008 may be linked or pinned to virtual memory nodes or vNodes 4016. When the memory node 4008 is linked or pinned to a corresponding vNode 4016, then data may be mapped directly from the memory nodes 4008 to the corresponding vNode 4016.

[0102] All the various components shown in host machine 4002 may be connected with and to each other, or communicate to each other via a bus (not shown) or via other coupling or communication channels or mechanisms. The host machine 4002 may further include a video display, audio device or other peripherals 4018 (e.g., a liquid crystal display (LCD), alpha-numeric input device(s) including, e.g., a keyboard, a cursor control device, e.g., a mouse, a voice recognition or biometric verification unit, an external drive, a signal generation device, e.g., a speaker,) a persistent storage device 4020 (also referred to as disk drive unit), and a network interface device 4022. The host machine 4002 may further include a data encryption module (not shown) to encrypt data. The components provided in the host machine 4002 are those typically found in computer systems that may be suitable for use with aspects of the present disclosure and are intended to represent a broad category of such computer components that are known in the art. Thus, the system 4000 can be a server, minicomputer, mainframe computer, or any other computer system. The computer may also include different bus configurations, networked platforms, multi-processor platforms, and the like. Various operating systems may be used including UNIX, LINUX, WINDOWS, QNX ANDROID, IOS, CHROME, TIZEN, and other suitable operating systems.

[0103] The disk drive unit 4024 also may be a Solid-state Drive (SSD), a hard disk drive (HDD) or other includes a computer or machine-readable medium on which is stored one or more sets of instructions and data structures (e.g., data/instructions 4026) embodying or utilizing any one or more of the methodologies or functions described herein. The data/instructions 4026 also may reside, completely or at least partially, within the main memory node 4008 and/or within the processor(s) 4006 during

execution thereof by the host machine 4002. The data/instructions 4026 may further be transmitted or received over a network 4028 via the network interface device 4022 utilizing any one of several well-known transfer protocols (e.g., Hyper Text Transfer Protocol (HTTP)).

[0104] The processor(s) 4006 and memory nodes 4008 also may comprise machine-readable media. The term "computer-readable medium" or “machine-readable medium” should be taken to include a single medium or multiple medium (e.g., a centralized or distributed database and/or associated caches and servers) that store the one or more sets of instructions. The term "computer-readable medium" shall also be taken to include any medium that is capable of storing, encoding, or carrying a set of instructions for execution by the host machine 4002 and that causes the host machine 4002 to perform any one or more of the methodologies of the present application, or that is capable of storing, encoding, or carrying data structures utilized by or associated with such a set of instructions. The term ”computer-readable medium” shall accordingly be taken to include, but not be limited to, solid-state memories, optical and magnetic media, and carrier wave signals. Such media may also include, without limitation, hard disks, floppy disks, flash memory cards, digital video disks, random access memory (RAM), read only memory (ROM), and the like. The example aspects described herein may be implemented in an operating environment comprising software installed on a computer, in hardware, or in a combination of software and hardware.

[0105] One skilled in the art will recognize that Internet service may be configured to provide Internet access to one or more computing devices that are coupled to the Internet service, and that the computing devices may include one or more processors, buses, memory devices, display devices, input/output devices, and the like. Furthermore, those skilled in the art may appreciate that the Internet service may be coupled to one or more databases, repositories, servers, and the like, which may be utilized to implement any of the various aspects of the disclosure as described herein.

[0106] The computer program instructions also may be loaded onto a computer, a server, other programmable data processing apparatus, or other devices to cause a series of operational steps to be performed on the computer, other programmable apparatus or other devices to produce a computer implemented process such that the instructions which execute on the computer or other programmable apparatus provide processes for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks.

[0107] Suitable networks may include or interface with any one or more of, for instance, a local intranet, a PAN (Personal Area Network), a LAN (Local Area Network), a WAN (Wide Area Network), a MAN (Metropolitan Area Network), a virtual private network (VPN), a storage area network (SAN), a frame relay connection, an Advanced Intelligent Network (AIN) connection, a synchronous optical network (SONET) connection, a digital T1, T3, E1 or E3 line, Digital Data Service (DDS) connection, DSL (Digital Subscriber Line) connection, an Ethernet connection, an ISDN (Integrated Services Digital Network) line, a dial-up port such as a V.90, V.34 or V.34bis analog modem connection, a cable modem, an ATM (Asynchronous Transfer Mode) connection, or an FDDI (Fiber Distributed Data Interface) or CDDI (Copper Distributed Data Interface) connection. Furthermore, communications may also include links to any of a variety of wireless networks, including WAP (Wireless Application Protocol), GPRS (General Packet Radio Service), GSM (Global System for Mobile Communication), CDMA (Code Division Multiple Access) or TDMA (Time Division Multiple Access), cellular phone networks, GPS (Global Positioning System), CDPD (cellular digital packet data), RIM (Research in Motion, Limited) duplex paging network, Bluetooth radio, or an IEEE 802.11-based radio frequency network. The network 4028 can further include or interface with any one or more of an RS-232 serial connection, an IEEE-1394 (Firewire) connection, a Fiber Channel connection, an IrDA (infrared) port, a SCSI (Small Computer Systems Interface) connection, a USB (Universal Serial Bus) connection or other wired or wireless, digital

or analog interface or connection, mesh or Digi® networking.

[0108] In general, a cloud-based computing environment is a resource that typically combines the computational power of a large grouping of processors (such as within web servers) and/or that combines the storage capacity of a large grouping of computer memories or storage devices. Systems that provide cloud-based resources may be utilized exclusively by their owners or such systems may be accessible to outside users who deploy applications within the computing infrastructure to obtain the benefit of large computational or storage resources.

[0109] The cloud is formed, for example, by a network of web servers that comprise a plurality of computing devices, such as the host machine 4002, with each server 4030 (or at least a plurality thereof) providing processor and/or storage resources. These servers manage workloads provided by multiple users (e.g., cloud resource customers or other users). Typically, each user places workload demands upon the cloud that vary in real-time, sometimes dramatically. The nature and extent of these variations typically depends on the type of business associated with the user.

[0110] It is noteworthy that any hardware platform suitable for performing the processing described herein is suitable for use with the technology. The terms "computer-readable storage medium" and "computer-readable storage media" as used herein refer to any medium or media that participate in providing instructions to a CPU for execution. Such media can take many forms, including, but not limited to, non-volatile media, volatile media, and transmission media. Non-volatile media include, for example, optical or magnetic disks, such as a fixed disk. Volatile media include dynamic memory, such as system RAM. Transmission media include coaxial cables, copper wire and fiber optics, among others, including the wires that comprise one aspect of a bus. Transmission media can also take the form of acoustic or light waves, such as those generated during radio frequency (RF) and infrared (IR) data communications. Common forms of computer-readable media include, for example, a flexible disk, a hard disk, magnetic tape, any other magnetic medium, a CD-ROM disk,

digital video disk (DVD), any other optical medium, any other physical medium with patterns of marks or holes, a RAM, a PROM, an EPROM, an EEPROM, a FLASH EPROM, any other memory chip or data exchange adapter, a carrier wave, or any other medium from which a computer can read.

[0111] Various forms of computer-readable media may be involved in carrying one or more sequences of one or more instructions to a CPU for execution. A bus carries the data to system RAM, from which a CPU retrieves and executes the instructions. The instructions received by system RAM can optionally be stored on a fixed disk either before or after execution by a CPU.

[0112] Computer program code for carrying out operations for aspects of the present technology may be written in any combination of one or more programming languages, including an object oriented programming language such as Java, Smalltalk, C++, or the like and conventional procedural programming languages, such as the "C" programming language, Go, Python, or other programming languages, including assembly languages. The program code may execute entirely on the user's computer, partly on the user's computer, as a stand-alone software package, partly on the user's computer and partly on a remote computer or entirely on the remote computer or server. In the latter scenario, the remote computer may be connected to the user's computer through any type of network, including a local area network (LAN) or a wide area network (WAN), or the connection may be made to an external computer (for example, through the Internet using an Internet Service Provider).

[0113] Examples of the system or method according to various aspects of the present disclosure are provided below and may include any one or more than one, and any combination thereof.

[0114] Clause 1. A computer-implemented method, comprising: retrieving, by a peer analysis module, a peer set comprising peer values and weight percentages corresponding to each of the peer values; calculating, by the peer analysis module, a

true metric for the peer set based on the peer values and the weight percentages; determining, by a compliance module, a number of peers represented in the peer set; determining, by the compliance module, a highest weight percentage of the weight percentages; determining, by the compliance module, that the true metric is not sharable with an external entity based on at least one of the highest weight percentage or the number of peers; and generating, by an array module, an array representing the true metric, wherein the array is sharable with the external entity, and wherein the generating the array comprises: calculating a weighted standard deviation of the peer set based on the peer values and the weight percentages; generating a first random number; generating a second random number; calculating an upper bound of the array based on the true metric, the weighted standard deviation, and the first random number; and calculating a lower bound of the array based on the true metric, the weighted standard deviation, and the second random number.

[0115] Clause 2. The method of Clause 1, wherein the calculating the upper bound of the array comprises adding a first product of the weighted standard deviation and the first random number to the true metric, and wherein the calculating the lower bound of the array comprises subtracting a second product of the weighted standard deviation and the second random number from the true metric.

[0116] Clause 3. The method of any of Clauses 1-2, wherein the generating the array further comprises: determining the peer set does not satisfy an edge case rule; executing a first biased coin flip; generating the first random number in a first lower range if the first bias coin flip results in heads; generating the first random number in a first upper range if the first bias coin flip results in tails; executing a second biased coin flip; generating the second random number in a second lower range if the second bias coin flip results in heads; and generating the second random number in a second upper range the second bias coin flip results in tails.

[0117] Clause 4. The method of any of Clauses 1-3, wherein the first lower range

is 0.5 to 1.0, wherein the first upper range is 1.0 to 1.5, wherein the second lower range is 0.5 to 1.0, and wherein the second upper range is 1.0 to 1.5.

[0118] Clause 5. The method of any of Clauses 1-4, wherein the generating the array further comprises: determining the peer set satisfies an edge case rule; and generating the first random number and the second random number to be in an edge case range.

[0119] Clause 6. The method of any of Clauses 1-5, wherein the determining the peer set satisfies an edge case rule comprises determining the highest weight percentage is greater than 70%, and wherein the edge case range is 1.0 to 1.5.

[0120] Clause 7. The method of any of Clauses 1-5, wherein the determining the peer set satisfies an edge case rule comprises determining the highest weight percentage is greater than 80%, and wherein the edge case range is 1.5 to 2.0.

[0121] Clause 8. The method of any of Clauses 1-5, wherein the determining the peer set satisfies an edge case rule comprises determining the highest weight percentage is greater than 90%, and wherein the edge case range is 2.0 to 3.0.

[0122] Clause 9. The method of any of Clauses 1-5, wherein the determining the peer set satisfies an edge case rule comprises determining the number of peers is less than 4, and wherein the edge case range is 1.5 to 2.0.

[0123] Clause 10. The method of any of Clauses 1-9, further comprising: specifying, by a user, at least one of an absolute upper bound or an absolute lower bound; and modifying, by the array module, the array based on at least one of the absolute upper bound or the absolute lower bound.

[0124] Clause 11. The method of any of Clauses 1-10, wherein the determining that the true metric is not sharable with the external entity comprises at least one of:

determining the number of peers is less than a first predetermined threshold; or determining the highest weight percentage is greater than a second predetermined threshold.

[0125] Clause 12. The method of any of Clauses 1-11, wherein the first predetermined threshold is 5 peers, and wherein the second predetermined threshold is 50%.

[0126] Clause 13. A smart array system, comprising: a peer analysis module configured to retrieve a peer set from a metrics database and calculate a true metric for the peer set, wherein the peer set comprises peer values and weight percentages corresponding to each of the peer values; a compliance module configured to determine that the peer set does not comply with at least one privacy policy rule; and an array module configured to generate an array based on the compliance module determining that the peer set does not comply with the at least one privacy policy rule, wherein the array represents the true metric, and wherein the array comprises: an upper bound calculated based on the true metric, a weighted standard deviation of the peer set, and a first random number; and a lower bound calculated based on the true metric, the weighted standard deviation of the peer set, and a second random number.

[0127] Clause 14. The smart array system of Clause 13, wherein the array module is further configured to: calculate the upper bound of the array by adding a first product of the weighted standard deviation and the first random number to the true metric, and calculate the lower bound of the array by subtracting a second product of the weighted standard deviation and the second random number from the true metric.

[0128] Clause 15. The smart array system of any of Clauses 13-14, wherein the array module is further configured: execute a first biased coin flip; generate the first random number in a first lower range if the first bias coin flip results in heads; generate the first random number in a first upper range if the first bias coin flip results in tails;

execute a second biased coin flip; generate the second random number in a second lower range if the second bias coin flip results in heads; and generate the second random number in a second upper range the second bias coin flip results in tails.

[0129] Clause 16. The smart array system of any of Clauses 13-15, wherein the first lower range is 0.5 to 1.0, wherein the first upper range is 1.0 to 1.5, wherein the second lower range is 0.5 to 1.0, and wherein the second upper range is 1.0 to 1.5.

[0130] Clause 17. The smart array system of any of Clauses 13-14, wherein the array module is further configured to: determine the peer set satisfies an edge case rule; and generate the first random number and the second random number to be in an edge case range based on determining the peer set satisfies the edge case rule.

[0131] Clause 18. The smart array system of any of Clauses 13-17, wherein the array module is further configured to: modify at least one of the upper bound of the array or the lower bound of the array based on a user input.

[0132] Clause 19. The smart array system of any of Clauses 13-18, further comprising: a policy rules module configured to store the at least one privacy policy rule, wherein the compliance module is configured retrieve the at least one privacy policy rule from the policy rules module.

[0133] Clause 20. The smart array system of any of Clauses 13-19, wherein the at least one privacy policy rule is based on at least one of a number of peers in the peer set or a highest weight percentage of the weight percentages.

[0134] The foregoing detailed description has set forth various forms of the systems and/or processes via the use of block diagrams, flowcharts, and/or examples. Insofar as such block diagrams, flowcharts, and/or examples contain one or more functions and/or operations, it will be understood by those within the art that each function and/or operation within such block diagrams, flowcharts, and/or examples can

be implemented, individually and/or collectively, by a wide range of hardware, software, firmware, or virtually any combination thereof. Those skilled in the art will recognize that some aspects of the forms disclosed herein, in whole or in part, can be equivalently implemented in integrated circuits, as one or more computer programs running on one or more computers (e.g., as one or more programs running on one or more computer systems), as one or more programs running on one or more processors (e.g., as one or more programs running on one or more microprocessors), as firmware, or as virtually any combination thereof, and that designing the circuitry and/or writing the code for the software and or firmware would be well within the skill of one of skill in the art in light of this disclosure. In addition, those skilled in the art will appreciate that the mechanisms of the subject matter described herein are capable of being distributed as one or more program products in a variety of forms, and that an illustrative form of the subject matter described herein applies regardless of the particular type of signal bearing medium used to actually carry out the distribution.

[0135] Instructions used to program logic to perform various disclosed aspects can be stored within a memory in the system, such as dynamic random access memory (DRAM), cache, flash memory, or other storage. Furthermore, the instructions can be distributed via a network or by way of other computer readable media. Thus a machine-readable medium may include any mechanism for storing or transmitting information in a form readable by a machine (e.g., a computer), but is not limited to, floppy diskettes, optical disks, compact disc, read-only memory (CD-ROMs), and magneto-optical disks, read-only memory (ROMs), random access memory (RAM), erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), magnetic or optical cards, flash memory, or a tangible, machine-readable storage used in the transmission of information over the Internet via electrical, optical, acoustical or other forms of propagated signals (e.g., carrier waves, infrared signals, digital signals, etc.). Accordingly, the non-transitory computer-readable medium includes any type of tangible machine-readable medium suitable for

storing or transmitting electronic instructions or information in a form readable by a machine (e.g., a computer).

[0136] Any of the software components or functions described in this application, may be implemented as software code to be executed by a processor using any suitable computer language such as, for example, Python, Java, C++ or Perl using, for example, conventional or object-oriented techniques. The software code may be stored as a series of instructions, or commands on a computer readable medium, such as RAM, ROM, a magnetic medium such as a hard-drive or a floppy disk, or an optical medium such as a CD-ROM. Any such computer readable medium may reside on or within a single computational apparatus, and may be present on or within different computational apparatuses within a system or network.

[0137] As used in any aspect herein, the term "logic" may refer to an app, software, firmware and/or circuitry configured to perform any of the aforementioned operations. Software may be embodied as a software package, code, instructions, instruction sets and/or data recorded on non-transitory computer readable storage medium. Firmware may be embodied as code, instructions or instruction sets and/or data that are hard-coded (e.g., nonvolatile) in memory devices.

[0138] As used in any aspect herein, the terms "component," "system," "module" and the like can refer to a computer-related entity, either hardware, a combination of hardware and software, software, or software in execution.

[0139] As used in any aspect herein, an "algorithm" refers to a self-consistent sequence of steps leading to a desired result, where a "step" refers to a manipulation of physical quantities and/or logic states which may, though need not necessarily, take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It is common usage to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like. These and

similar terms may be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities and/or states.

[0140] A network may include a packet switched network. The communication devices may be capable of communicating with each other using a selected packet switched network communications protocol. One example communications protocol may include an Ethernet communications protocol which may be capable of permitting communication using a Transmission Control Protocol/Internet Protocol (TCP/IP). The Ethernet protocol may comply or be compatible with the Ethernet standard published by the Institute of Electrical and Electronics Engineers (IEEE) titled "IEEE 802.3 Standard", published in December, 2008 and/or later versions of this standard. Alternatively or additionally, the communication devices may be capable of communicating with each other using an X.25 communications protocol. The X.25 communications protocol may comply or be compatible with a standard promulgated by the International Telecommunication Union-Telecommunication Standardization Sector (ITU-T). Alternatively or additionally, the communication devices may be capable of communicating with each other using a frame relay communications protocol. The frame relay communications protocol may comply or be compatible with a standard promulgated by Consultative Committee for International Telegraph and Telephone (CCITT) and/or the American National Standards Institute (ANSI). Alternatively or additionally, the transceivers may be capable of communicating with each other using an Asynchronous Transfer Mode (ATM) communications protocol. The ATM communications protocol may comply or be compatible with an ATM standard published by the ATM Forum titled "ATM-MPLS Network Interworking 2.0" published August 2001, and/or later versions of this standard. Of course, different and/or after-developed connection-oriented network communication protocols are equally contemplated herein.

[0141] Unless specifically stated otherwise as apparent from the foregoing

disclosure, it is appreciated that, throughout the present disclosure, discussions using terms such as “processing,” “computing,” “calculating,” “determining,” “displaying,” or the like, refer to the action and processes of a computer system, or similar electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

[0142] One or more components may be referred to herein as “configured to,” “configurable to,” “operable/operative to,” “adapted/adaptable,” “able to,” “conformable/conformed to,” etc. Those skilled in the art will recognize that “configured to” can generally encompass active-state components and/or inactive-state components and/or standby-state components, unless context requires otherwise.

[0143] Those skilled in the art will recognize that, in general, terms used herein, and especially in the appended claims (e.g., bodies of the appended claims) are generally intended as “open” terms (e.g., the term “including” should be interpreted as “including but not limited to,” the term “having” should be interpreted as “having at least,” the term “includes” should be interpreted as “includes but is not limited to,” etc.). It will be further understood by those within the art that if a specific number of an introduced claim recitation is intended, such an intent will be explicitly recited in the claim, and in the absence of such recitation no such intent is present. For example, as an aid to understanding, the following appended claims may contain usage of the introductory phrases “at least one” and “one or more” to introduce claim recitations. However, the use of such phrases should not be construed to imply that the introduction of a claim recitation by the indefinite articles “a” or “an” limits any particular claim containing such introduced claim recitation to claims containing only one such recitation, even when the same claim includes the introductory phrases “one or more” or “at least one” and indefinite articles such as “a” or “an” (e.g., “a” and/or “an” should

typically be interpreted to mean “at least one” or “one or more”); the same holds true for the use of definite articles used to introduce claim recitations.

[0144] In addition, even if a specific number of an introduced claim recitation is explicitly recited, those skilled in the art will recognize that such recitation should typically be interpreted to mean at least the recited number (e.g., the bare recitation of “two recitations,” without other modifiers, typically means at least two recitations, or two or more recitations). Furthermore, in those instances where a convention analogous to “at least one of A, B, and C, etc.” is used, in general such a construction is intended in the sense one having skill in the art would understand the convention (e.g., “a system having at least one of A, B, and C” would include but not be limited to systems that have A alone, B alone, C alone, A and B together, A and C together, B and C together, and/or A, B, and C together, etc.). In those instances where a convention analogous to “at least one of A, B, or C, etc.” is used, in general such a construction is intended in the sense one having skill in the art would understand the convention (e.g., “a system having at least one of A, B, or C” would include but not be limited to systems that have A alone, B alone, C alone, A and B together, A and C together, B and C together, and/or A, B, and C together, etc.). It will be further understood by those within the art that typically a disjunctive word and/or phrase presenting two or more alternative terms, whether in the description, claims, or drawings, should be understood to contemplate the possibilities of including one of the terms, either of the terms, or both terms unless context dictates otherwise. For example, the phrase “A or B” will be typically understood to include the possibilities of “A” or “B” or “A and B.”

[0145] With respect to the appended claims, those skilled in the art will appreciate that recited operations therein may generally be performed in any order. Also, although various operational flow diagrams are presented in a sequence(s), it should be understood that the various operations may be performed in other orders than those which are illustrated, or may be performed concurrently. Examples of such alternate

orderings may include overlapping, interleaved, interrupted, reordered, incremental, preparatory, supplemental, simultaneous, reverse, or other variant orderings, unless context dictates otherwise. Furthermore, terms like “responsive to,” “related to,” or other past-tense adjectives are generally not intended to exclude such variants, unless context dictates otherwise.

[0146] It is worthy to note that any reference to “one aspect,” “an aspect,” “an exemplification,” “one exemplification,” and the like means that a particular feature, structure, or characteristic described in connection with the aspect is included in at least one aspect. Thus, appearances of the phrases “in one aspect,” “in an aspect,” “in an exemplification,” and “in one exemplification” in various places throughout the specification are not necessarily all referring to the same aspect. Furthermore, the particular features, structures or characteristics may be combined in any suitable manner in one or more aspects.

[0147] As used herein, the singular form of “a”, “an”, and “the” include the plural references unless the context clearly dictates otherwise.

[0148] Any patent application, patent, non-patent publication, or other disclosure material referred to in this specification and/or listed in any Application Data Sheet is incorporated by reference herein, to the extent that the incorporated materials is not inconsistent herewith. As such, and to the extent necessary, the disclosure as explicitly set forth herein supersedes any conflicting material incorporated herein by reference. Any material, or portion thereof, that is said to be incorporated by reference herein, but which conflicts with existing definitions, statements, or other disclosure material set forth herein will only be incorporated to the extent that no conflict arises between that incorporated material and the existing disclosure material. None is admitted to be prior art.

[0149] In summary, numerous benefits have been described which result from

employing the concepts described herein. The foregoing description of the one or more forms has been presented for purposes of illustration and description. It is not intended to be exhaustive or limiting to the precise form disclosed. Modifications or variations are possible in light of the above teachings. The one or more forms were chosen and described in order to illustrate principles and practical application to thereby enable one of ordinary skill in the art to utilize the various forms and with various modifications as are suited to the particular use contemplated. It is intended that the claims submitted herewith define the overall scope.

ABSTRACT

Devices, systems, and methods for generating arrays are disclosed herein. In one aspect, a method for generating an array includes calculating a true metric for a peer set. The peer set includes peer values and weight percentages corresponding to each of the peer values. The method further includes determining that the peer set does not comply with at least one privacy policy rule. The method further includes generating the array based on determining that the peer set does not comply with the at least one privacy policy rule. The array represents the true metric and is generated by (i) calculating an upper bound based on the true metric, a weighted standard deviation of the peer set, and a first random number and (ii) calculating a lower bound based on the true metric, the weighted standard deviation of the peer set, and a second random number.

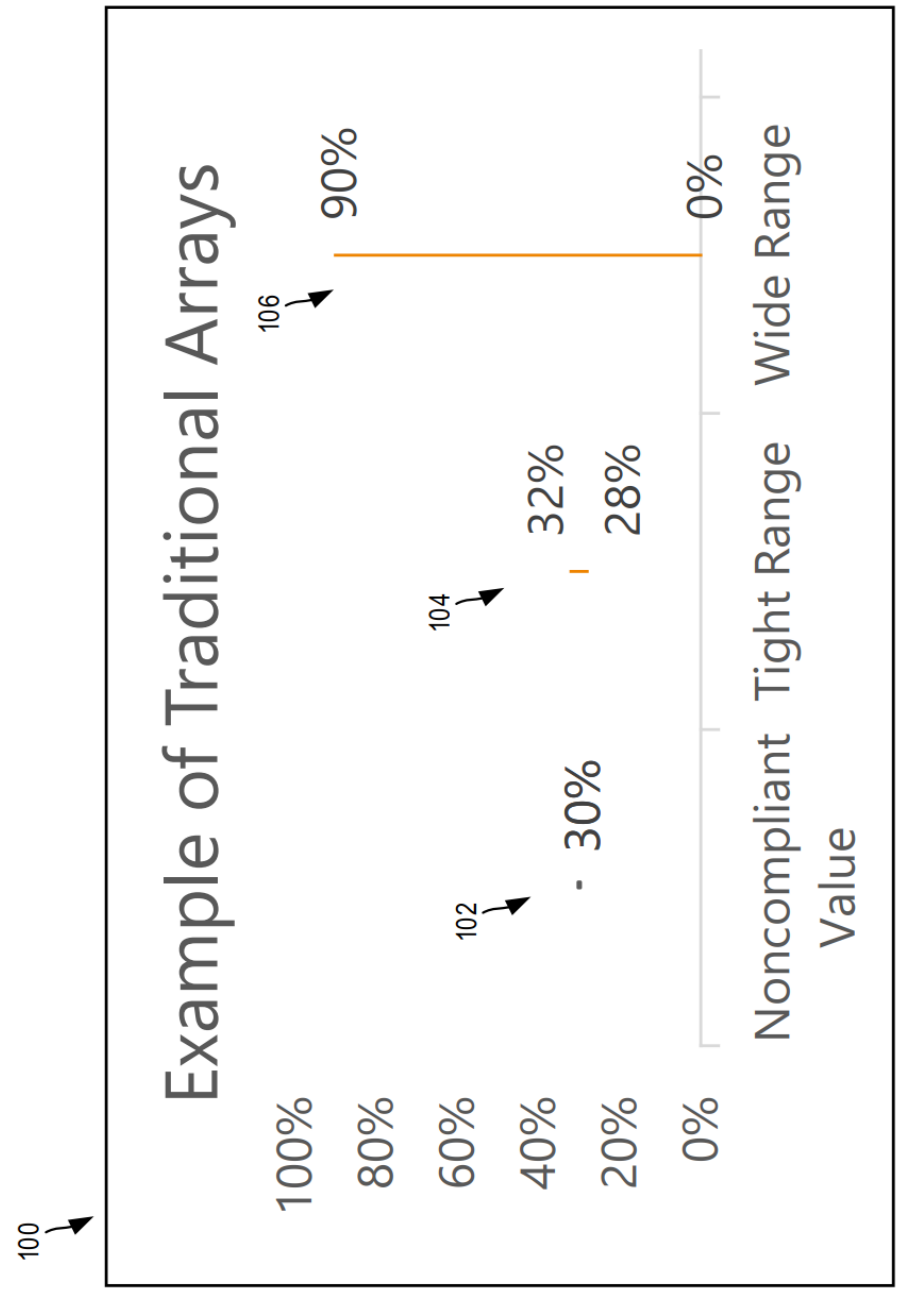
100
Example of Traditional Arrays
100%
80%
60%
40%
20%
0%
102
30%
104
32%
28%
106
90%
0%
Noncompliant
Tight Range
Wide Range
Value

FIG. 1

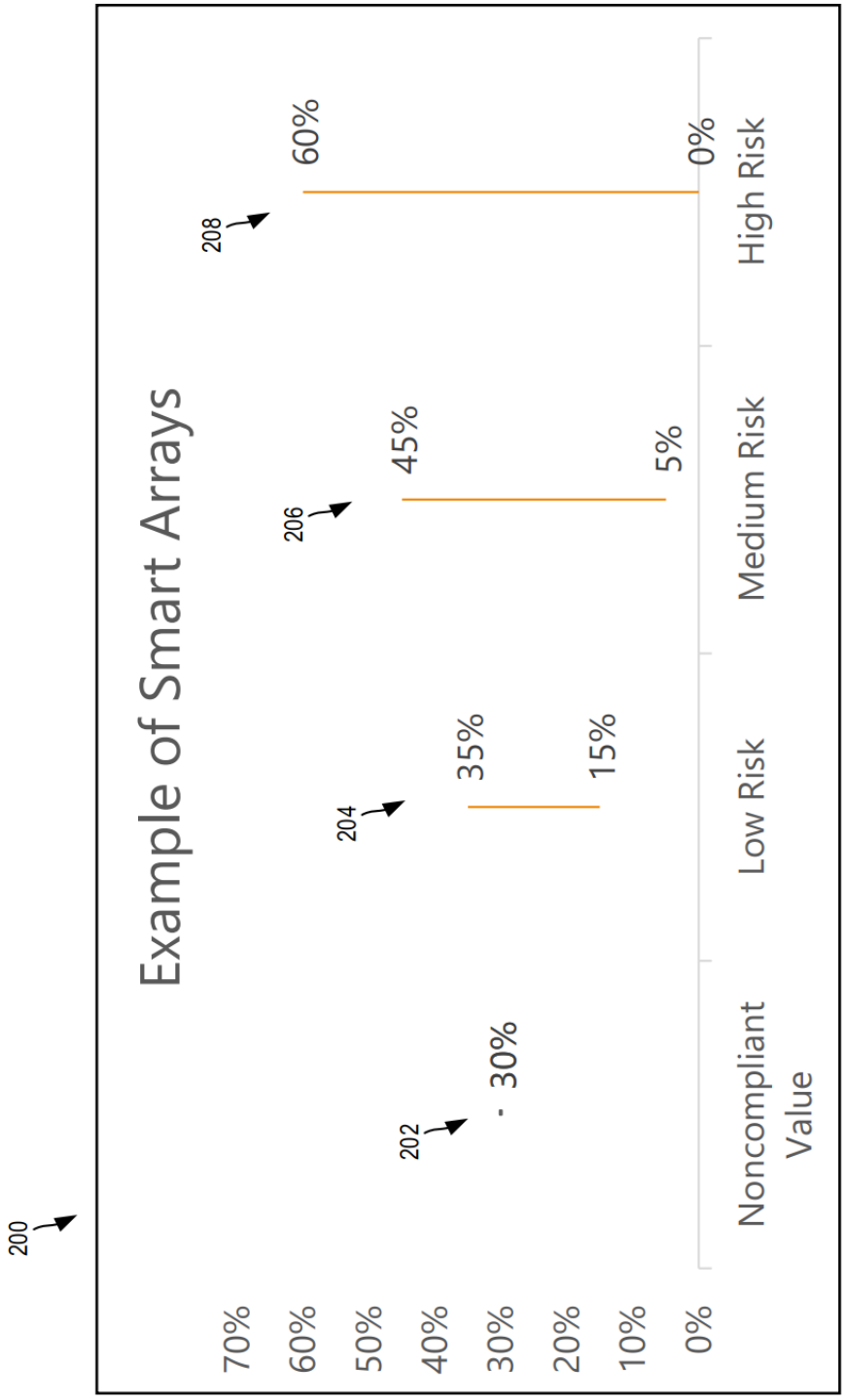
200
Example of Smart Arrays
70%
60%
50%
40%
30%
20%
10%
0%
202
30%
204
35%
15%
206
45%
5%
208
60%
0%
Noncompliant Value
Low Risk
Medium Risk
High Risk

FIG. 2

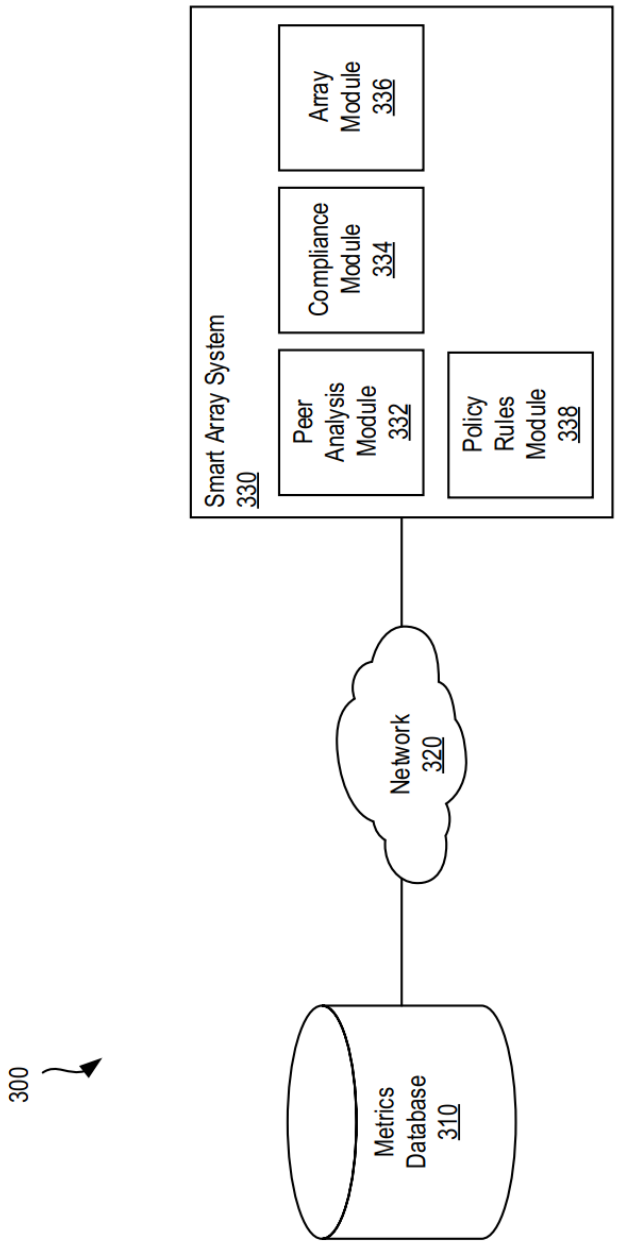
300
Metrics Database 310
Network 320
Smart Array System 330
Peer Analysis Module 332
Compliance Module 334
Array Module 336
Policy Rules Module 338

FIG. 3

4/13

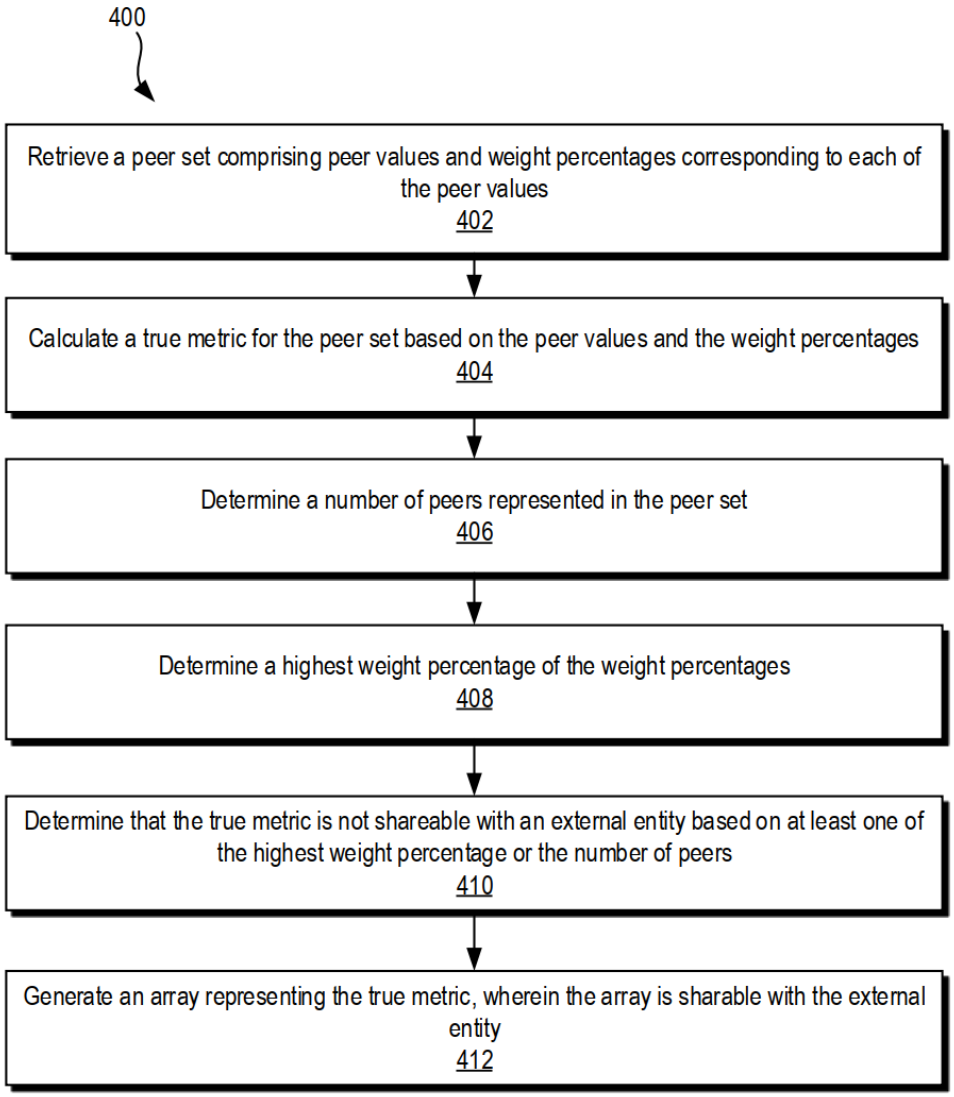
400
Retrieve a peer set comprising peer values and weight percentages corresponding to each of the peer values
402
Calculate a true metric for the peer set based on the peer values and the weight percentages
404
Determine a number of peers represented in the peer set
406
Determine a highest weight percentage of the weight percentages
408
Determine that the true metric is not shareable with an external entity based on at least one of the highest weight percentage or the number of peers
410
Generate an array representing the true metric, wherein the array is sharable with the external entity
412

FIG. 4

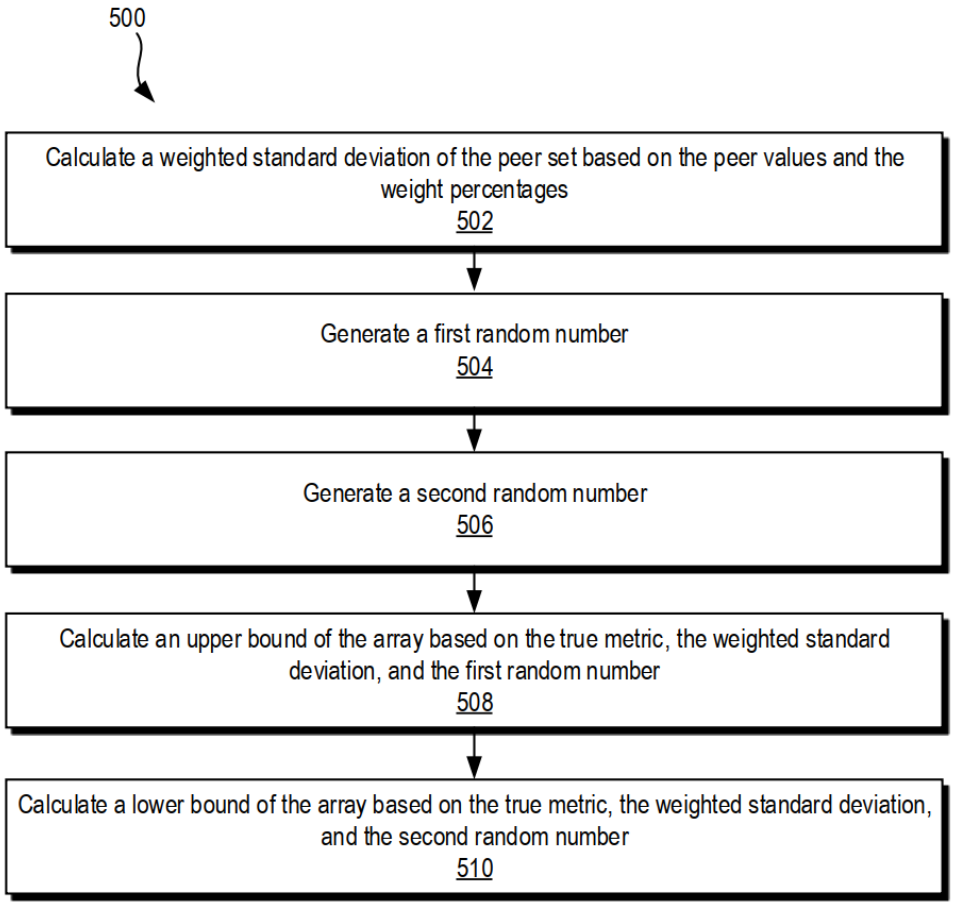
500
Calculate a weighted standard deviation of the peer set based on the peer values and the weight percentages
502
Generate a first random number
504
Generate a second random number
506
Calculate an upper bound of the array based on the true metric, the weighted standard deviation, and the first random number
508
Calculate a lower bound of the array based on the true metric, the weighted standard deviation, and the second random number
510

FIG. 5

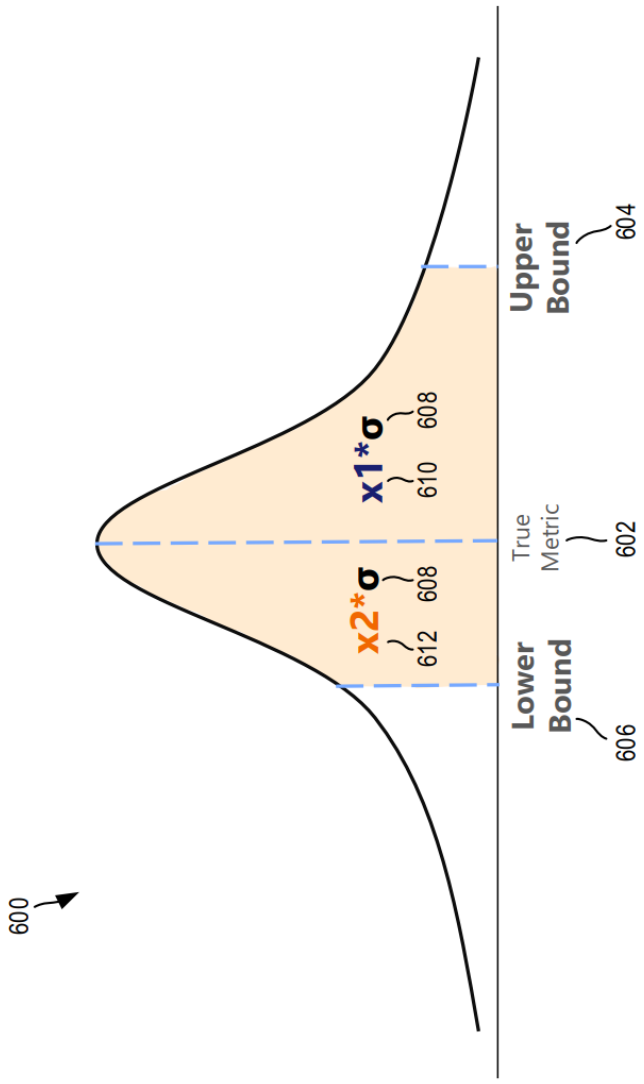
600
Upper Bound
604
x1*σ
608
610
True Metric
602
x2*σ
608
612
Lower Bound
606

FIG. 6

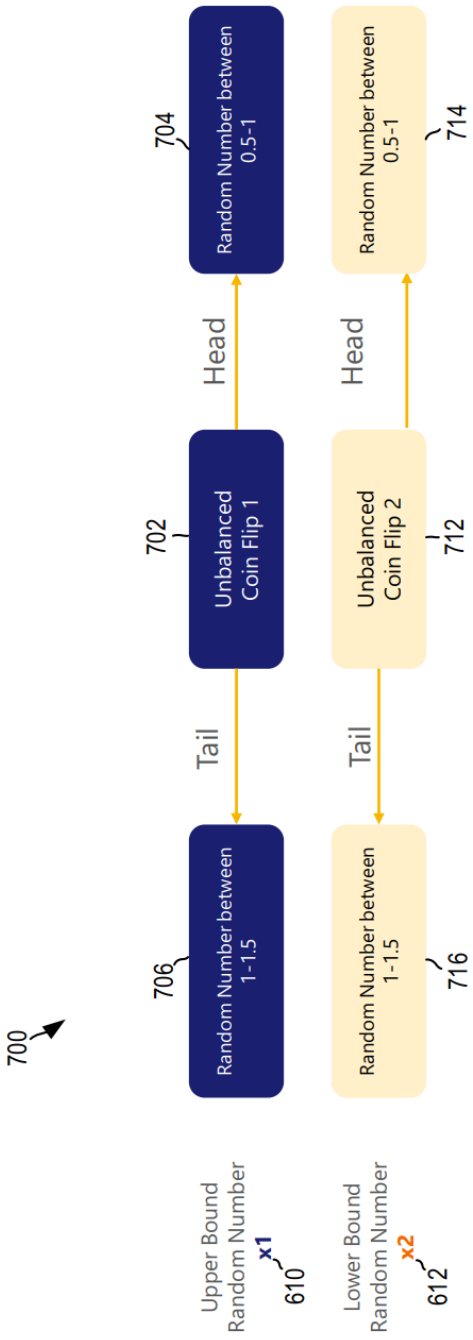
700
Random Number between 1-1.5
706
Tail
Unbalanced Coin Flip 1
702
Head
Random Number between 0.5-1
704
Random Number between 1-1.5
716
Tail
Unbalanced Coin Flip 2
712
Head
Random Number between 0.5-1
714
Upper Bound Random Number x1
610
Lower Bound Random Number x2
612

FIG. 7

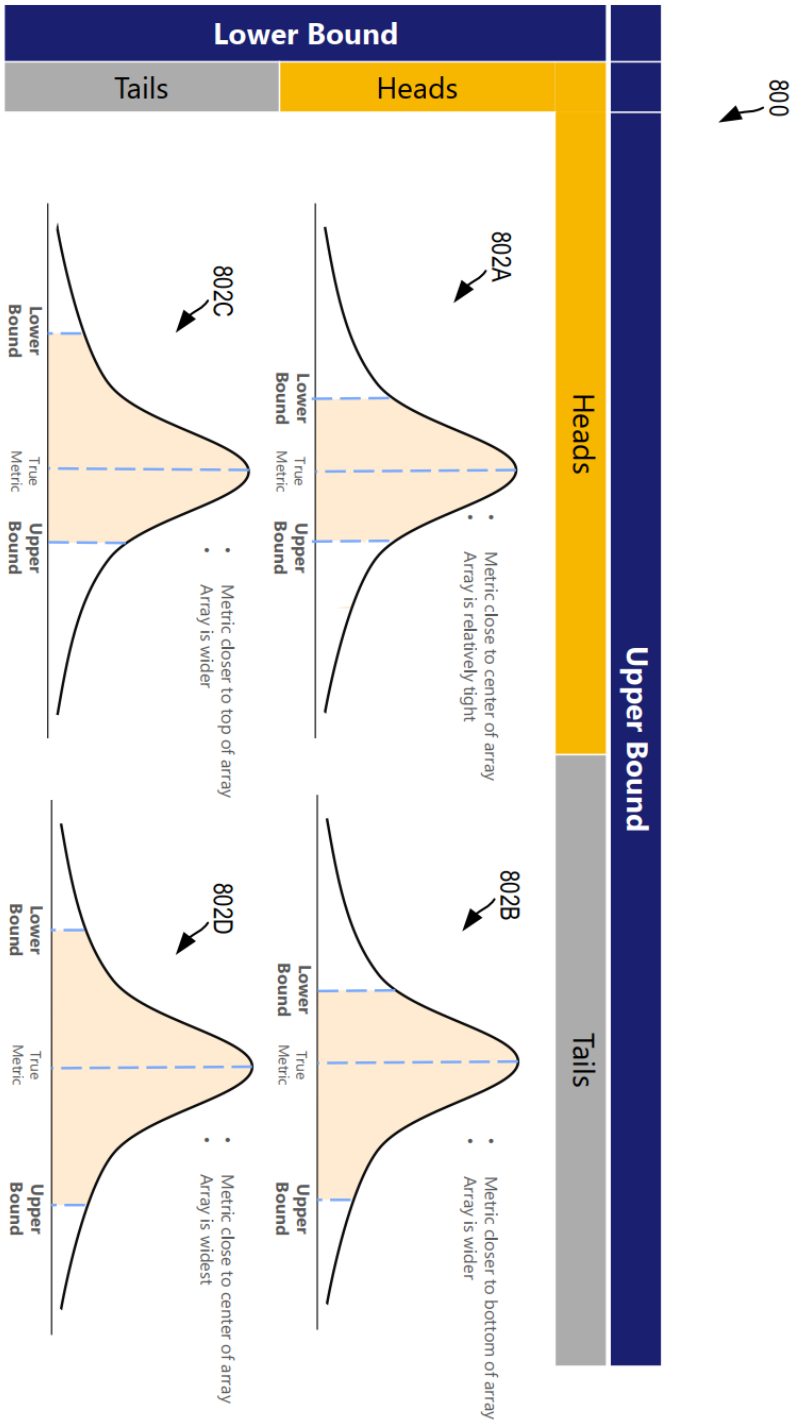
800
Upper Bound
Heads
Tails
Lower Bound
Heads
Tails
802A
Lower Bound
True Metric
Upper Bound
Metric close to center of array
Array is relatively tight
802B
Lower Bound
True Metric
Upper Bound
Metric closer to bottom of array
Array is wider
802C
Lower Bound
True Metric
Upper Bound
Metric closer to top of array
Array is wider
802D
Lower Bound
True Metric
Upper Bound
Metric close to center of array
Array is widest

FIG. 8

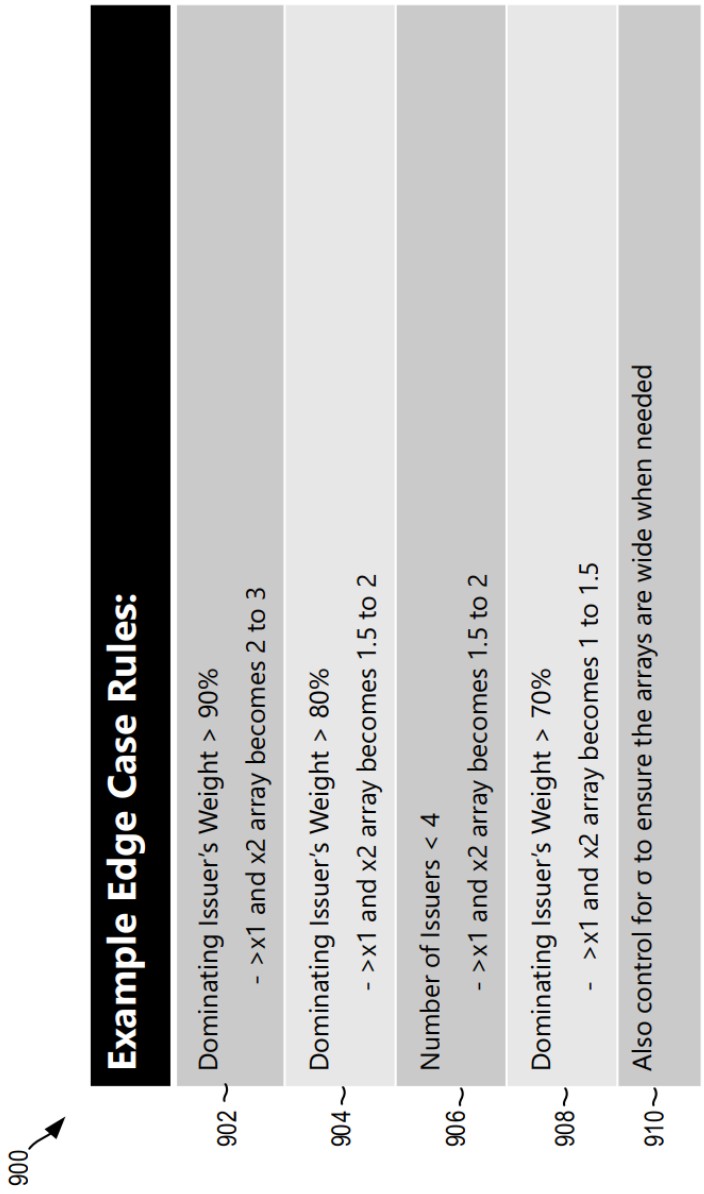
900
Example Edge Case Rules:
902
Dominating Issuer's Weight > 90%
- >x1 and x2 array becomes 2 to 3
904
Dominating Issuer's Weight > 80%
- >x1 and x2 array becomes 1.5 to 2
906
Number of Issuers < 4
- >x1 and x2 array becomes 1.5 to 2
908
Dominating Issuer's Weight > 70%
- >x1 and x2 array becomes 1 to 1.5
910
Also control for σ to ensure the arrays are wide when needed

FIG. 9

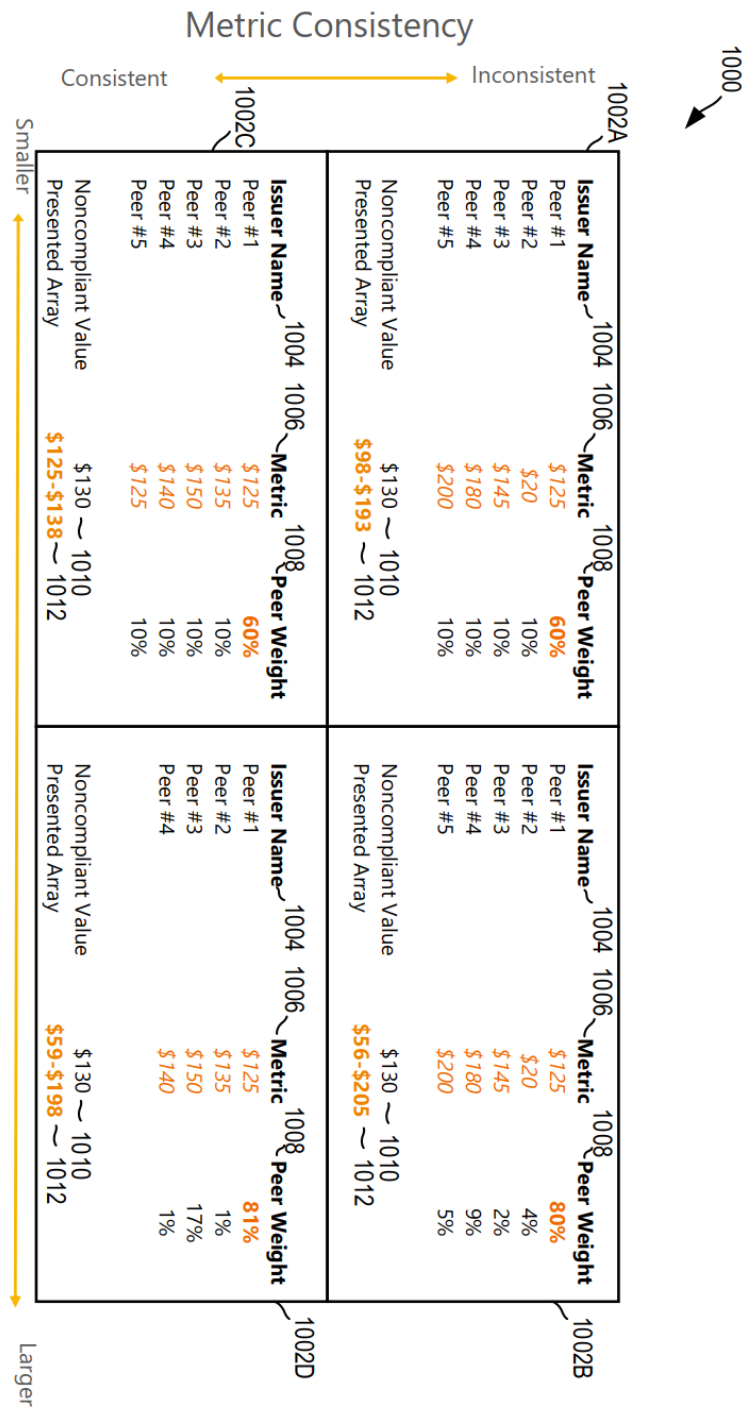
1000
Metric Consistency
Consistent
Inconsistent
Smaller
Larger
Largest Peer
1002A
1002B
1002C
1002D
1004
1006
1008
1010
1012
Issue Name
Metric
Peer Weight
Peer #1
Peer #2
Peer #3
Peer #4
Peer #5
$125
$20
$145
$180
$200
60%
10%
10%
10%
10%
Noncompliant Value Presented Array
$130 ~ 1010
$98-$193 ~ 1012
Issue Name
Metric
Peer Weight
Peer #1
Peer #2
Peer #3
Peer #4
Peer #5
$125
$135
$150
$140
$125
60%
10%
10%
10%
10%
Noncompliant Value Presented Array
$130 ~ 1010
$125-$138 ~ 1012
Issue Name
Metric
Peer Weight
Peer #1
Peer #2
Peer #3
Peer #4
Peer #5
$125
$20
$145
$180
$200
80%
4%
2%
9%
5%
Noncompliant Value Presented Array
$130 ~ 1010
$56-$205 ~ 1012
Issue Name
Metric
Peer Weight
Peer #1
Peer #2
Peer #3
Peer #4
$125
$135
$150
$140
81%
1%
17%
1%
Noncompliant Value Presented Array
$130 ~ 1010
$59-$198 ~ 1012

FIG. 10

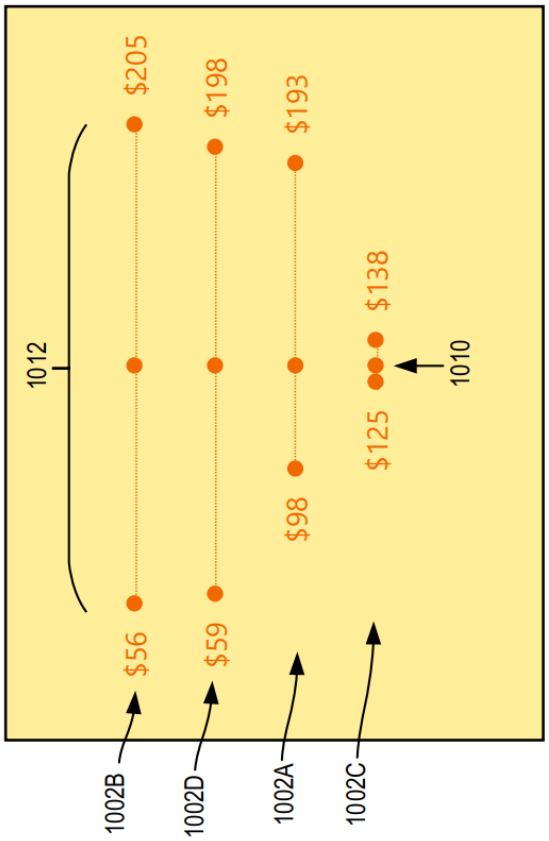

FIG. 11

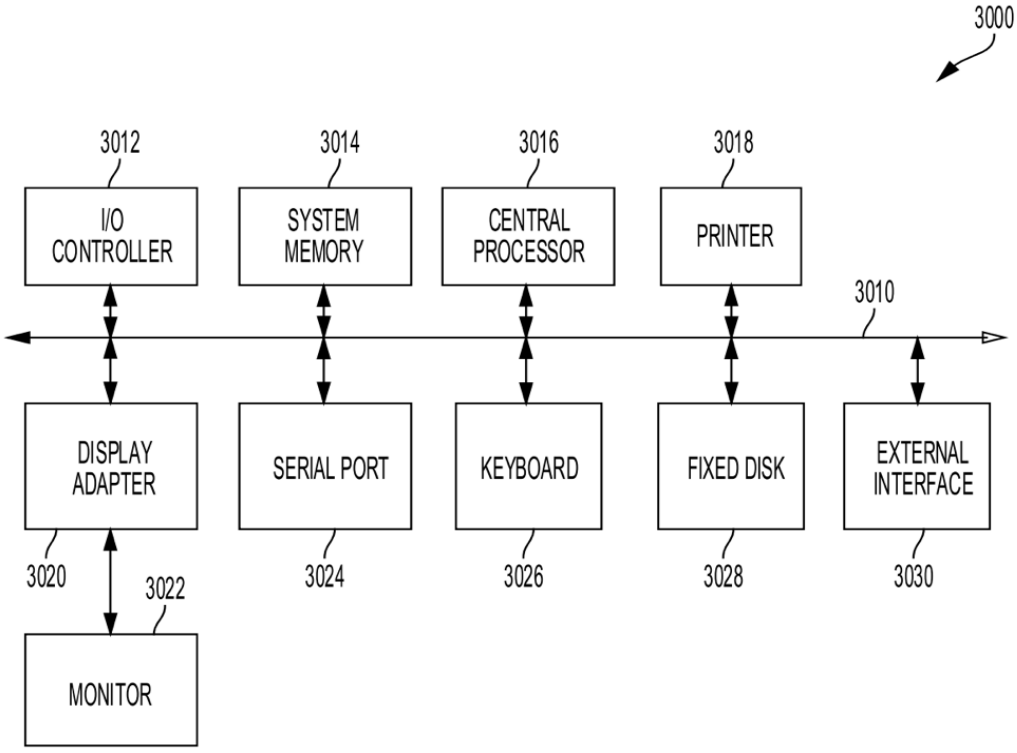
3000
3012
3014
3016
3018
I/O CONTROLLER
SYSTEM MEMORY
CENTRAL PROCESSOR
PRINTER
3010
DISPLAY ADAPTER
SERIAL PORT
KEYBOARD
FIXED DISK
EXTERNAL INTERFACE
3020
3022
3024
3026
3028
3030
MONITOR

FIG. 12

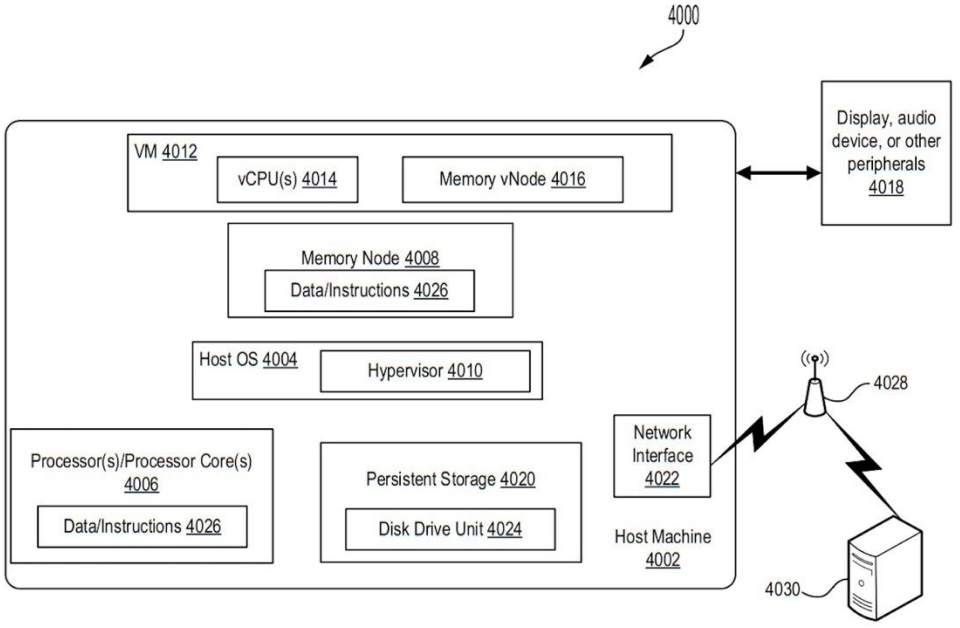
4000
VM 4012
vCPU(s) 4014
Memory vNode 4016
Display, audio device, or other peripherals 4018
Memory Node 4008
Data/Instructions 4026
Host OS 4004
Hypervisor 4010
Processor(s)/Processor Core(s) 4006
Data/Instructions 4026
Persistent Storage 4020
Disk Drive Unit 4024
Network Interface 4022
Host Machine 4002
4028
4030

FIG. 13