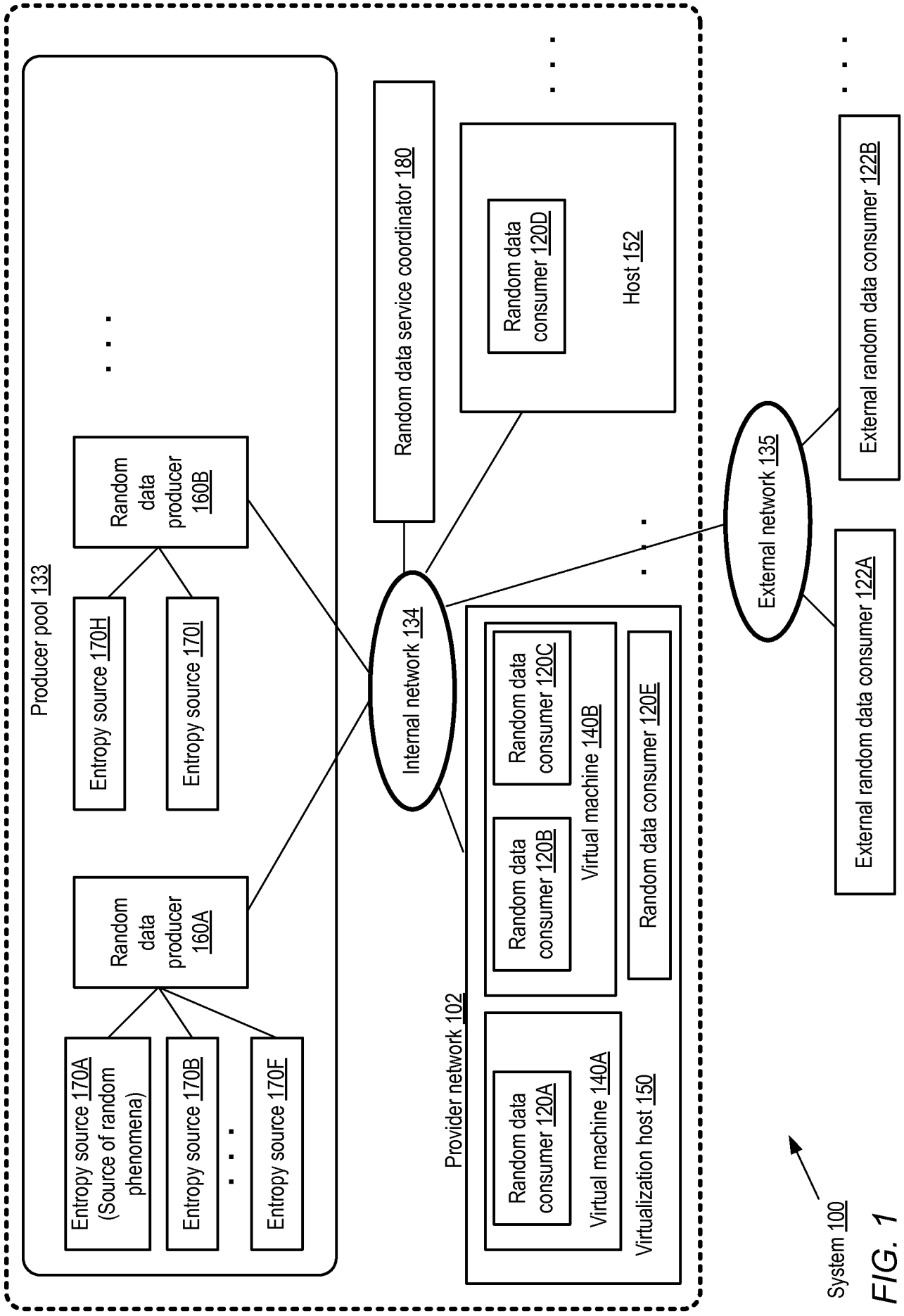
System 100
Producer pool 133
Entropy source (Source of random phenomena) 170A
Entropy source 170B
Entropy source 170F
Random data producer 160A
Entropy source 170H
Entropy source 170I
Random data producer 160B
Internal network 134
Random data service coordinator 180
Provider network 102
Virtualization host 150
Virtual machine 140A
Random data consumer 120A
Virtual machine 140B
Random data consumer 120B
Random data consumer 120C
Random data consumer 120E
Host 152
Random data consumer 120D
External network 135
External random data consumer 122A
External random data consumer 122B

FIG. 1

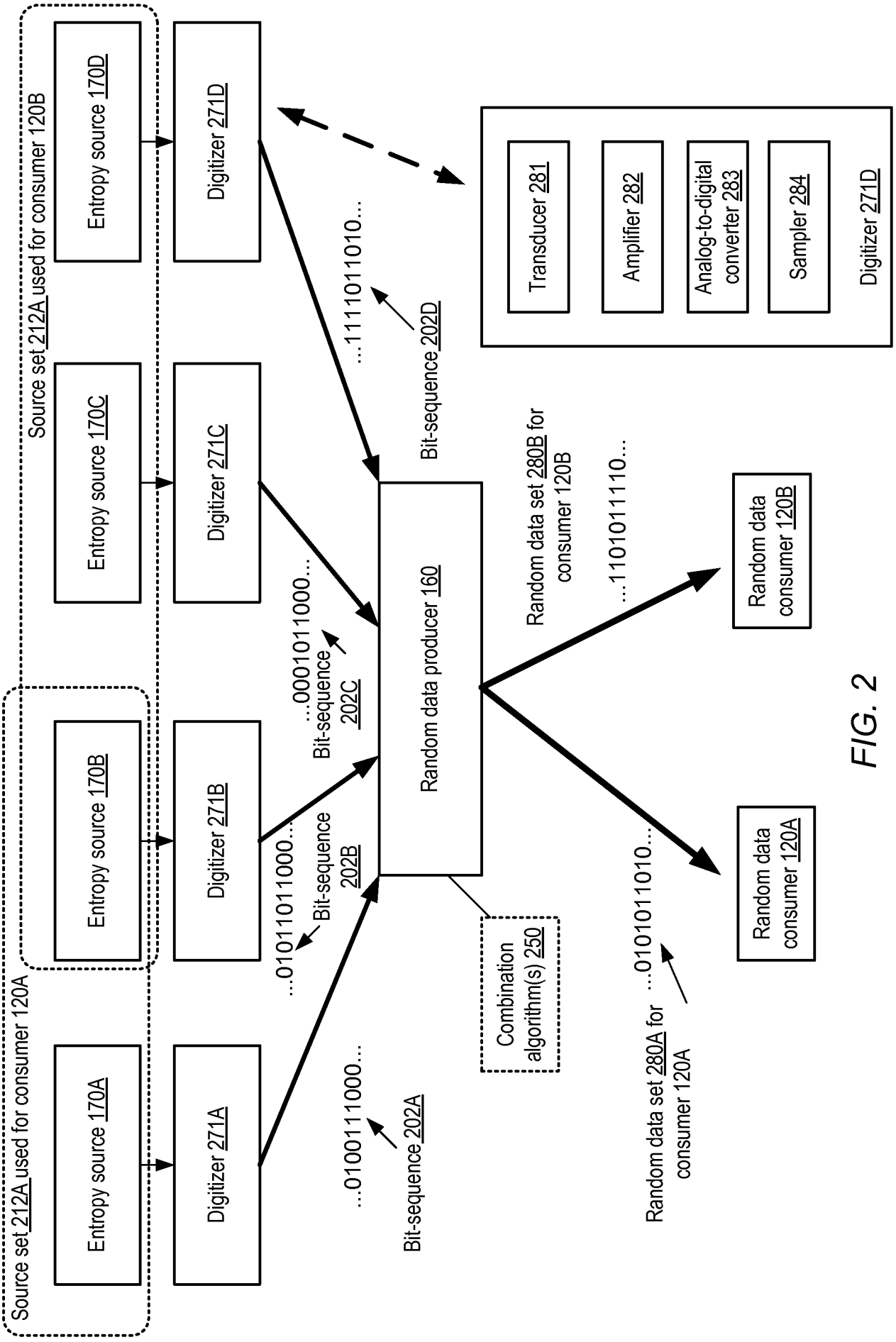
Source set 212A used for consumer 120A
Entropy source 170A
Entropy source 170B
Source set 212A used for consumer 120B
Entropy source 170C
Entropy source 170D
Digitizer 271A
Digitizer 271B
Digitizer 271C
Digitizer 271D
...0100111000...
Bit-sequence 202A
...0101011000...
Bit-sequence 202B
...0001011000...
Bit-sequence 202C
...1111011010...
Bit-sequence 202D
Random data producer 160
Combination algorithm(s) 250
Random data set 280A for consumer 120A
...0101011010...
Random data set 280B for consumer 120B
...1101011110...
Random data consumer 120A
Random data consumer 120B
Transducer 281
Amplifier 282
Analog-to-digital converter 283
Sampler 284
Digitizer 271D

FIG. 2

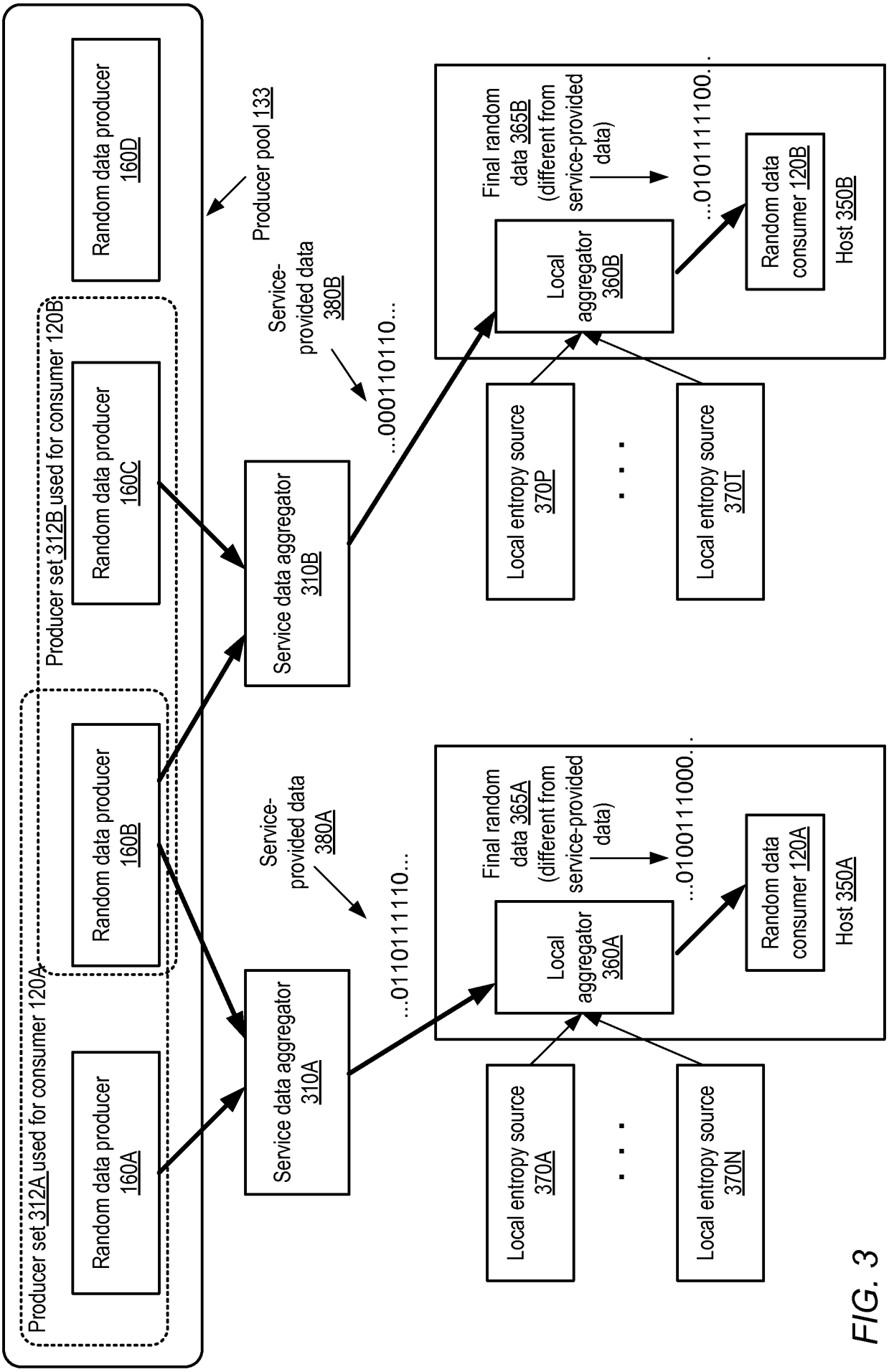
Producer pool 133
Producer set 312A used for consumer 120A
Producer set 312B used for consumer 120B
Random data producer 160A
Random data producer 160B
Random data producer 160C
Random data producer 160D
Service data aggregator 310A
Service data aggregator 310B
Service-provided data 380A
...01101111110...
Service-provided data 380B
...00011011O...
Local entropy source 370A
Local entropy source 370N
Local entropy source 370P
Local entropy source 370T
Local aggregator 360A
Local aggregator 360B
Final random data 365A (different from service-provided data)
...0100111000...
Final random data 365B (different from service-provided data)
...0101111100...
Random data consumer 120A
Random data consumer 120B
Host 350A
Host 350B

FIG. 3

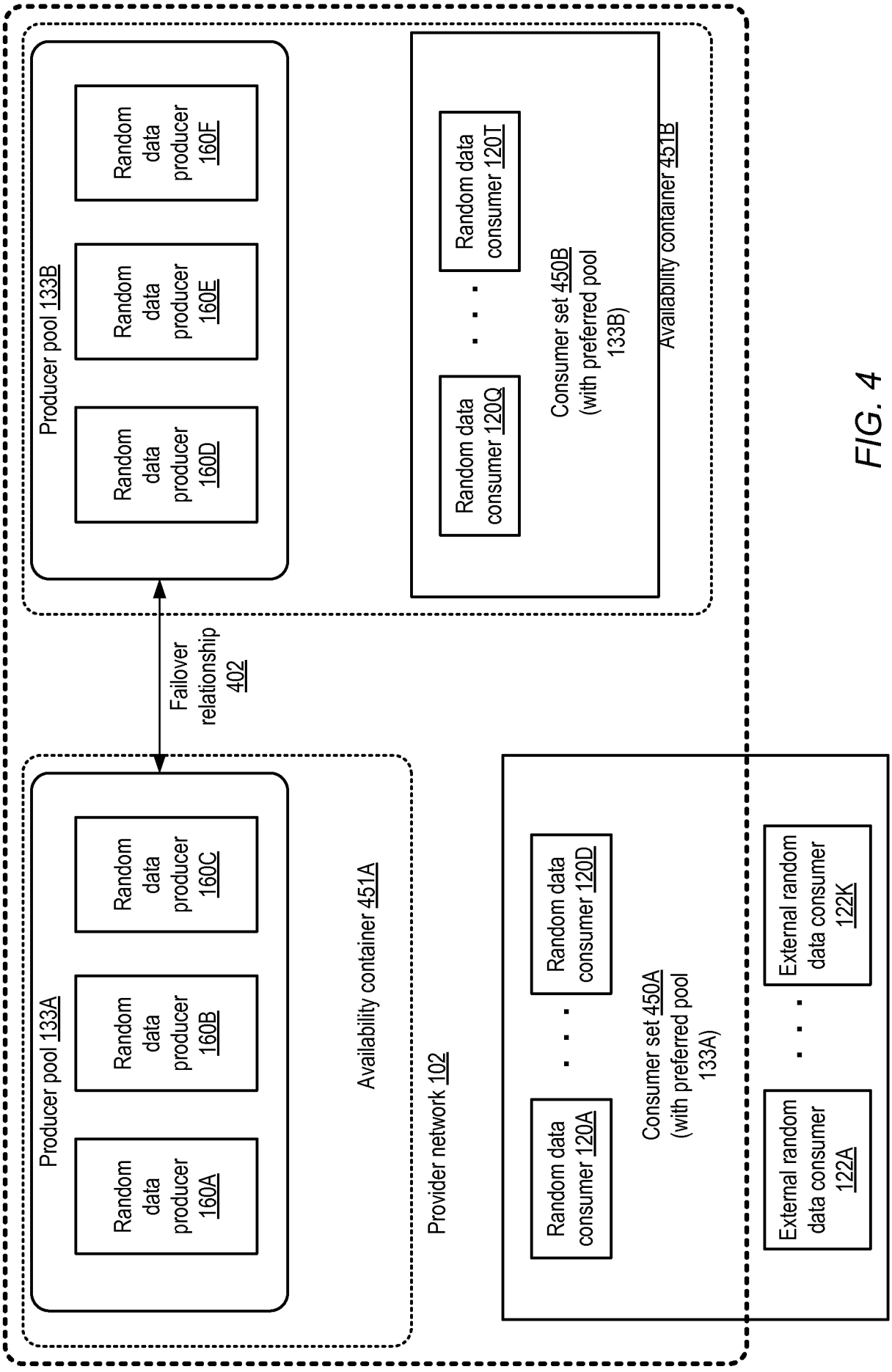
Provider network 102
Availability container 451A
Producer pool 133A
Random data producer 160A
Random data producer 160B
Random data producer 160C
Failover relationship 402
Availability container 451B
Producer pool 133B
Random data producer 160D
Random data producer 160E
Random data producer 160F
Consumer set 450A (with preferred pool 133A)
Random data consumer 120A
Random data consumer 120D
External random data consumer 122A
External random data consumer 122K
Consumer set 450B (with preferred pool 133B)
Random data consumer 120Q
Random data consumer 120T

FIG. 4

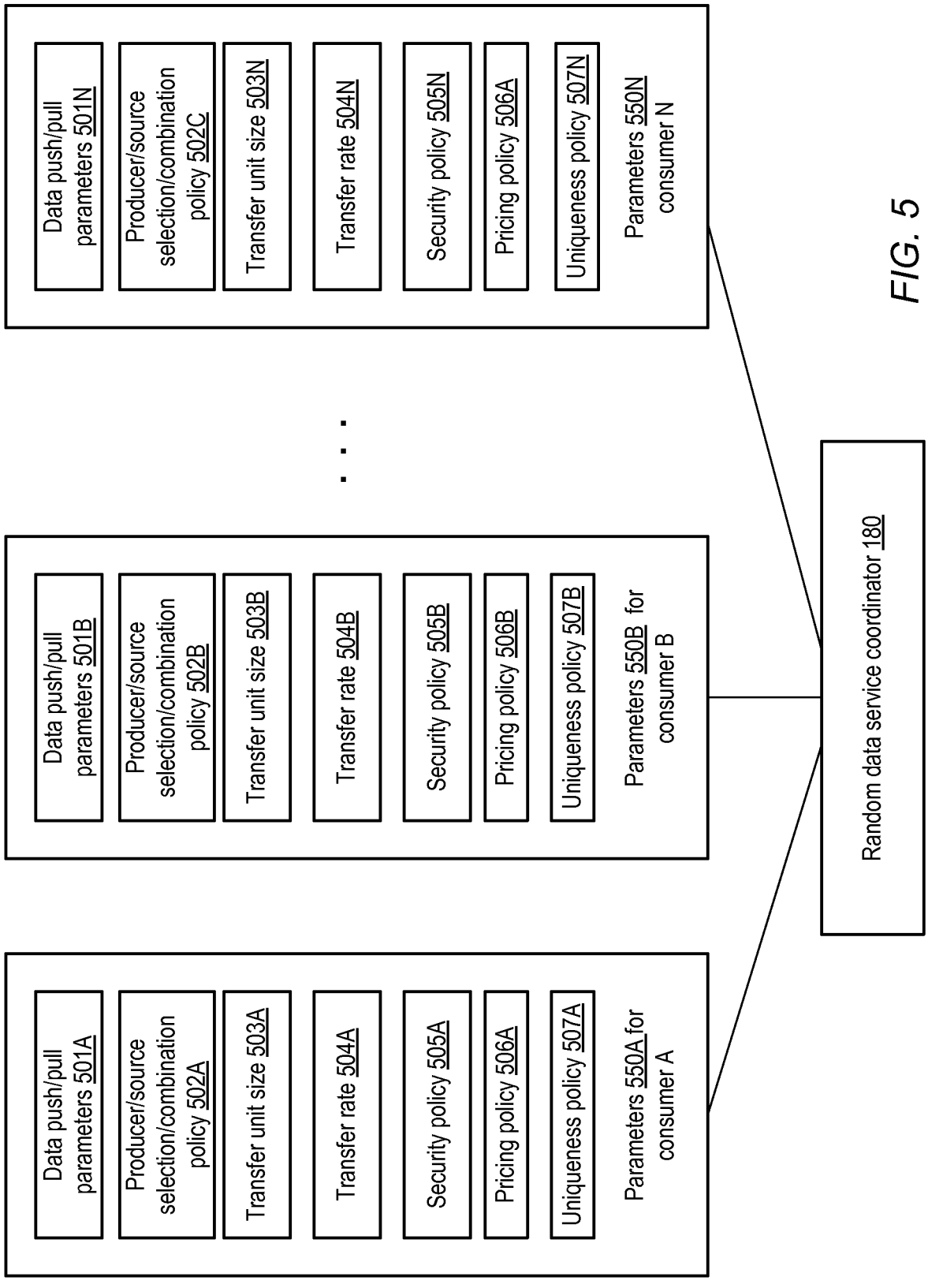
Data push/pull parameters 501A
Producer/source selection/combination policy 502A
Transfer unit size 503A
Transfer rate 504A
Security policy 505A
Pricing policy 506A
Uniqueness policy 507A
Parameters 550A for consumer A
Data push/pull parameters 501B
Producer/source selection/combination policy 502B
Transfer unit size 503B
Transfer rate 504B
Security policy 505B
Pricing policy 506B
Uniqueness policy 507B
Parameters 550B for consumer B
. . .
Data push/pull parameters 501N
Producer/source selection/combination policy 502C
Transfer unit size 503N
Transfer rate 504N
Security policy 505N
Pricing policy 506A
Uniqueness policy 507N
Parameters 550N consumer N
Random data service coordinator 180

FIG. 5

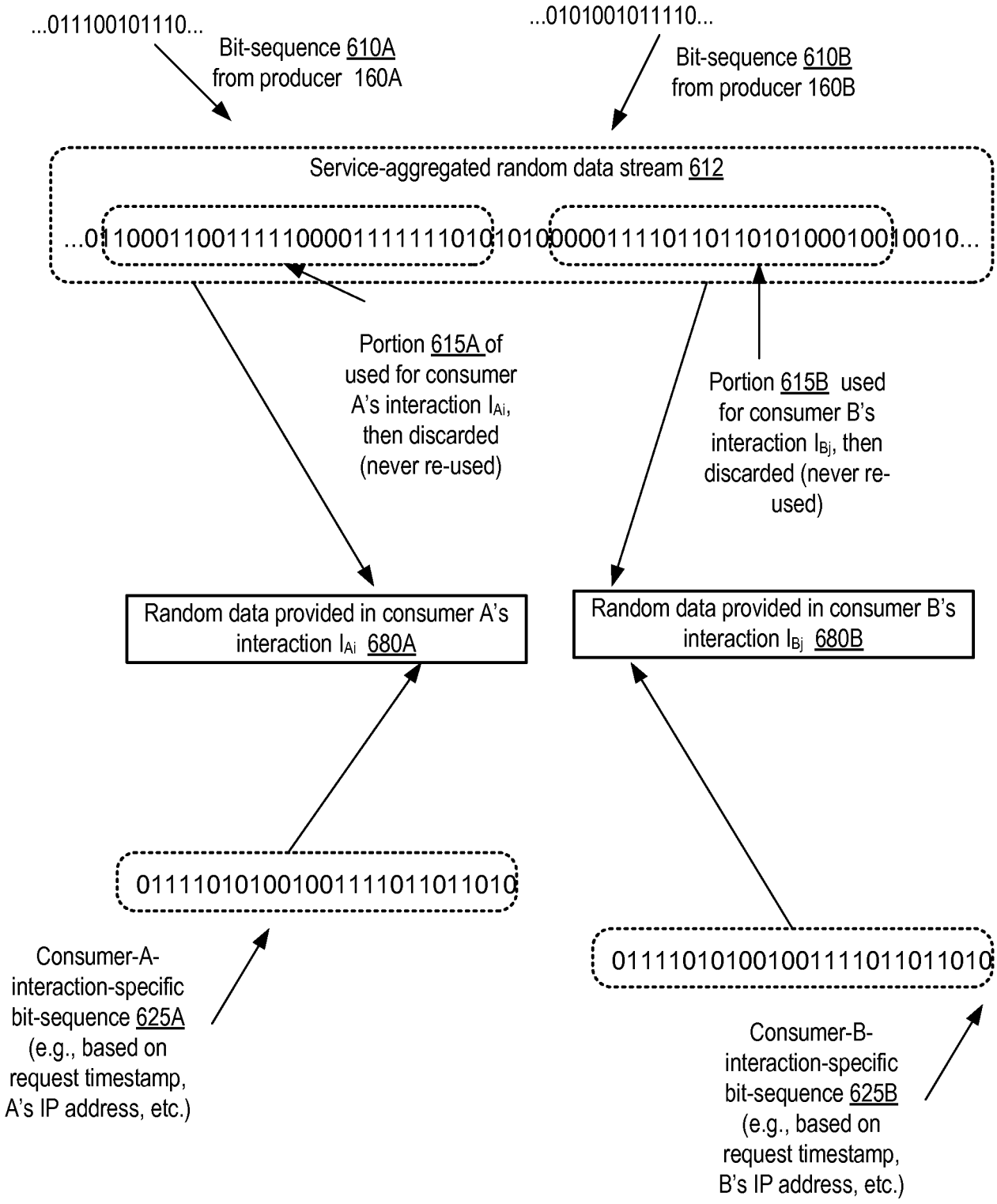
...011100101110...
Bit-sequence 610A from producer 160A
...0101001011110...
Bit-sequence 610B from producer 160B
Service-aggregated random data stream 612
...0110001100111110000111111101010100000111101101101010001001 0010...
Portion 615A of used for consumer A's interaction I_{Ai}, then discarded (never re-used)
Portion 615B used for consumer B's interaction I_{Bj}, then discarded (never re-used)
Random data provided in consumer A's interaction I_{Ai} 680A
Random data provided in consumer B's interaction I_{Bj} 680B
01111010100100111101101010
01111010100100111101101010
Consumer-A-interaction-specific bit-sequence 625A (e.g., based on request timestamp, A's IP address, etc.)
Consumer-B-interaction-specific bit-sequence 625B (e.g., based on request timestamp, B's IP address, etc.)

FIG. 6

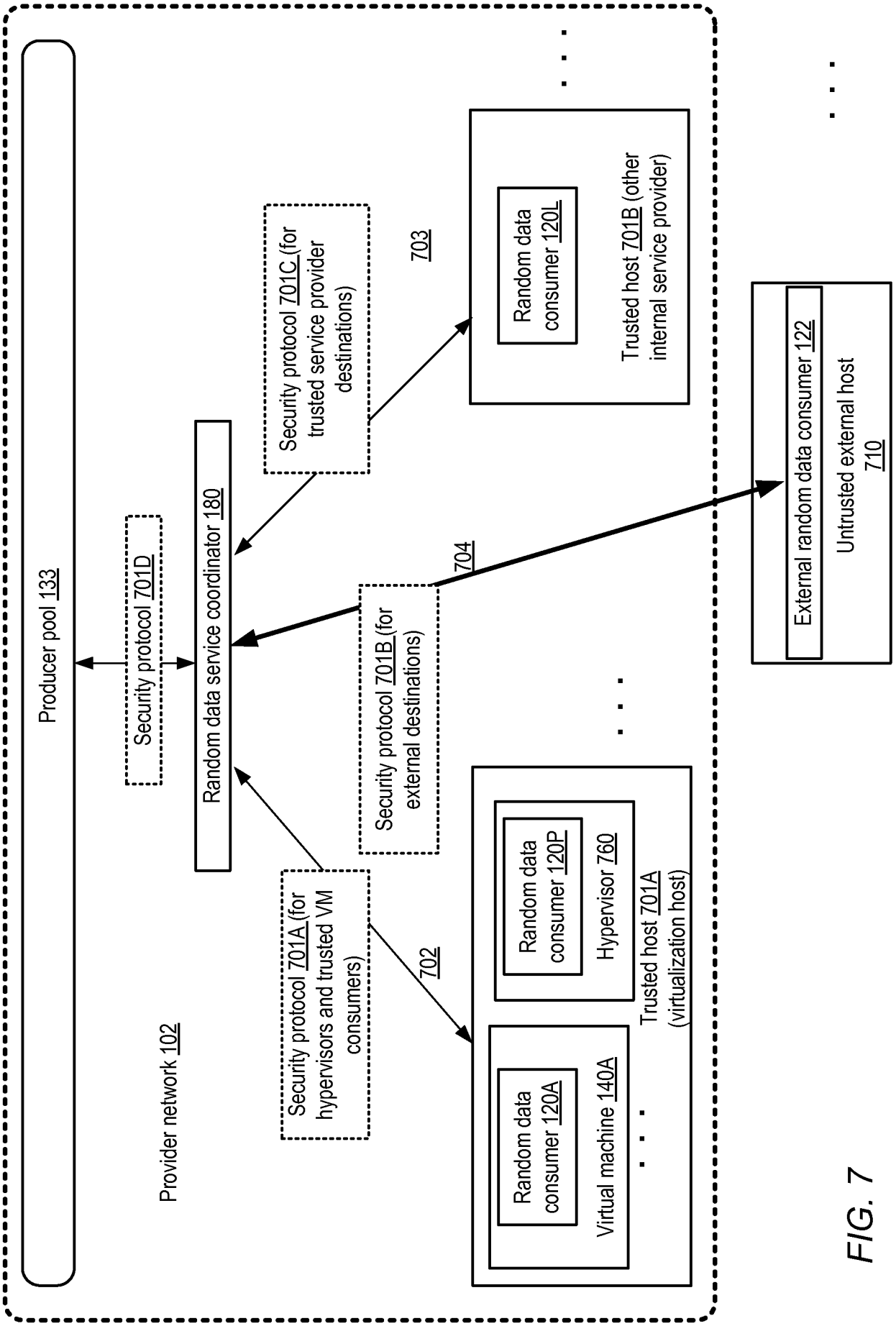
Producer pool 133
Security protocol 701D
Random data service coordinator 180
Security protocol 701C (for trusted service provider destinations)
703
Random data consumer 120L
Trusted host 701B (other internal service provider)
External random data consumer 122
Untrusted external host 710
704
Security protocol 701B (for external destinations)
Provider network 102
Security protocol 701A (for hypervisors and trusted VM consumers)
702
Random data consumer 120P
Hypervisor 760
Random data consumer 120A
Virtual machine 140A
Trusted host 701A (virtualization host)

FIG. 7

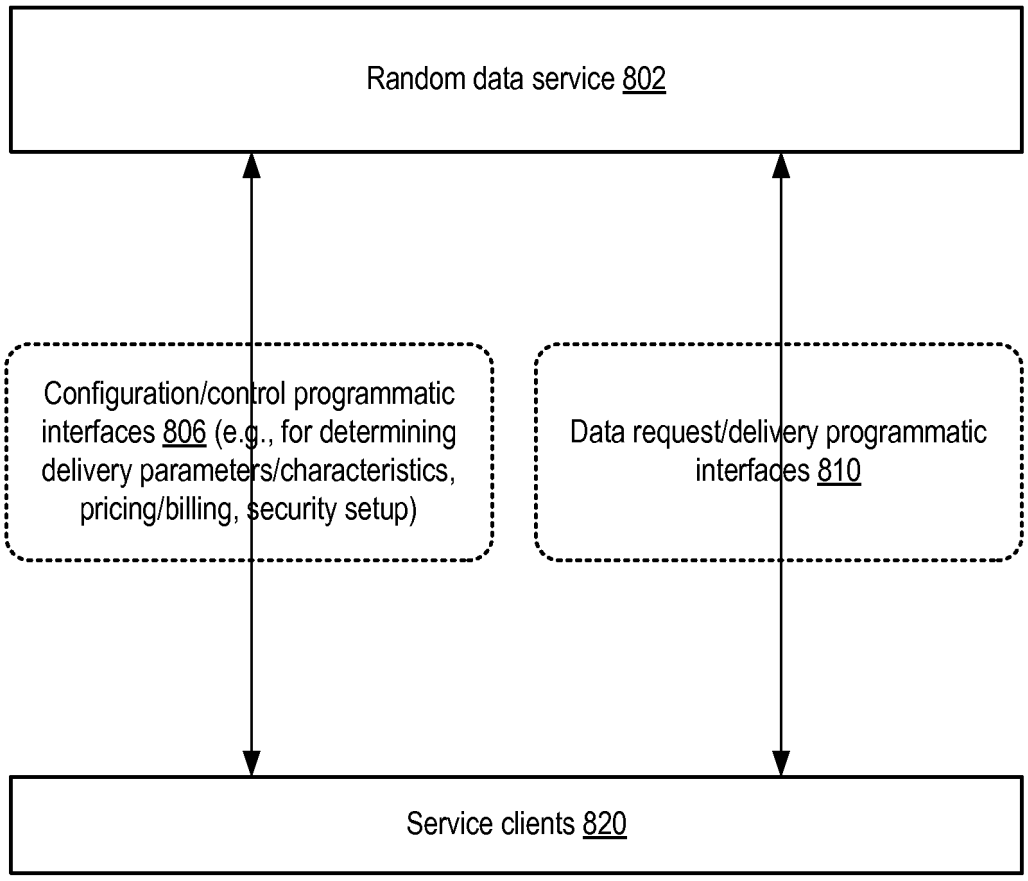
Random data service 802
Configuration/control programmatic interfaces 806 (e.g., for determining delivery parameters/characteristics, pricing/billing, security setup)
Data request/delivery programmatic interfaces 810
Service clients 820

FIG. 8

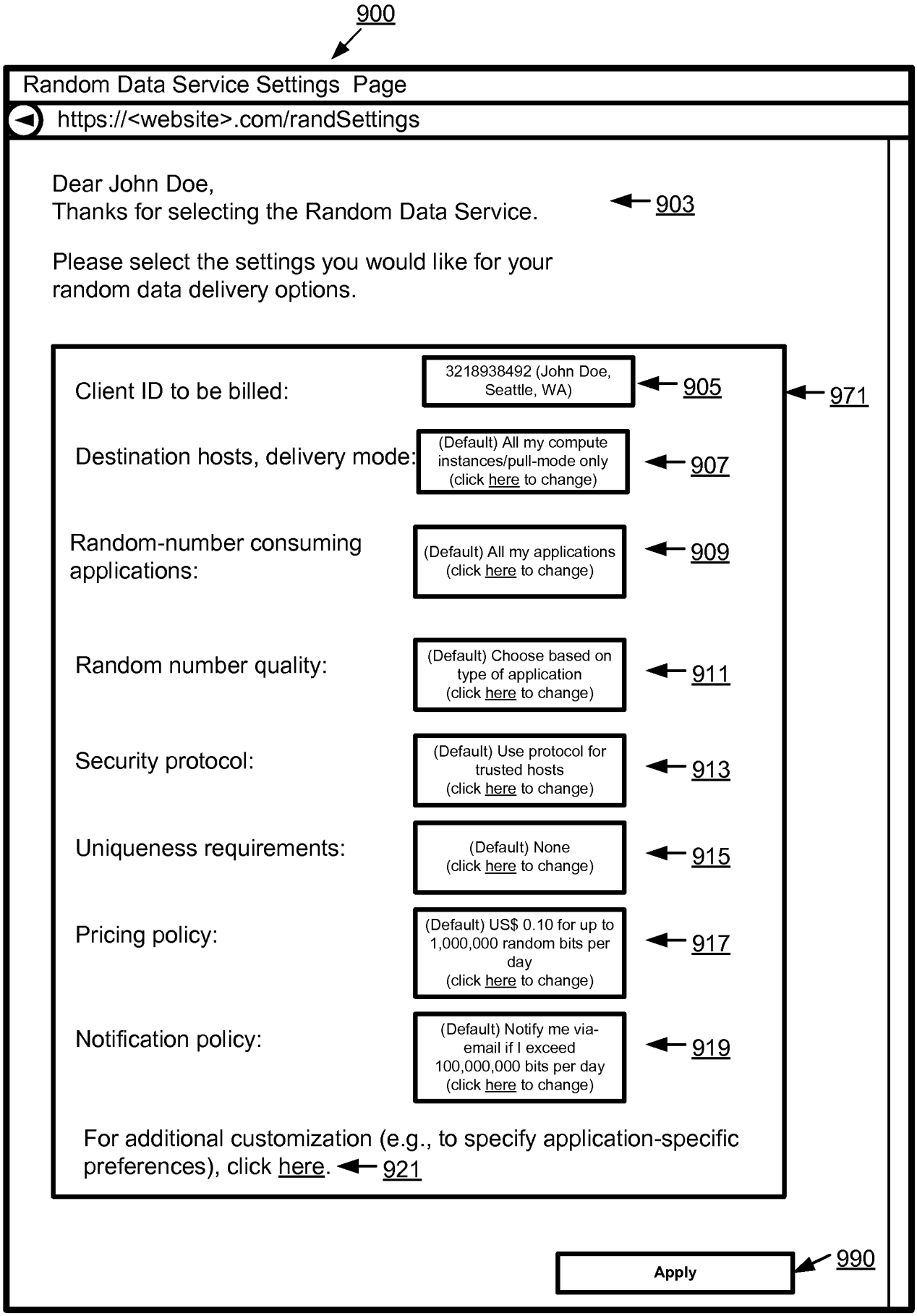
900
Random Data Service Settings Page
https://<website>.com/randSettings
Dear John Doe,
Thanks for selecting the Random Data Service.
903
Please select the settings you would like for your random data delivery options.
Client ID to be billed:
3218938492 (John Doe, Seattle, WA)
905
971
Destination hosts, delivery mode:
(Default) All my compute instances/pull-mode only (click here to change)
907
Random-number consuming applications:
(Default) All my applications (click here to change)
909
Random number quality:
(Default) Choose based on type of application (click here to change)
911
Security protocol:
(Default) Use protocol for trusted hosts (click here to change)
913
Uniqueness requirements:
(Default) None (click here to change)
915
Pricing policy:
(Default) US$ 0.10 for up to 1,000,000 random bits per day (click here to change)
917
Notification policy:
(Default) Notify me via-email if I exceed 100,000,000 bits per day (click here to change)
919
For additional customization (e.g., to specify application-specific preferences), click here.
921
Apply
990

FIG. 9

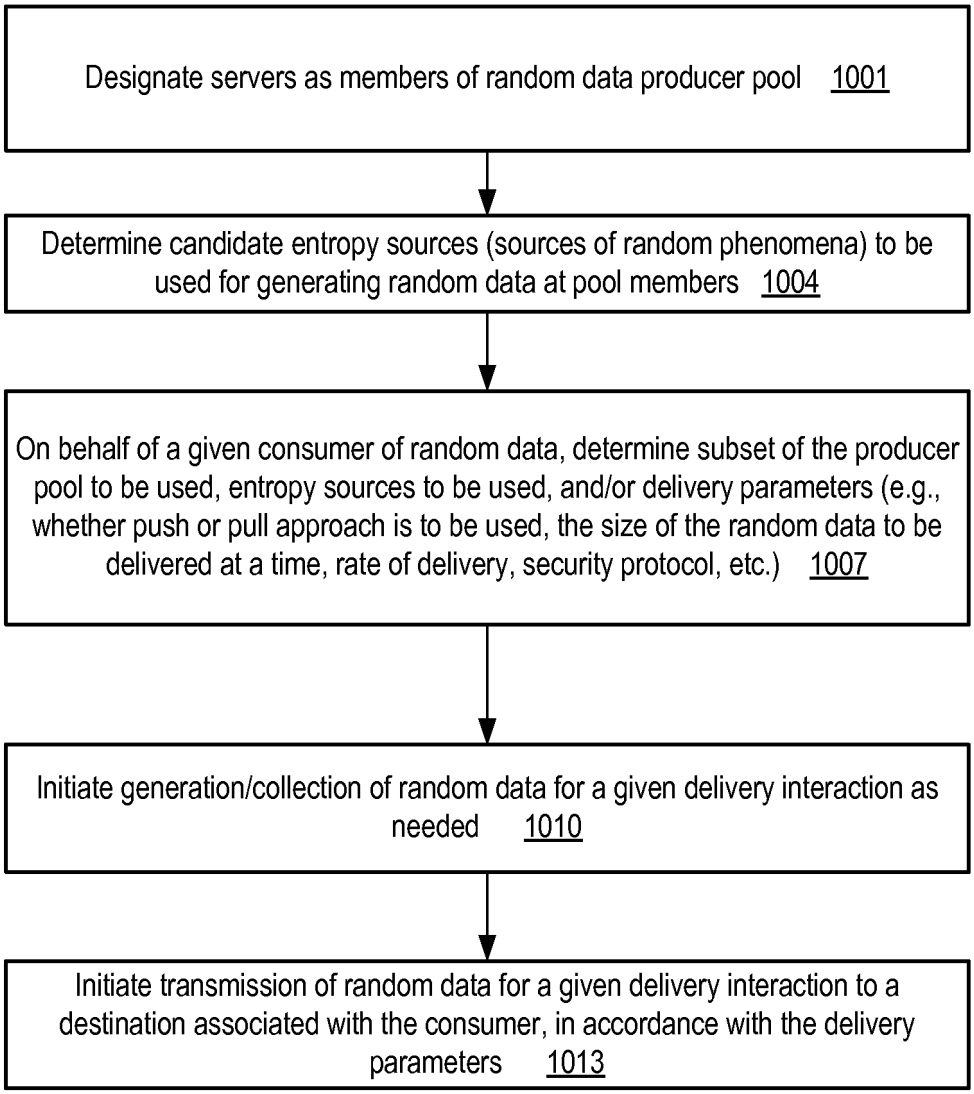
Designate servers as members of random data producer pool 1001
Determine candidate entropy sources (sources of random phenomena) to be used for generating random data at pool members 1004
On behalf of a given consumer of random data, determine subset of the producer pool to be used, entropy sources to be used, and/or delivery parameters (e.g., whether push or pull approach is to be used, the size of the random data to be delivered at a time, rate of delivery, security protocol, etc.) 1007
Initiate generation/collection of random data for a given delivery interaction as needed 1010
Initiate transmission of random data for a given delivery interaction to a destination associated with the consumer, in accordance with the delivery parameters 1013

FIG. 10

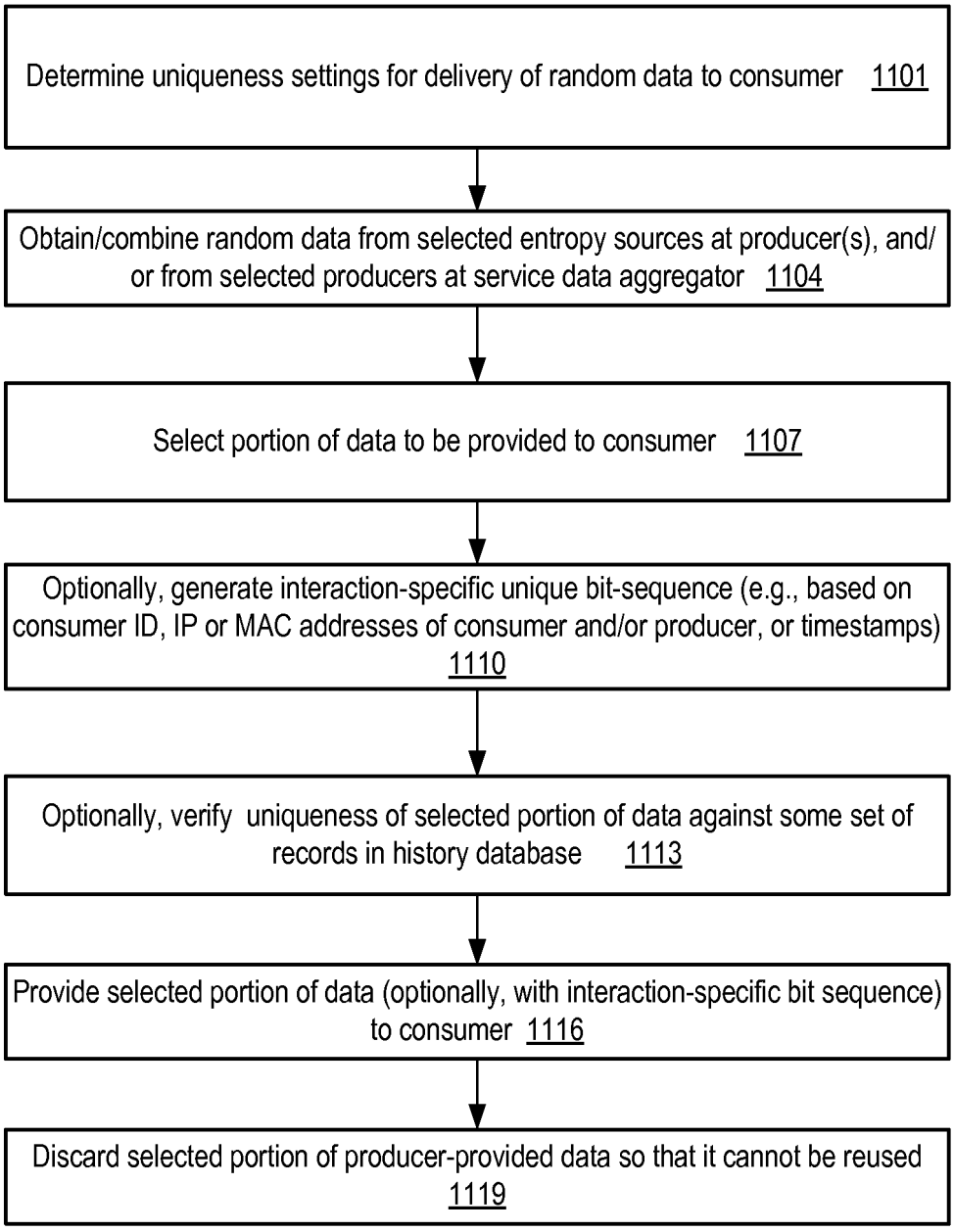
Determine uniqueness settings for delivery of random data to consumer 1101
Obtain/combine random data from selected entropy sources at producer(s), and/or from selected producers at service data aggregator 1104
Select portion of data to be provided to consumer 1107
Optionally, generate interaction-specific unique bit-sequence (e.g., based on consumer ID, IP or MAC addresses of consumer and/or producer, or timestamps) 1110
Optionally, verify uniqueness of selected portion of data against some set of records in history database 1113
Provide selected portion of data (optionally, with interaction-specific bit sequence) to consumer 1116
Discard selected portion of producer-provided data so that it cannot be reused 1119

FIG. 11

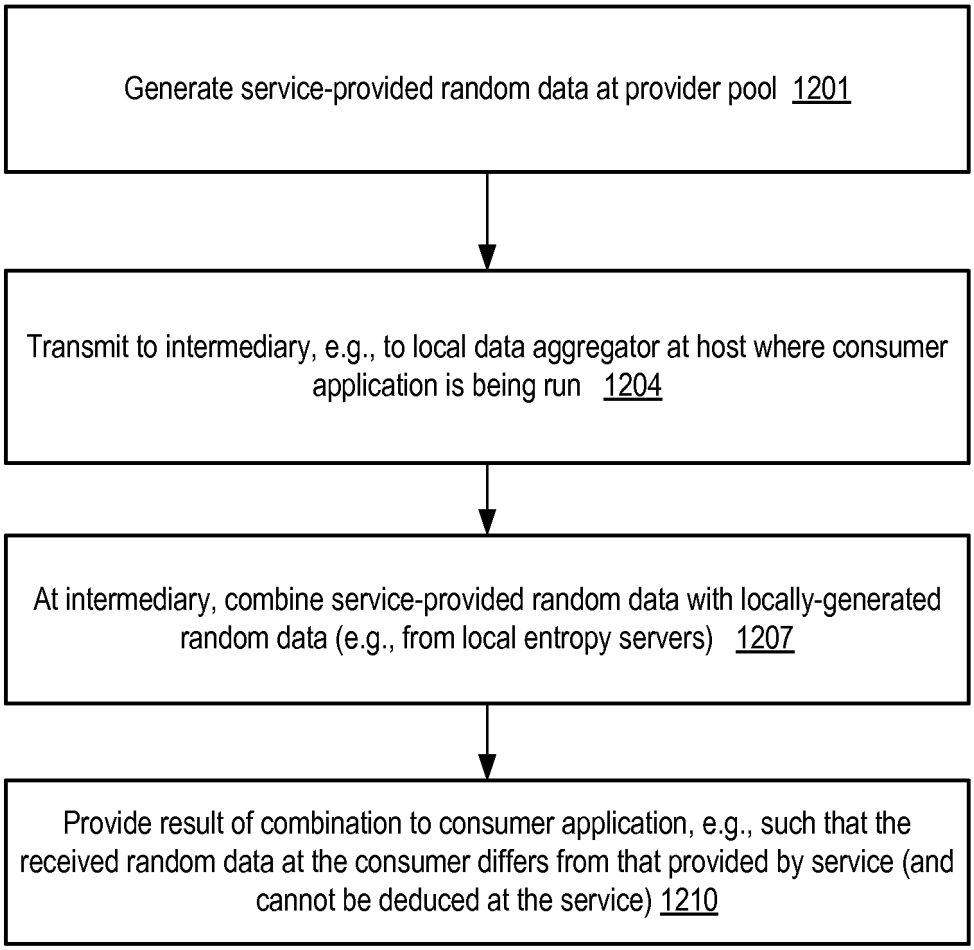
Generate service-provided random data at provider pool 1201
Transmit to intermediary, e.g., to local data aggregator at host where consumer application is being run 1204
At intermediary, combine service-provided random data with locally-generated random data (e.g., from local entropy servers) 1207
Provide result of combination to consumer application, e.g., such that the received random data at the consumer differs from that provided by service (and cannot be deduced at the service) 1210

FIG. 12

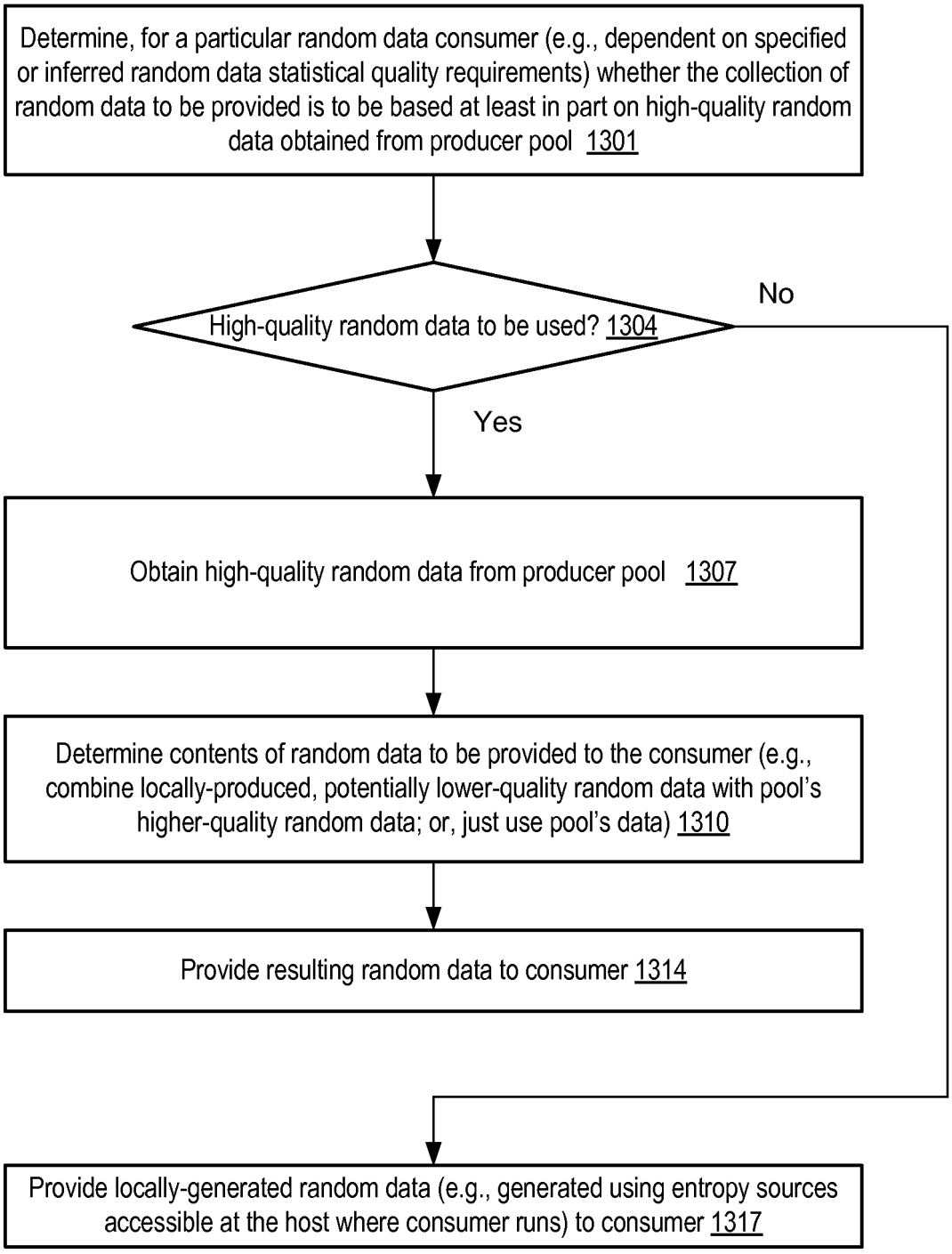
Determine, for a particular random data consumer (e.g., dependent on specified or inferred random data statistical quality requirements) whether the collection of random data to be provided is to be based at least in part on high-quality random data obtained from producer pool 1301
High-quality random data to be used? 1304
No
Yes
Obtain high-quality random data from producer pool 1307
Determine contents of random data to be provided to the consumer (e.g., combine locally-produced, potentially lower-quality random data with pool's higher-quality random data; or, just use pool's data) 1310
Provide resulting random data to consumer 1314
Provide locally-generated random data (e.g., generated using entropy sources accessible at the host where consumer runs) to consumer 1317

FIG. 13

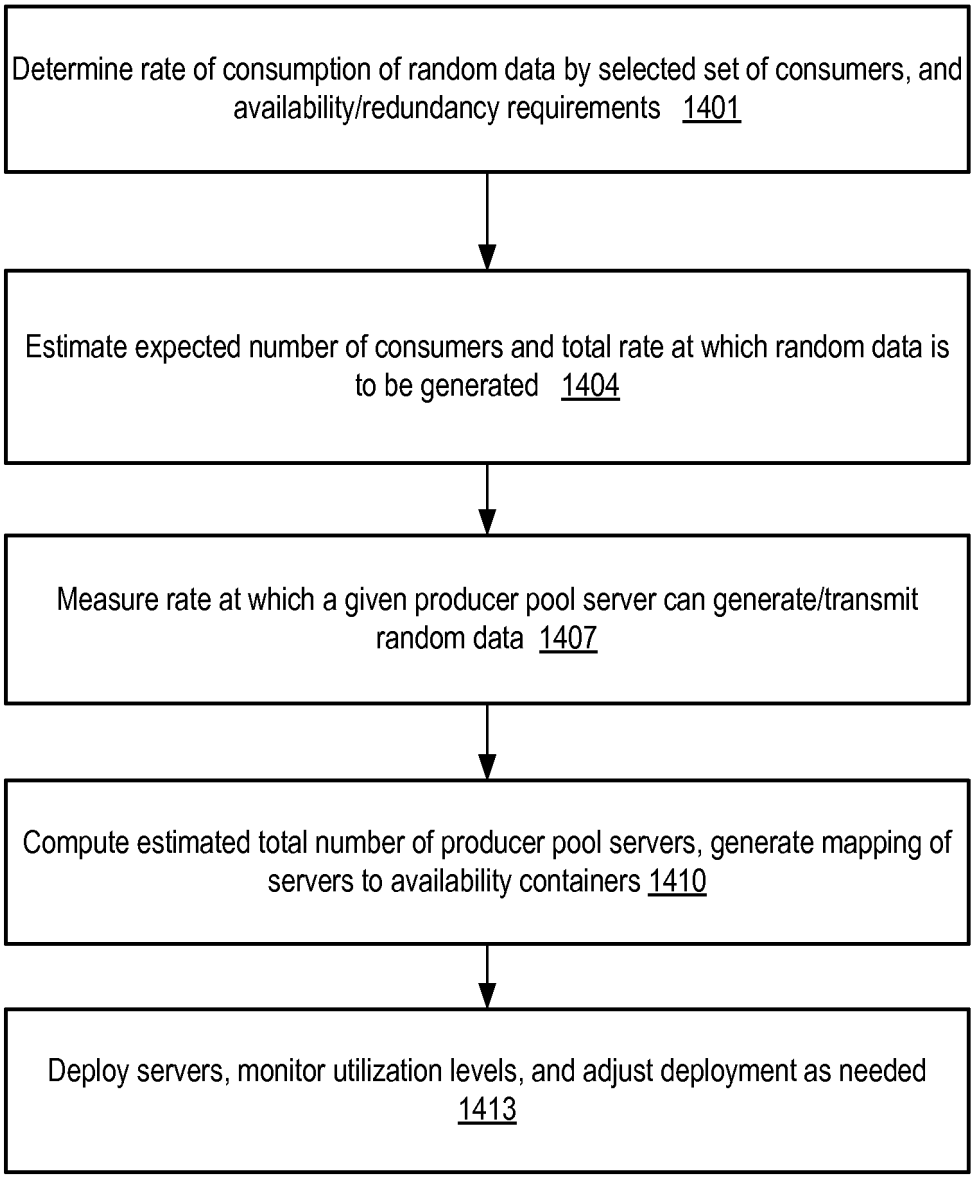
Determine rate of consumption of random data by selected set of consumers, and availability/redundancy requirements 1401
Estimate expected number of consumers and total rate at which random data is to be generated 1404
Measure rate at which a given producer pool server can generate/transmit random data 1407
Compute estimated total number of producer pool servers, generate mapping of servers to availability containers 1410
Deploy servers, monitor utilization levels, and adjust deployment as needed 1413

FIG. 14

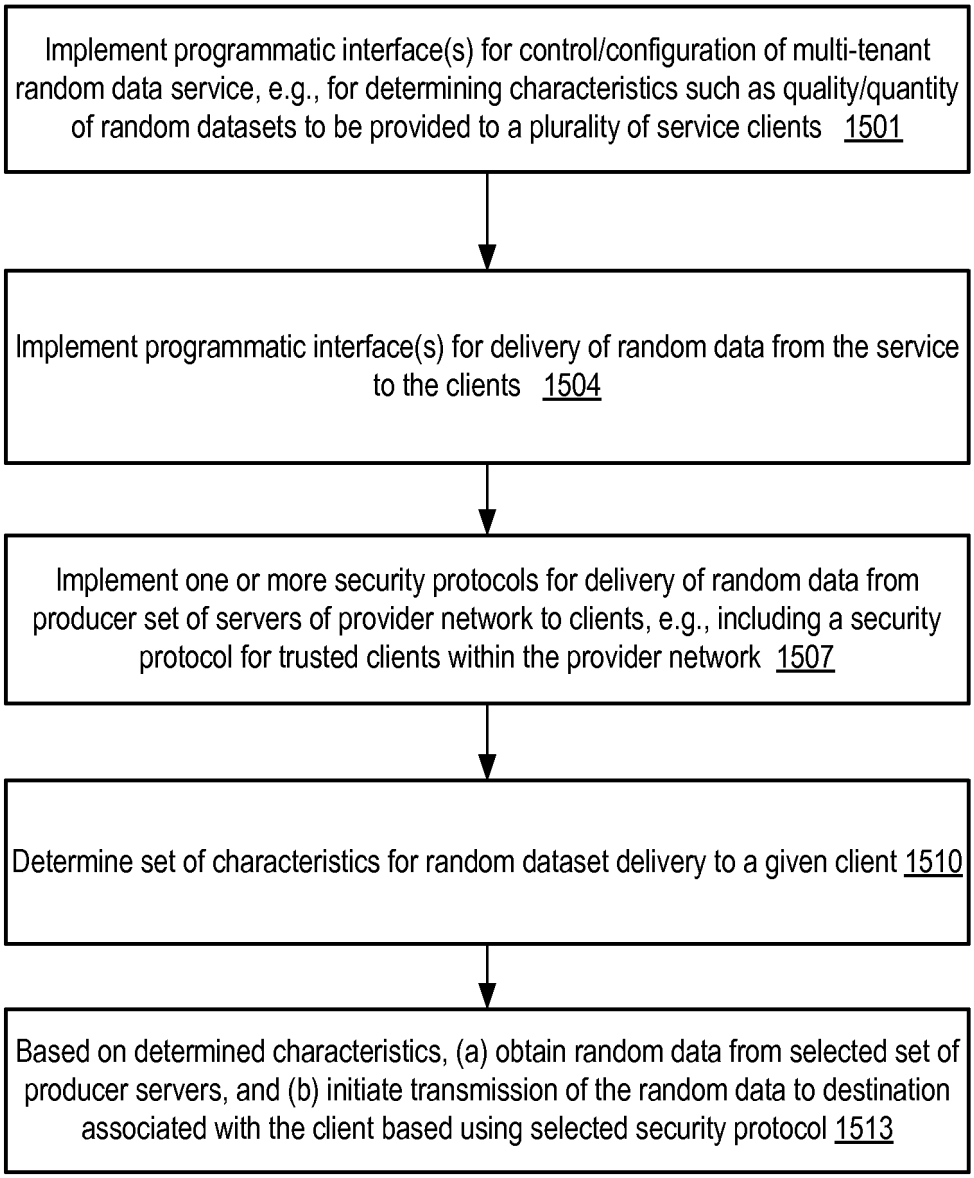
Implement programmatic interface(s) for control/configuration of multi-tenant random data service, e.g., for determining characteristics such as quality/quantity of random datasets to be provided to a plurality of service clients 1501
Implement programmatic interface(s) for delivery of random data from the service to the clients 1504
Implement one or more security protocols for delivery of random data from producer set of servers of provider network to clients, e.g., including a security protocol for trusted clients within the provider network 1507
Determine set of characteristics for random dataset delivery to a given client 1510
Based on determined characteristics, (a) obtain random data from selected set of producer servers, and (b) initiate transmission of the random data to destination associated with the client based using selected security protocol 1513

FIG. 15

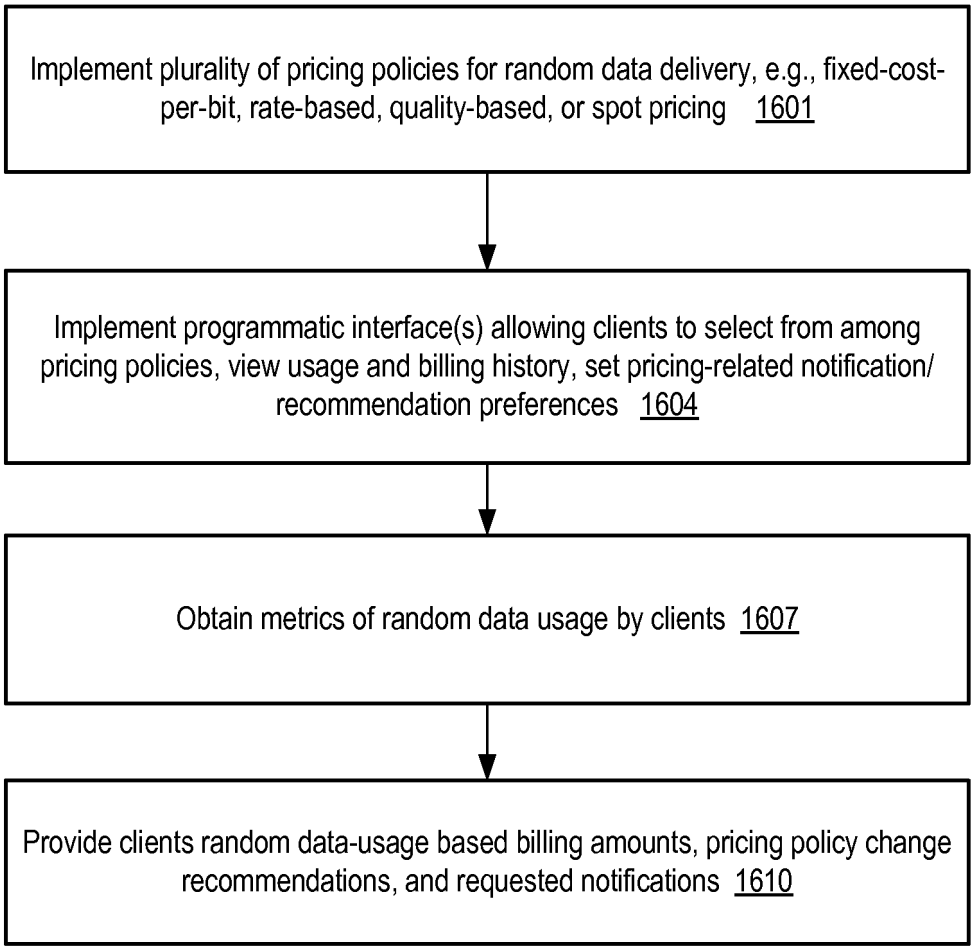
Implement plurality of pricing policies for random data delivery, e.g., fixed-cost-per-bit, rate-based, quality-based, or spot pricing 1601
Implement programmatic interface(s) allowing clients to select from among pricing policies, view usage and billing history, set pricing-related notification/ recommendation preferences 1604
Obtain metrics of random data usage by clients 1607
Provide clients random data-usage based billing amounts, pricing policy change recommendations, and requested notifications 1610

FIG. 16

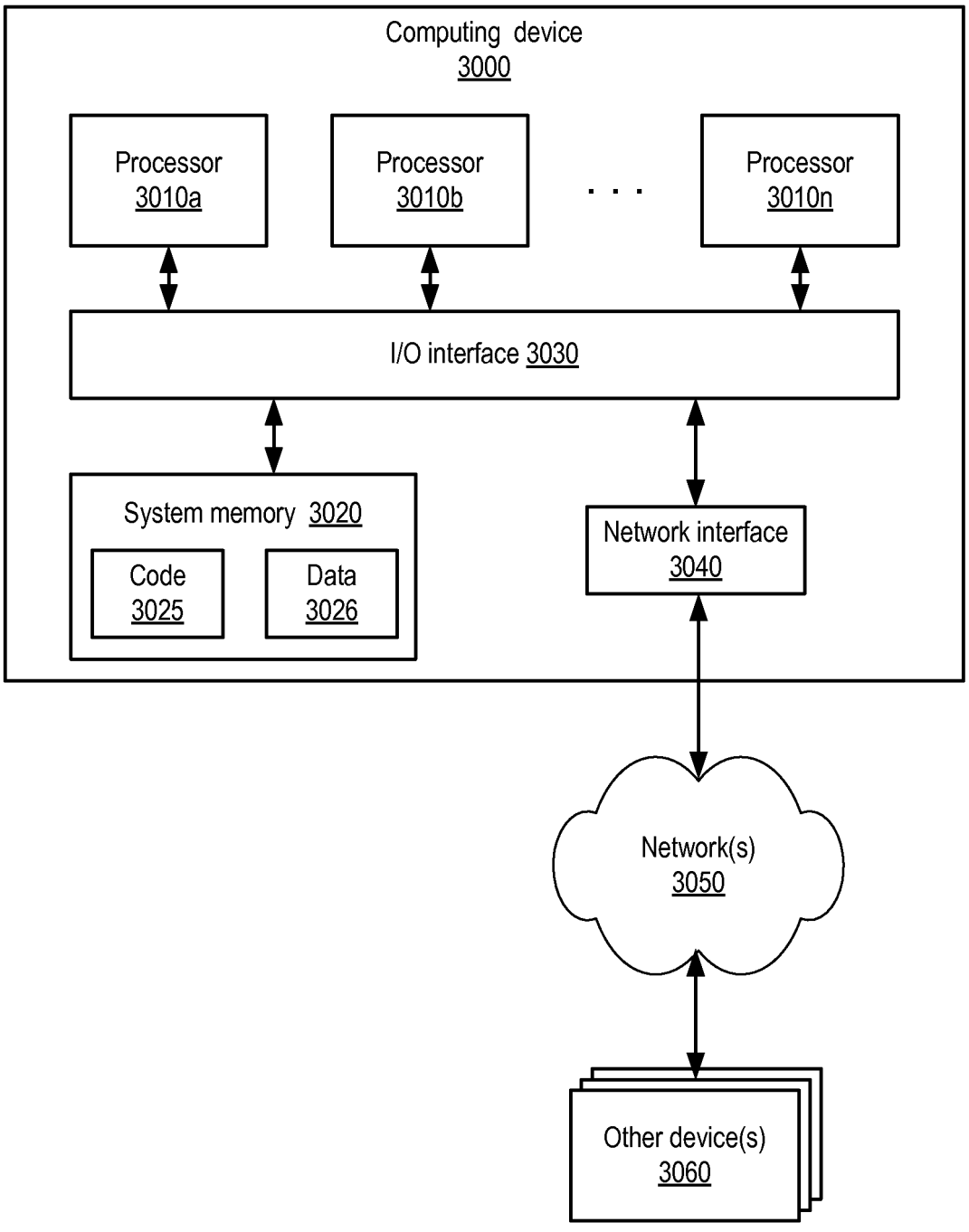
Computing device
3000
Processor
3010a
Processor
3010b
Processor
3010n
I/O interface 3030
System memory 3020
Code
3025
Data
3026
Network interface
3040
Network(s)
3050
Other device(s)
3060

FIG. 17