(51) International Patent Classification:
A63F 3/06 (2006.01)

(21) International Application Number:
PCT/US2012/058204

(22) International Filing Date:
30 September 2012 (30.09.2012)

(25) Filing Language:
English

(26) Publication Language:
English

(30) Priority Data:
61/619,672 30 September 2011 (30.09.2011) US
61/624,516 16 April 2012 (16.04.2012) US
61/675,495 25 July 2012 (25.07.2012) US


(72) Inventors; and

(74) Agent: ZACHARIAS, Peter, K.; Blank Rome LLP, One Logan Square, 130 North 18th Street, Philadelphia, Pennsylvania 19103 (US).


(54) Title: LOTTERY POOLING AND CONSOLIDATION METHOD AND SYSTEM

(57) Abstract: Systems and methods of pooling and consolidating pools of lottery tickets. A method of pooling includes receiving information identifying one or more lottery tickets, receiving a request to pool the one or more lottery tickets, associating the one or more lottery tickets with one or more pools based on the request and predefined pooling rules and parameters for each of the one or more pools, and transmitting an indication of the one or more pools with which the one or more lottery tickets are associated. A method for consolidating a plurality of pools of lottery tickets includes steps of moving one or more tickets associated with an unfilled pool to a consolidation pool, using limited swapping first and unlimited swapping later, as needed, and moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using limited swapping first and unlimited swapping later, as needed.

[Continued on next page]

Published:
— with international search report (Art. 21(3))
LOTTERY POOLING AND CONSOLIDATION METHOD AND SYSTEM

CROSS REFERENCE TO RELATED APPLICATIONS


FIELD OF THE INVENTION

[0002] The present invention relates to systems and methods for pooling and consolidating lottery tickets and, more specifically, to systems and methods for pooling lottery tickets into one or more pools according to a pooling request from a lottery participant and consolidating one or more unfilled pools into other unfilled pools or into filled pools prior to a lottery drawing.

BACKGROUND OF THE INVENTION

[0003] Lottery sales in the United States are big business, generating over $58 billion in sales in 2009 and 2010. Financial records for 41 state lotteries that ended their fiscal years in
June 2011 show that 28 had higher sales than the year before. Seventeen of those states set all-
time sales records. The highest year-over-year increase for a state was about 6%.

[0004] Lottery sales are split between prizes, administrative costs, retailer commissions, and state profits. Approximately 50% to 60% of U.S. lottery sales are paid out as prizes to winners, with administrative costs for advertising, employee salaries, and other operating expenses usually accounting for 1% to 10% of sales. Retailers retain 5% to 8% of sales in the form of commissions and approximately 2% as bonuses for selling winning tickets. The remaining 30% to 40% is profit turned over to the state. As of 2006, a total of $234.1 billion has been given to various beneficiaries since the beginning of lotteries in each state. Lotteries have been an important source of revenues for the states.

[0005] The Multi State Lottery Association (MUSL) is a non-profit, government-benefit association owned and operated by its member lotteries. After its formation in 1987, MUSL’s first multi-state game was Lotto Merica. The game ran four years before being replaced. On April 22, 1992, the first Powerball® drawing was held. The MUSL’s current games are Powerball®, Mega Millions®, Hot Lotto®, Wildcard 2®, 2 by 2®, and Ca$hola™. Each MUSL member offers one or more of the games administered by MUSL.

SUMMARY OF THE INVENTION

[0006] In accordance with an exemplary aspect of the present invention, there is provided a method for pooling lottery tickets. The method includes a step of receiving information identifying one or more lottery tickets, a step of receiving a request to pool the one or more lottery tickets, a step of associating the one or more lottery tickets with one or more pools based on the request, predefined pooling rules, and predefined pooling parameters for each of the one or more pools, and a step of transmitting an indication of the one or more pools with which the one or more lottery tickets are associated.
In accordance with another exemplary aspect of the present invention, there is provided a method for consolidating a plurality of lottery tickets associated with a plurality of pools prior to a lottery drawing. The method includes steps of moving one or more tickets associated with an unfilled pool to a consolidation pool, using limited swapping as needed, moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using limited swapping as needed, moving one or more tickets associated with an unfilled pool to a consolidation pool, using unlimited swapping as needed, and moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using unlimited swapping as needed.

In accordance with yet another exemplary aspect of the present invention, there is provided a method for pooling and consolidating lottery tickets. The method includes steps of receiving information identifying one or more lottery tickets, receiving a request to pool the one or more lottery tickets from a pooling participant, associating the one or more lottery tickets with one or more pools based on the request, predefined pooling rules, and predefined pooling parameters for each of the one or more pools, transmitting an indication of the one or more pools with which the one or more lottery tickets are associated, and prior to a lottery drawing, performing sub-steps of consolidation. The sub-steps of consolidation include moving one or more tickets associated with an unfilled pool to a consolidation pool, using limited swapping as needed, moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using limited swapping as needed, moving one or more tickets associated with an unfilled pool to a consolidation pool, using unlimited swapping as needed, and moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using unlimited swapping as needed.

BRIEF DESCRIPTION OF THE DRAWINGS

For the purpose of illustration, there are shown in the drawings certain embodiments of the present invention. In the drawings, like numerals indicate like elements throughout. It should be understood that the invention is not limited to the precise arrangements, dimensions, and instruments shown. In the drawings:
[0010] FIG. 1A is an exemplary embodiment of a lottery ticket comprising a barcode having information encoded therein, in accordance with an exemplary embodiment of the present invention;

[0011] FIG. IB is another exemplary embodiment of a lottery ticket comprising two barcodes having information encoded therein, in accordance with an exemplary embodiment of the present invention;

[0012] FIG. 2 illustrates an exemplary embodiment of a system for pooling lottery tickets and consolidating pools of lottery tickets, such as those of FIGS. 1A and IB, the system comprising one or more pooling participant computer systems and a pooling coordinator computer system, in accordance with an exemplary embodiment of the present invention;

[0013] FIG. 3 illustrates an exemplary embodiment of a method by which a user requests to open an account in the system of FIG. 2 and requests that one or more lottery tickets, such as those of FIG. 1A or IB, be pooled, in accordance with an exemplary embodiment of the present invention;

[0014] FIG. 4A illustrates an exemplary embodiment of a method by which the pooling coordinator computer system receives the account request of FIG. 3 and creates the account, in accordance with an exemplary embodiment of the present invention;

[0015] FIG. 4B illustrates an exemplary embodiment of a method by which the pooling coordinator computer system receives and optionally validates the one or more lottery tickets transmitted with the pooling request of FIG. 3, in accordance with an exemplary embodiment of the present invention;

[0016] FIG. 4C illustrates an exemplary embodiment of a method by which the pooling coordinator computer system pools the lottery tickets received and optionally validated in FIG. 4B, in accordance with an exemplary embodiment of the present invention;
FIG. 5A illustrates an exemplary embodiment of a method by which the pooling coordinator computer system transmits pooling information, consolidates pools prior to a lottery drawing, and notifies pooling participants of the results of the lottery drawing, in accordance with an exemplary embodiment of the present invention;

FIG. 5B illustrates an exemplary embodiment of a method by which the pooling participant computer system receives pooling information and receives notifications of the results of a lottery drawing, in accordance with an exemplary embodiment of the present invention;

FIGS. 6A and 6B illustrate an exemplary embodiment of a method by which the pooling coordinator computer system effects pooling of a batch of lottery tickets received from a pooling participant, in accordance with an exemplary embodiment of the present invention;

FIGS. 7A and 7B illustrates an exemplary embodiment of a method by which the pooling coordinator computer system effects consolidation of pools of lottery tickets, in accordance with an exemplary embodiment of the present invention;

FIGS. 8A through 8I illustrate an exemplary embodiment of a pooling interface portal, in accordance with an exemplary embodiment of the present invention; and

FIG. 9 illustrates an exemplary alternative embodiment of the ticket of FIG. 1 in which the ticket is only active when a jackpot for a specified lottery is equal to or greater than a predetermined threshold, in accordance with an exemplary embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Reference to the drawings illustrating various views of exemplary embodiments of the present invention is now made. In the drawings and the description of the drawings herein, certain terminology is used for convenience only and is not to be taken as limiting the
embodiments of the present invention. Furthermore, in the drawings and the description below, like numerals indicate like elements throughout.

[0024] Lottery tickets may be pooled for various reasons. One example of a conventional lottery pool may be as informal as one person telling another person that he will split lottery winnings if the other person agrees to do the same. Another example of a conventional lottery pool is an office pool. Participants at a place of employment may contribute money to a pool, and a designated person uses the pooled money to purchase a set of lottery tickets. Winnings are distributed according to rules established by the pool participants. Yet another example of conventional pooling may involve a written agreement among the pool participants, who are obligated to contribute money to the pool at agreed-upon times. In some cases, such a pool submits a Tax Identification Number to the Internal Revenue Service.

[0025] In several of these examples of conventional pools, each participant puts money into the pool to buy tickets together with other participants, so that the tickets belong to the group, and winnings are distributed as agreed upon by the participants. In accordance with exemplary embodiments of the present invention, pooling participants buy tickets separately and then join one or more pools using a computerized pooling system. Each pooler retains physical possession of his tickets. The system does not buy, sell, hold, or take positions on lottery tickets and does not share in lottery winnings.

[0026] Referring now to FIG. 1A, there is illustrated an exemplary embodiment of a lottery ticket, generally designated as 100, used by one or more exemplary embodiments of the present invention. The lottery ticket 100 comprises a text block 110 comprising text 120 identifying the lottery in which the ticket 100 is participating, a ticket number 125 that uniquely identifies the ticket, text 130 identifying the date and time of purchase of the ticket 100, and the lottery numbers 140 of the ticket 100. The ticket further comprises one or more scannable items comprising data encoded therein. The encoded data comprises an identification code that uniquely identifies the ticket 100. In the exemplary embodiment of the lottery ticket 100 illustrated in FIG. 1A, the one or more scannable items comprise one or more bar codes 140. As
described below, it is to be understood that the one or more scannable items are not limited to
being bar codes 140.

[0027] FIG. 1B illustrates an exemplary alternative embodiment of the lottery ticket 100, 
generally designated in FIG. 1B as 100', in accordance with an exemplary embodiment of the 
present invention. The lottery ticket 100' is similar to the lottery ticket 100, but the lottery ticket 
100' is located on a receipt 150 printed at a point of sale. The lottery ticket 100' comprises one 
or more scannable items, e.g., a bar code 140', comprising encoded data that includes some or all 
of the data encoded in the one or more scannable items of the ticket 100 illustrated in FIG. 1A. 
As with the lottery ticket 100, it is to be understood that the one or more scannable items of the 
lottery ticket 100' are not limited to being bar codes 140'.

[0028] The receipt 150 comprises the ticket 100' and one or more further scannable items 
comprising data encoded therein. The encoded data comprises information relating to the sale of 
the ticket 100' contained in the receipt 150. For example, such encoded data may include any of 
the text 120 identifying the lottery in which the ticket 100' is participating, the ticket number 125 
of the ticket 100', the date and time 130 of purchase of the ticket 100', the lottery numbers 140 
of the ticket 100, the location of the point of sale of the ticket 100', and/or a unique identification 
code for the receipt 150. In the exemplary embodiment of the receipt 150 illustrated in FIG. 1B, 
the one or more further scannable items comprise one or more bar codes 160. As described 
below, it is to be understood that the one or more scannable items are not limited to being bar 
codes 160.

[0029] Although FIG. 1B designates the portion of the receipt 150 that is similar to the 
lottery ticket 100 as the lottery ticket 100', it is to be understood that, where context permits 
herein, the entire receipt 150 may also be thought of as a lottery ticket, not just the portion 100'. 
Further, it is to be understood that other exemplary embodiments in which the receipt 150 
comprises more than one lottery ticket 100' are contemplated. In one variation on such other 
exemplary embodiments, each lottery ticket 100' may have its own scannable item 140' on the 
receipt 150, the scannable item 140' encoded with a unique identification code that unique 
identifies the respective lottery ticket 100'. Alternatively, all lottery tickets 100' together may
have one, combined scannable item 140' on the receipt, the scannable item 140' encoded with a unique identification code that uniquely identifies the batch of lottery tickets 100'.

[0030] Referring now to FIG. 2, there is illustrated an exemplary embodiment of a networked computer system, generally designated as 200, for establishing user accounts, receiving requests to pool blocks or batches of tickets from lottery participants, pooling blocks or batches of lottery tickets into one or more pools, consolidating pools as needed prior to a lottery drawing, calculating lottery winnings for each lottery participant, and displaying pooling information and lottery results, in accordance with an exemplary embodiment of the present invention. The system 200 comprises a computer system 210 operated by a pooling coordinator. The reference number, "210," is used herein to refer interchangeably to the computer system operated by the pooling coordinator and to the pooling coordinator itself.

[0031] The computer system 210 is connected to one or more remote computer systems, each of which is used or operated by a pooling participant. In the exemplary embodiment illustrated in FIG. 2, the computer system 210 is connected to a remote mobile computer system 220 and a personal computer system 230 via a network 240. In an exemplary embodiment, the network 240 is the Internet.

[0032] It is to be understood that the mobile computer system 220 and the personal computer system 230 are not limited to being mobile and personal computer systems, respectively, but may be any personal computer system or mobile computer system, such as a laptop computer, a smart phone, a tablet PC, etc., known in the art. The reference numbers, "220" and "230," are used herein to refer interchangeably to the computer systems respectively operated by pooling participants and to the users of such computer systems themselves. Such users are also referred to as "pooling participants" herein.

[0033] A pooling participant uses the computer system 220 or 230 to create an account with the computer system 210, to upload lottery tickets to the computer system 260, and to transmit pooling requests to the computer system 210. The computer system 210 receives the lottery tickets and the pooling requests, pools the lottery tickets into pools specified in the
pooling requests depending on a set of predetermined rules, consolidates the pools, monitors lottery drawings, calculates lottery winnings for each lottery participant, and displays pooling information and lottery results.

[0034] The computer system 210 comprises a computer system 260, e.g., a web server, which hosts web services accessed by the computer systems 220 and 230. In one variation of this exemplary embodiment, the web server 260 provides access to the web services by way of a secure website (hereinafter "pooling interface website"), which the users of the computer systems 220 and 230 access to register with the pooling coordinator 210, upload their lottery tickets, and pool their lottery tickets. In another variation of this exemplary embodiment, the computer systems 220 and 230 access the web services via a specially programmed application (hereinafter "pooling interface application") downloaded from the web server 260. In yet another variation of this exemplary embodiment, the computer systems 220 and 230 access the web services via the pooling interface website and the pooling interface application depending on whether the user is creating an account, uploading lottery tickets, transmitting pooling requests, or monitoring lottery drawings. In any of these variations, the web server 260 may provide the pooling interface application for download via an unsecure webpage. The terms, "pooling interface website" and "pooling interface application" are referred to generally below as a "pooling interface portal."

[0035] The computer system 260 receives a request to create an account with the pooling coordinator from the computer system 220 or the computer system 230 via the pooling interface portal. If the request is a valid request to create an account, the computer system 260 creates and stores the account within a computer readable medium. In an exemplary embodiment, the computer readable medium comprises a database 265. It is to be understood that the computer readable medium may be any storage devices, such as a hard disk array, known in the art.

[0036] The computer system 260 receives uploaded lottery tickets from the computer system 220 or the computer system 230 via the pooling interface portal. The computer system 260 also receives pooling requests from the computer system 220 or the computer system 230 via the pooling interface portal. The computer system 260 performs pooling based on the
requests, predetermined rules, and predetermined parameters. The computer system 260 stores the pooled lottery tickets in one or more pools stored within the database 265.

[0037] FIG. 3 illustrates an exemplary method, generally designated as 300, of requesting an account, uploading one or more lottery tickets, and requesting that uploaded lottery tickets be entered into one or more pool(s), in accordance with an exemplary embodiment of the present invention. The steps of the method 300 are performed on the pooling participant 220 or 230 side of the system 200. As described below, certain steps are performed by the pooling participant, and certain steps are performed by the computer system 220 or 230 operated by the pooling participant.

[0038] Each pooling participant's computer system 220, 230, etc. is programmed with software instructions stored in a tangible computer-readable medium (e.g., a hard disk or a solid state memory), which instructions, when executed by the computer system 220 or 230, cause the computer system to perform the relevant steps of the method 300. In the exemplary embodiment in which the computer systems 220 and 230 access the computer system 260 by way of a web site, the steps of the method 300 are performed in a web application operating in the website accessed by a web browser executed by the computer systems 220 and 230. In the exemplary embodiment in which the computer systems 220 and 230 access the computer system 210 by way of a pooling interface application, the steps of the method 300 are performed by the application, which is executed in the computer systems 220 and 230. For ease of discussion, description of the method 300 is made with reference to the pooling participant using the computer system 220. It is to be understood that the steps of the method 300 may be performed by any pooling participant and such pooling participant's computer system, e.g., the computer system 230.

[0039] FIG. 4A illustrates exemplary methods, generally designated as 400A, 400B, and 400C, performed on the pooling coordinator side of the system 200, specifically by the computer system 260, in accordance with an exemplary embodiment of the present invention. The method 400A is a method by which the computer system 260 receives a request from a user to create an account with the pooling coordinator and creates the account. The method 400B is a method by
which the computer system 260 receives uploaded lottery tickets from users and validates the lottery tickets. The method 400C is a method by which the computer system 260 receives requests to pool the lottery tickets received in the method 400B and pools the lottery tickets. Together, the methods 300 and 400A, 400B, and 400C constitute a method of pooling lottery tickets provided by a user, for example, the user 220.

[0040] The computer system 260 is programmed with software instructions stored in a tangible computer-readable medium (e.g., a hard disk or a solid state memory), which instructions, when executed by the computer system 260, cause the computer system to perform the relevant steps of the methods 400 through 700. Thus, by executing relevant software instructions, the computer system 260 receives account requests and processes them (method 400A), receives uploaded lottery tickets and (optionally) validates them (method 400B), receives and processes pooling requests (methods 400B and 600), pools tickets per received pooling requests (method 400C), consolidates pools (method 500A and 700), and provides results of lottery drawings (500B).

[0041] Referring now to FIGS. 3, 4A, 4B, and 4C together, the method 300 begins with the user 220 requesting an account with the pooling coordinator 210, Step 310. It is contemplated that the user 220 may request the account via either the pooling interface website, via a secure account creation website, or via the pooling interface application. In the exemplary embodiment in which the user uses the pooling interface application to request the account, the user 220 may have downloaded the pooling interface application prior to the Step 310 or during the Step 310 from a website that may be part of the pooling interface website or another unsecure website hosted by the computer system 260.

[0042] In the Step 310, the user 220 inputs relevant account information (name, user name, address, phone number, credit card information, etc.) into either the pooling interface portal or a secure account creation website. The user 220 directs the pooling interface portal or the secure account creation website to issue a request to the computer system 260 to create a new user account for the user 220. The computer system 220 transmits this information with the IP address of the computer system 220 over the Internet 240 to the computer system 260 as a
request to create a new account on the basis of the transmitted information, Step 310. The method 300 continues to a break A and waits for the computer system 260 to respond. The method of pooling continues to the method 400A via break A.

[0043] The computer system 210 receives the request, Step 410A. The computer system 210 then validates the request by searching the database 265 to determine whether the user 220 already has an account with the pooling participant 210, Step 420A. If the user 220 already has an account within the database 265, i.e., if the request is not valid, Step 430A, the method 400A continues to a Step 440A in which the computer system 260 transmits a refusal to create a new account to the computer system 220. The method 400A returns to the method 300 via break A. If the user 220 does not already have an account within the database 265, i.e., if the request is valid, Step 430A, the method 400A continues to a Step 450A in which the computer system 260 creates a new account for the user 220 and adds the information received with the request to the database 265. After account creation, the computer system 260 transmits an acknowledgement to the computer system 220, Step 460A. The method 400A returns to the method 300 via the break A.

[0044] After returning from break A, the method 300 continues to a Step 315 in which the computer system 220 determines whether the account request in the Step 310 was successful in creating a new account. If not, the method 300 terminates, Step 390. If it was, the computer system 220 receives confirmation that the account was created and relevant information concerning the account, e.g., confirmation of user name, Step 320. In embodiments in which the pooling interface portal includes the pooling interface application, the user 220 may be given the option of downloading the pooling interface application in the Step 320 if the user 220 used the account creation website or the pooling interface website to request account creation in the Step 310. Processing continues to a Step 330.

[0045] It is to be understood that every execution of the method 300 does not necessarily involve performance of the Steps 310, 315, and 320. In fact, desirably, such steps are performed only once, when the user 220 directs the computer system 200 to perform the method 300 for the first time to create a new account. Thereafter, during subsequent iterations of the method 300,
the computer system 220 performs only the remaining steps of the method 300 of pooling, as the
user 220 will have already had an account created in a prior execution of the method 300. Thus,
after the account of the user 220 is created, each normal, subsequent performance of the method
300 begins with the Step 330.

[0046] In the Step 330, the user 220 purchases one or more lottery tickets 100 from a vendor. The user 220 then accesses the pooling interface portal and logs into the system 260, Step 340. Using the computer system 220, the user 220 scans the purchased one or more lottery tickets 100 and uploads the one or more lottery tickets 100 in batches, Step 350. In an exemplary embodiment, the user 220 captures an image of the one or more lottery tickets and uploads the image in the Step 350 via the pooling interface portal. Specifically, the computer system 220 transmits the image of the one or more lottery tickets over the network 240 to the computer system 260 in the Step 350. The method 300 breaks at break B and the pooling method continues via the method 400B.

[0047] In the method 400B, the computer system 260 receives the image of the one or more lottery tickets 100, Step 410B. The one or more lottery tickets 100 are thereby "uploaded" to the computer system 260.

[0048] Before storing the uploaded one or more lottery tickets 100 in the data base 265, the computer system 260 decodes the data encoded in the one or more scannable items of the one or more lottery tickets 100 captured in the image. Because the one or more lottery tickets 100 are uploaded as an image, the computer system 260 decodes the image to ascertain the identification code encoded in the one or more scannable items. The computer system 260 transmits the identification code to the computer system 250 operated by the lottery operator for validation, Step 420B, as part of a validation request. Examples of lottery operators include Scientific Gaming, Gtech, the MUSL, or a state lottery. It is contemplated that, in other exemplary embodiments, the Step 420B and subsequent steps relating to validation are optional.

[0049] The lottery operator 250 receives the identification code and validates the one or more lottery tickets 100 by validating the identification code. Validation checks the
identification code against the lottery operator 250's records of sales. Because the one or more scannable items, e.g., bar code 140, of each of the one or more lottery tickets 100 are encoded with information, such as the identification code, unique to the one or more lottery tickets 100, the corresponding state lottery knows exactly which distributor handled each ticket 100 and the time, date, and location of sale of the each ticket. If the identification code exists within the lottery operator 250's records of existing lottery tickets, the lottery operator 250 determines that the identification code is valid. Otherwise, it determines that it is not.

[0050] The lottery operator 250 replies to the validation request with an indication of whether the one or more lottery tickets 100 are validated and with relevant ticket information, e.g., the lottery 120, ticket number 125, drawing date, purchase location, purchase timestamp 130, etc. The computer system 260 receives the ticket information and the validation results, Step 430B. If the one or more lottery tickets 100 were validated, Step 440B, the method 400B continues to a Step 450B, in which the computer system 460 logs the ticket information and successful validation in the database 265 in a new record for the received one or more lottery tickets 100. In the Step 450B, the computer system 260 also transmits a notification to the computer system 220 that the one or more tickets 100 were successfully validated.

[0051] If the one or more lottery tickets were not validated, Step 440B, the method 400B continues to a Step 460B, in which the computer system 260 logs the decoded data and failed validation in the database 265 and transmits a notification to the computer system 220 that the one or more tickets 100 failed validation. The computer system 260 also transmits pooling options to the computer system 220 in the Step 450B. The computer system 260 terminates the pooling process at the computer system 210's side. After the Step 450B or 460B, the method 400B ends via break B, by which the pooling process continues back to the method 300.

[0052] In one exemplary embodiment of the method 400B, if one portion of the tickets 100 were successfully validated and the other portion of the tickets 100 were not, the method 400B splits after the Step 440B. The computer system 260 logs the ticket information and partial validation success in the database 265 in the Step 450B and the decoded data and partial validation failure in the Step 460B. Partial validation success and failure are transmitted in Steps
450B and 460B, respectively, and the method 400B ends via break B, by which the pooling process continues back to the method 300 for continued processing with respect to the successfully validated lottery tickets 100.

[0053] In another exemplary embodiment, the partial failure in the Step 440B causes the computer system 460 to terminate the pooling process at its side and the computer system 220 to terminate the pooling process at its side, even though some lottery tickets 100 were successfully validated. The user 220 may then have the option of resubmitting all lottery tickets 100 for validation. In some scenarios, the user 220’s account may be flagged as possibly fraudulent or engaged in fraudulent activities for further attention by the pooling coordinator 210.

[0054] After returning to the method 300 from the break B, the computer system 220 receives the results of the validation transmitted in the Steps 450B and/or 460B and the pooling options transmitted in the Step 450B, Step 360. If all of the one or more lottery tickets 100 were validated, Step 365, the method 300 proceeds to a Step 370. If all of the one or more lottery tickets 100 failed validation, Step 365, the method 300 terminates in the Step 390. If some of the one or more tickets 100 were validated, Step 365, then if the method 300 is of the exemplary embodiment in which processing may continue, the method 300 proceeds to the Step 370 for pooling the validated tickets 100. Otherwise, if the method 300 is of the exemplary embodiment in which all tickets 100 must pass validation in order to proceed, the method 300 terminates in the Step 390. The user 220 may attempt to revalidate the one or more lottery tickets 100 by performing the steps of the method 300 again, if the computer system 260 so permits.

[0055] After passing the Step 365, the user 220 may use the pooling interface portal to request inclusion of the validated tickets 100 into a pool or a pool type selected by the user 220 from the options provided by the computer system 260, Step 370. The user 220 uses the pooling interface portal to request how the user 220 desires the validated tickets 100 to be pooled in the Step 370 by selecting a pool or a type of pool into which the tickets 100 are to be pooled. The computer system 220 transmits the pooling request to the computer system 460 in the Step 370. The method 300 breaks at break C and the pooling method continues via the method 400C.
The computer system 260 receives the pooling request transmitted by the computer system 220 in the Step 370. The computer system 260 then pools the one or more validated lottery tickets 100 included in the request into a pool or type of pool requested by the user 220 subject to pooling rules and pooling parameters followed by the computer system 260. The computer system 260 adds the validated lottery ticket 100 to a pool in the database 265 depending on the user 220’s request, modified as required by the pooling rules and pooling parameters, Step 420C, and transmits a message to the computer system 220 indicating whether the pooling request was fulfilled as requested or whether it was modified in any way to satisfy the pooling rules, Step 430C. The method 400C returns to the method 300 via the break C. The computer system 220 receives the pooling message, Step 380, and displays the result of the user 220's pooling request. The method 300 is thereby complete, Step 390.

An example of a record in the database 265 for an uploaded lottery ticket 100 is illustrated below in Table 1. The record is created by the computer system 260 in the Step 410B. The left column of the table indicates the fields of the record. The right column is an explanation of such fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inc</td>
<td>Auto incrementer, starting at zero for all pools.</td>
</tr>
<tr>
<td>Unas Dig Sig</td>
<td>The digital signature from the Unassigned Tickets record that contains the 5 tickets are now assigning. This maps the assigned ticket back to the unassigned ticket, which in turn maps all the way back to the original scan by the smartphone user.</td>
</tr>
<tr>
<td>User</td>
<td>Username of the pooling participant.</td>
</tr>
<tr>
<td>Tix</td>
<td>The ticket numbers of the five tickets to be assigned.</td>
</tr>
<tr>
<td>Pool ID</td>
<td>ID of the pool these tickets are being assigned to.</td>
</tr>
<tr>
<td>Date</td>
<td>The drawing date for the pool.</td>
</tr>
<tr>
<td>TS</td>
<td>The current server timestamp.</td>
</tr>
<tr>
<td>PDig Sig</td>
<td>The digital signature of the record in this pool that precedes this record. If this is the first record (INC=zero), then this is null. Otherwise, this record is mapped to the previous record. For record 1, the Dig Sig for record 0 goes here. For record 2, the Dig Sig for record 1 goes here, and so on. This allows the record to be hashed, incorporating the previous record. This results in a chaining of records, so that the integrity of each record is linked to the previous record. Changing any single record results in a broken chain and a red flag. Conversely, matching chained records proves database integrity.</td>
</tr>
</tbody>
</table>
Table 1

[0058] The record includes a filed, "Pool ID," that indicates how the lottery ticket 100 has been pooled. The lottery ticket 100 is added to a pool when the computer system 260 sets the Pool ID field to the ID of a pool in the Step 420C.

[0059] Table 1 indicates exemplary fields of a record of a lottery ticket 100 stored in the database 265. It is to be understood that the record is not limited to including these fields, and it is to be understood that the record may contain other fields, such as "Record Hash," "Dig Sig," "Consolidation Pool ID," "Consolidation Timestamp," "Consolidation Record Hash," and "Consolidation Dig Sig." The field, "Consolidation Pool ID," indicates how the lottery ticket 100 has been consolidated. The lottery ticket 100 is consolidated to a pool when the computer system 260 sets the Consolidation Pool ID field to the ID of a pool in the Step 520A (described below).

[0060] FIG. 5A illustrates an exemplary method, generally designated as 500A, of transmitting pooling information, consolidating pools, and notifying pooling participants of lottery results, in accordance with an exemplary embodiment of the present invention. FIG. 5B illustrates an exemplary method, generally designated as 500B, of receiving pooling information and results of a lottery drawing, in accordance with an exemplary embodiment of the present invention. The steps of the method 500A are performed by the computer system 260. The steps of the method 500B are performed by the computer system 210 or 220.

[0061] Referring now to FIGS. 5A and 5B together, periodically, the computer system 260 transmits information concerning the current statuses of pools in the pooling coordinator computer system 210, Step 510A. The computer system 220 receives such information, Step 510B. In one exemplary embodiment, such information is tailored on a user-by-user basis. Thus, the computer system 260 transmits information to the computer system 220 concerning the pools in which the user 220 is participating. The computer system 220 receives such information and displays it, Step 510B. In another exemplary embodiment, such information concerns all pools currently maintained by the computer system 260 for a particular lottery. Thus, the computer system 260 transmits information to the computer system 220 concerning all pools for
a lottery in which the user 220 is currently participating. In still another exemplary embodiment, the computer system 260 transmits information concerning all pools maintained by the computer system 260 for all lotteries. The computer system 260 repeats the Step 510A on a periodic basis, and the computer system 220 repeats the Step 510B to receive each update transmitted by the computer system in the Step 510A on a periodic basis.

[0062] Prior to the time of lottery drawing, the computer system 260 stops accepting new pooling requests and transmits a final update of pooling information in the Step 510A. The computer system 220 receives such information, Step 510B. The method 500A proceeds to a Step 520A in which the computer system 260 consolidates the lottery pools in the database 265 associated with the lottery drawing.

[0063] At the time of lottery drawing, the computer system 260 communicates with the computer system 250 of the lottery operator and receives the results of the lottery, Step 530A. The computer system 260 calculates the lottery winnings for each participant of a pool containing a winning ticket based on the pools stored in the database 265, as consolidated in the Step 520A, and transmits notifications of the results to all participants, Step 540A. The computer system 220 and all other pooling participants receive the results from the computer system 260, Step 520B. Winnings include matching all numbers in a lottery ticket or any winning combination of numbers. In the Step 540A, the computer system 260 also notifies the lottery operator 250 of the participants winning the drawing. The method 500A is thereby complete, Step 550A, as is the method 500B, Step 530B. In an exemplary embodiment, the computer system 260 stops accepting pooling requests in the Step 510A 15 minutes prior to the drawing in the Step 530A. This gives the computer 260 time to perform consolidation in the Step 520A prior to the drawing in the Step 530A.

[0064] By logging uploaded and pooled tickets in the database 465, by maintaining pooling participants' accounts in the database 265, and by linking the users' accounts with their uploaded and pooled tickets, the computer system 260 is able to tie winning tickets to their purchasers. Because each registered user has his name, address, phone number, email address, credit card information, and/or smart phone information in the computer system 260, there is
certainty that a specific lottery ticket in the database 265 corresponds to a particular, identifiable pooling participant. If a pooling participant who holds a winning ticket attempts to claim his ticket was never part of a winning pool or attempts to give his winning ticket to a friend or family member so as to avoid sharing in winnings, the system 210 is able to prove that the winning ticket was, in fact, purchased at the time, date, and location stored in the database 265 and that such ticket was uploaded to the computer system 260. The uploaded image provides further proof.

[0065] The pooling coordinator 210 does not share in lottery winnings. Instead, in an exemplary embodiment, in the Step 450B, the pooling coordinator 210 charges the user 220 a fee for each validated, registered ticket with an optional minimum charge. In an exemplary embodiment, the pooling coordinator 210 charges $0.20 per validated, registered ticket. In a variation of this exemplary embodiment, the pooling coordinator 210 further imposes a $1.00 minimum charge, meaning that the user 220 is required to upload tickets and request validation of at least 5 tickets. In yet another exemplary embodiment, the user 220 has the option of choosing a monthly subscription, e.g., $5 per month to upload and validate an unlimited number of tickets to any type of pool. In still another exemplary embodiment, the system 210 provides for non-pooling but registered users to upload tickets to track winnings using the system 210 without pooling. This may be provided by the system 210 as a free service. In such an embodiment, the Steps 360-380 are skipped, but the Step 520B is still available.

[0066] In an exemplary variation on the embodiment of the method 400 illustrated in FIGS. 4A-4C, the computer system 260 pools the one or more validated lottery tickets 100 into a pool stored in the database 265 using cipher-block chaining in the Step 420C. As each block of one or more validated lottery tickets 100 is pooled, the data stored in the database 265 for that block is also based upon a previous block of one or more validated lottery tickets 100 pooled and stored in the database 265. Data in the previous block is incorporated into the new data block, and so on, thereby creating a chain of dependency. In such an embodiment, prior to consolidation in the Step 520A, the computer system 260 validates the data in the database 265.
Any irregularities are dealt with by flagging suspect data in the database 265 for further review by the pooling coordinator 210.

[0067] Referring now to FIGS. 6A and 6B together, there is illustrated an exemplary method, generally designated as 600, of receiving a pooling request and effecting pooling, in accordance with an exemplary embodiment of the present invention. Pooling is desirably executed in blocks of five tickets. It is to be understood that pooling is not limited to blocks of five tickets. Other exemplary embodiments in which pooling is effected on a ticket-by-ticket basis, in blocks of two, three, four, six, or any other number of tickets are contemplated.

[0068] Pooling according to the method 600 is guided by a few general predefined pooling rules: (1) One batch of tickets is added at a time; (2) A pooling participant cannot have more than one batch of tickets in any pool; (3) A ticket cannot be added to a pool that is filled to its maximum capacity; and (4) a predetermined number, N, of pools are to be filled to a consolidation percentage for consolidation purposes. An oldest pool is the pool that was created at the earliest time compared to all other pools. A newest pool is the pool that was created at the most recent time compared to all other pools.

[0069] There is no limit to the number of pools in which the user 220 may participate in by submitting blocks or batches of tickets. A pool is filled once it reaches a pool minimum fill percentage. A pool maximum is the number of tickets that would fill a pool to a predetermined capacity. The pool maximum may be expressed as a percentage of a predetermined capacity of a pool. A block or batch is a set of tickets.

[0070] A consolidation pool is a pool that is intentionally left not completely filled during the pooling so that it can be used to store extra tickets for during pool consolidation. A consolidation maximum is the number of tickets that would fill a consolidation pool to a predetermined capacity. The consolidation maximum may be expressed as a percentage of the predetermined capacity of a consolidation pool. The number, N, is a predetermined maximum number of consolidation pools.
[0071] A regular pool is a pool that is not a consolidation pool. An active pool is a pool that has at least one batch of lottery tickets in it. An unfilled pool is an active pool that is filled to less than the pool maximum. An overflow maximum is the number of tickets over pool maximum by which a pool may be overfilled. The overflow maximum may also be expressed as a percentage over pool maximum. The pooling coordinator 210 reserves the right to fill pools to the pool maximum plus the overflow maximum.

[0072] The pool maximum, the consolidation maximum, and the overflow maximum may be expressed as percentages. Exemplary values for the pool maximum (percentage) and consolidation (maximum) percentage are 100% and 85%, respectively. An exemplary value for the overflow (maximum) percentage is 10%. For example, an overflow maximum of 10% on a 500-ticket pool means 50 extra tickets can be added to the pool during the consolidation process, if necessary. An exemplary value for a size of a block or batch is 5. It is to be understood that the size of blocks or batches is not limited to five tickets. Other exemplary embodiments in which blocks or batches comprise one, two, three, four, six, or any other number of tickets are contemplated.

[0073] The method 600 begins with the computer system 260 receiving a request for pooling comprising a batch of lottery tickets and a request to place them into a type of pool selected by the pooling participant, Step 610. The computer system 260 determines whether an active pool of the type of pool specified in the request exists, Step 620. If an active pool of the type does not exist, the method 600 proceeds to a Step 625 in which the computer system 260 creates a new pool having the specified type in the database 265 and places the batch of tickets into the pool in the database 265, Step 625. The computer system 260 then determines whether the number of consolidation pools in the database 265 is less than N, Step 627. If there are fewer than N consolidation pools in the database 265, the computer system 260 marks the new pool as a consolidation pool, Step 629, and the method 600 ends, Step 690. Otherwise, if there are N or greater than N consolidation pools in the database 265, the method 600 proceeds from the Step 627 to the Step 690 and ends.
Referring again to the Step 620, if an active pool of the specified type does exist for this lottery, the method 600 proceeds from the Step 620 to a Step 630 in when the computer system 260 finds and selects the oldest active pool in the database 265. The computer system 260 then determines whether the selected oldest active pool is filled to the pool maximum, Step 640. If the selected oldest active pool is not filled to the pool maximum, the method 600 proceeds to a Step 645. Otherwise, it proceeds to a Step 650.

In the Step 645, the computer system 260 determines whether the pooling participant already has a batch of lottery tickets in the selected oldest active pool in the database 265. If the pooling participant does, the method 600 proceeds to the Step 650. Otherwise, it proceeds to a Step 660.

In the Step 660, the computer system 260 determines whether the selected oldest active pool is a consolidation pool. If it is, the method proceeds to a Step 665. Otherwise, it proceeds to a Step 675 in FIG. 6B via D in FIG. 6A. In the Step 665, the computer system 260 determines whether the selected oldest active pool has reached the consolidation percentage. If it has, the method 600 proceeds to the Step 650. Otherwise, the method 600 proceeds to a Step 670, and the computer system 260 adds the batch of lottery tickets to the selected oldest active pool. The method 600 then ends, Step 690.

As described, the method 600 may proceed to the Step 650 directly from the Steps 640, 645, and 665, depending on whether the conditions in those steps are met. In the Step 650, the computer system 260 determines whether any other active pools exist. If such a pool exists, the method 600 proceeds to a Step 655 in which the computer system 260 finds and selects the next oldest active pool in the database 265. Processing then loops back to the Step 640 for further processing of the selected next oldest active pool. In the Step 650, if there is no other active pool, the method 600 continues to the Step 625 and creates a new pool, as described above with reference to Step 625. Processing proceeds from the Step 625 as described above.

As described above, if the computer system 260 determines in the Step 660 that the oldest selected active pool is not a consolidation pool, then the method 600 continues to a
Step 675 in FIG. 6B. In the Step 675, the computer system 260 adds the received batch of lottery
tickets to the selected oldest active pool. The computer system 260 then determines whether the
selected oldest active pool has reached the consolidation percentage with the addition of the
batch of lottery tickets, Step 680. If it has, the computer system 600 marks the selected oldest
active pool as a consolidation pool, Step 683. Otherwise, the method 600 returns to FIG. 6A via
D and ends, Step 690.

[0079] From the Step 683, the method 600 moves to a Step 685 in which the computer
system 260 determines whether the number of consolidation pools is greater than N. If it is, then
the computer system 260 unmarks the oldest consolidation pool, i.e., sets the oldest consolidation
pool as a non-consolidation pool, and returns to FIG. 6A via D and ends, Step 690. If it is not,
then the computer system 260 returns to FIG. 6A from the Step 685 via D and ends, Step 690.

[0080] In the exemplary embodiment of the method 600 illustrated and described, tickets
go into the active pool in the Step 675 that was changed in a previous iteration of the method 600
in the Step 687 from a consolidation to a non-consolidation pool. This is the case because the
pool that was changed in the Step 687 from a consolidation to a non-consolidation pool is now
the oldest regular pool that is not filled to the pool maximum. Thus, the method 600 provides
that the consolidation pools will be the last N active pools, if the number of active pools equals
or exceeds N. Otherwise, all pools will be consolidation pools.

[0081] As described above, in an exemplary embodiment, the pooling performed in the
Step 420C is performed in accordance with the method 600. In such exemplary embodiment,
batches of tickets uploaded by a user may be pooled anytime, as determined by the user 220.
Other exemplary embodiments of pooling are contemplated. For example, in an alternative
exemplary embodiment of the methods 400 and 500, the Steps 420C and 430C are omitted from
the method 400. Instead, pooling is performed in the Step 510A of the method 500A prior to
consolidation. After the tickets are pooled in the Step 510A, the computer system 260 transmits
the results of pooling in the Step 510A, and the computer system 220 receives such results in the
Step 510B. Pooling participants are notified of the pooling prior to consolidation.
An exemplary alternative embodiment of pooling performed in the Step 5 10A is now described. In this exemplary embodiment, pooling is performed after all tickets for a lottery are submitted. In the discussion below, S is the block size; T is the overall number of tickets; M is the maximum number of tickets any one person enters to be pooled; L is the desired number of blocks per pool (leverage); $L_{\text{max}}$ is the maximum number of blocks per pool; $L_{\text{min}}$ is the minimum number of blocks per pool; $B=T/S$ is the total number of blocks of tickets entered for pooling; and $P_{\text{min}}=M/S$ is the number of blocks of tickets entered by the user with the most tickets to be pooled. Because no individual user can have more than one block in a pool, $P_{\text{min}}$ is also the minimum number of pools necessary if all of this user's tickets are pooled with the necessary leverage ratio. This exemplary alternative embodiment proceeds unless $B<L_{\text{min}}$. Otherwise, pooling is performed according to the following steps:

**Step 1:** $B/L$ is a desirable number of pools based on the number of blocks and the desired leverage ratio. If $B/L<P_{\text{min}}$, go to Step 2. If $B/L\geq P_{\text{min}}$, then calculate $P_{\text{upper}}=\text{ceiling}(B/L)$ and $P_{\text{lower}}=\text{floor}(B/L)$. Then calculate $P$, the number of pools, using the following criteria:

1. If $P_{\text{lower}}=0$ and $B/P_{\text{upper}}<L_{\text{min}}$, pooling cannot proceed.
2. Else: If $P_{\text{lower}}=0$ and $B/P_{\text{upper}}\geq L_{\text{min}}$, set $P=P_{\text{upper}}$.
3. Else: Set $P=0$.
4. Else: If $B/P_{\text{lower}}>L_{\text{max}}$ and $B/P_{\text{upper}}<L_{\text{min}}$, then remove blocks until $B=P_{\text{lower}} * L_{\text{max}}$.
   - Remove blocks one at a time, at random, always from the user with the most blocks. Set $P=P_{\text{lower}}$.
5. Else: If $B/P_{\text{upper}}<L_{\text{min}}$, then $P=P_{\text{lower}}$.
6. Else: Calculate $P$ according to the following equation:

$$P = \begin{cases} P_{\text{upper}} & \text{if } \left| \frac{B}{P_{\text{upper}}} - L \right| < \left| \frac{B}{P_{\text{lower}}} - L \right| \\ P_{\text{lower}} & \text{otherwise} \end{cases}$$

(1)
Skip to Step 4, as proceeding to Steps 2 and 3 is contingent on the criterion (B/L<P_min) being satisfied in the beginning of Step 1.

Step 2. If B/L<P_min, then let L'=L-1 and check if B/L'>P_min. If so, repeat Step 1 using L’ instead of L. If not, repeat this step, subtracting 1 from L’ until B/L'>P_min. If L’<L_min (this should only happen if B/Lmin>P_min), then go to Step 3.

Step 3. Not all of the tickets can be pooled given the minimum leverage requirement L_min and the block size S (which is also the maximum number of tickets an individual can have in a pool). Some blocks must be removed to proceed. Remove blocks randomly from the user with the most blocks (thereby changing P_min and B), one by one, until B/L_min>P_min. Now repeat Step 1.

Step 4. Once P, the number of pools, is determined, begin random allocation of blocks to pools. Create pools 1 . . . P. The iterative process for allocation follows:

Substep a. Begin by letting x=0. Go to Substep (b);

Substep b. Select a user at random. If there is no other user with blocks to be allocated, the pooling method is complete. Otherwise, go to Substep (c); and

Substep c. Randomly select a pool with x blocks in it which does not contain a block from the selected user. Randomly select one of the user's blocks and allocate it to the chosen pool. If the user has more blocks to allocate and there are pools remaining with x blocks, repeat this Substep. If the user has no more blocks, go to Substep (b). If there are no pools left with x blocks, increment x by 1 and repeat this Substep.

Another exemplary alternative embodiment of pooling performed in the Step 420C is now described. In this exemplary embodiment, the computer system 260 creates and manages virtual pools that have the following characteristics:

1. A virtual pool is the online analog to a traditional, paper-tracked lottery pool.
2. A virtual pool can be a fixed size or it can be permitted to grow to accommodate an arbitrary number of entries.

3. Fixed-size virtual pools allow a pooling participant to choose among various size options for an individual lottery. These virtual pools require balancing when:

\[
\text{[number of entries] \mod [pool size]} \neq 0
\]  

(2)

Balancing is likely to occur, because there is only a \(1:\{\text{pool size}\}\) chance for the result above to equal 0.

4. Social, private, or charity-focused virtual pools grow organically according to the number of entries they receive and as a result, these virtual pools do not require balancing. They have no predefined rules or parameters other than ticket type, i.e., uploaded tickets are accepted only if they pertain to the lottery associated with the pool.

5. Pooling participants see single virtual pools for each entry they make in a fixed-size pool.

For virtual pools that specify a pool size, there are two approaches for consolidation: static balancing and dynamic balancing. They are not mutually exclusive, but static pool may be more advantageous. The pool size is a predefined pooling parameter. Rebalancing according to the pool size is a predetermined pooling rule.

Dynamic balancing balances the pools on demand as ticket entries are being added. Dynamic balancing has the following characteristics: (1) A balancing step may not always be required for a new entry; (2) A balancing step may be required before or after a new entry is added; and (3) Balancing activity is predictable based on the [number of entries] and [pool size] values, even if the [number of entries] value is not known beforehand, since it can be accurately estimated by choosing a value for [number of entries]. Static pooling balances the pools in a single pass once the virtual pool is closed to new entries. At that time both factors that affect the balancing equation are known.
[00106] When allocating lottery ticket entries at random into virtual pools, the computer system 260 constructs a series of randomly identical pools for a given lottery/pool-size combination, and then populates the randomly identical pools with tickets from the pooling participants. For any given two entries that are otherwise identical, the computer system 260 may decide to favor or disfavor one entry over another when populating a randomly identical pool. There are various reasons for this, including marketing ones. As long as the pools are still randomly identical, the pooling coordinator 210 is maintaining its contract with the pooling participants. Two exemplary allocation strategies are described below.

[00107] First-in least moved strategy: By assigning a weight value to an entry, the computer system 260 can make that entry less likely to be shuffled during a rebalancing operation. The balancing algorithm takes an entry’s weight into account when deciding which entries to move. An example of this would be to favor early entries over later ones without disturbing any of the randomness factors relating to chances at winning.

[00108] Second-chance strategies: The pooling coordinator 210 is not a lottery but, as an independent business, it could use its pooling technology to award prizes at its own discretion. For example, by creating an entry during a particular day or time span, pooling participants might be able to win prizes unrelated to the lottery business, i.e. a "Friday night scanners 2nd chance pool". Because scanning lottery tickets also captures location information, players might be able to win prizes by scanning from a specific location. For example, a convenience store might be interested in providing random awards to players who scan their tickets while at the convenience store.

[00109] Referring now to FIGS. 7A and 7B, there is illustrated an exemplary method, generally designated as 700, of consolidating lottery tickets, in accordance with an exemplary embodiment of the present invention. Consolidation is desirably executed in blocks of five tickets. It is to be understood that consolidation is not limited to being performed for blocks of five tickets. Other exemplary embodiments in which consolidation is performed for individual tickets or blocks of two, three, four, six, or any other number of tickets are contemplated.
[00110] Consolidation according to the method 700 is guided by a few general predefined consolidation rules: (1) Consolidation pools are filled to a max fill percentage (100%) by moving batches of tickets from oldest unfilled non-consolidation pools to oldest unfilled consolidation pools; (2) Once consolidation pools are filled, the oldest remaining unfilled pool is filled by moving batches of tickets from a next oldest unfilled pool; (3) The second rule is repeated until only one pool is left; (4) If, after Rule 3, the remaining pool is not filled to minimum fill percentage, then it is determined whether there are enough overflow slots to move all batches tickets from this pool to the already filled pools; (5) If the determination in Rule 4 is that there are enough overflow slots, batches of tickets are moved from the last pool to the filled pools, adding one batch of ticket to each filled pool, starting with the newest unfilled pool, before adding another batch of tickets to each filled pool. Thus, all filled pools receive one extra batch before a second batch is added to any filled pool and so on; (6) Rule 5 is followed until only filled pools remain; (7) If the determination in Rule 4 is that there are not enough overflow slots, tickets are moved from filled pools back into an unfilled pool until the unfilled pool reaches the minimum fill percentage; one batch from each pool is moved, starting with the newest filled pool; (8) If at any point the tickets are moved into a pool in which a pooling participant already has tickets, swap these tickets with another pooling participant's tickets, making sure the swapped pooling participant does not have tickets in the pool. Tickets are swapped from the newest possible filled pool; and (9) If it is not possible to fill pools and still have only one batch of tickets for each pooling participant in a pool, allow more than one batch of tickets per pooling participant, e.g., 2, 3, etc. Consolidation provides that all pools will be filled to at least the minimum fill percentage and no more than the overflow percentage.

[00111] Each of Steps 710 through 750B performed by the computer system 260 to effect consolidation in accordance with the predefined consolidation rules are described below:

[00112] Step 710: The computer system 260 finds the oldest pool that is not max filled. The computer system 260 moves each batch of tickets from this pool to the oldest consolidation pool that is not max filled. When moving ticket batches, the computer system 260 attempts to
move tickets added to the pool more recently before moving tickets added to the pool at an earlier time.

[00113] If the consolidation pool to which a ticket batch is moved contains another batch of tickets for the given pooling participant, then the computer system 260 swaps the tickets just moved with tickets from another pool. Swapping is performed according to limited swapping (see swapping process below) with the following guidelines:

[00114] a. First use the newest possible consolidation pool for the swap;

[00115] b. If all consolidation pools have tickets for the pooling participant, swap into the newest possible max filled pool; and

[00116] c. If all max filled pools have tickets for that pooling participant, move the ticket batch back to the pool from which it came.

[00117] This Step 710 is repeated for each successive next oldest pool that is not max filled, until one of two conditions is met:

[00118] d. All consolidation pools are filled; or

[00119] e. An attempt has been made to move all tickets that are in non-max filled pools.

[00120] The computer system 260 determines if any tickets remain in unfilled non-consolidation pools, Step 715. If there are none, the method 700 is complete and ends at a Step 790. If there are, then the method 700 proceeds to a Step 720.

[00121] Step 720: The computer system 260 finds oldest pool that is not max filled. The computer system 260 moves tickets from next oldest non-max filled pool to this pool until it is at the pool maximum. If there are not enough tickets in the next oldest non-max filled pool, the computer system 260 moves as many as possible and then repeats the process using the next oldest non-max filled pool. It swaps tickets to consolidation pools or max filled pools if
necessary using the same algorithm outlined in Step 710. Swapping is performed according to limited swapping (see swapping process below).

[00122] Once the oldest non-max filled pool is at the pool maximum, the computer system 260 repeats the Step 720 for the next oldest non-max filled pool. It continues until one of two conditions are met:

[00123] a. Only one non-max filled pool remains, or

[00124] b. All ticket batches that could be moved according to the rules have been moved.

[00125] The computer system 260 checks if any tickets remain in unfilled non-consolidation pools, Step 725. If not, the pool consolidation process 700 is complete and ends at the Step 790. If there are, the method 700 proceeds to a Step 730.

[00126] Step 730: The computer system 260 repeats Step 710, but performs unlimited swapping in the swapping process. This provides that either all consolidation pools will be filled or no more tickets will remain in unfilled non-consolidation pools.

[00127] The computer system 260 checks if any tickets remain in unfilled non-consolidation pools, Step 735. If not, the pool consolidation process 700 is complete and ends at the Step 790. If there are, the method 700 proceeds to a Step 740 illustrated in FIG. 7B via 7B in FIG. 7A.

[00128] Step 740: The computer system 230 repeats Step 720, but performs unlimited swapping. This provides that there is no more than one unfilled non-consolidation pool.

[00129] The computer system 260 checks if any tickets remain in an unfilled non-consolidation pool, Step 745. If all unfilled non-consolidation pools are either empty or are at or above the minimum fill percentage, the pool consolidation process 700 is complete and ends via the Step 790. Otherwise, the method 700 proceeds to a Step 747.
Step 747: The computer system 260 calculates the total overflow capacity. The total overflow capacity equals the number of filled pools multiplied by the overflow percentage above the pool maximum. If any consolidation pools are not at the pool maximum, the computer system 260 adds their remaining capacity. This is the overflow that can be handled by the current pools.

The computer system 260 then checks the total overflow capacity against the total number of unfilled tickets (tickets in the unfilled, non-consolidation pool), Step 749. If it is equal to or greater than the total number of unfilled tickets, the method 700 proceeds to a Step 750A. Otherwise, it proceeds to a Step 750B.

Step 750A: The computer system 260 moves each batch of tickets in the remaining unfilled pool to the filled pools, in accordance with the following rules: (1) Add one batch of tickets per filled pool, starting with the newest filled pool; (2) When all filled pools have one extra batch of tickets, add another batch, starting with the newest filled pool; (3) Do not fill any pool past pool maximum plus the overflow percentage capacity; and (4) Swap tickets if necessary, with no limit (see swapping process below). The pool consolidation process 700 is now complete and ends via the Step 790.

Step 750B: The computer system 260 moves ticket batches from the newest filled pools to the remaining unfilled pool until the number of tickets in the unfilled pool reaches the minimum fill percentage, in accordance with the following rules: (1) Move one batch of tickets from each filled pool, starting with the newest filled pool; (2) When all filled pools have had one batch of tickets moved from them, move another batch, starting with the newest filled pool; (3) Do not move tickets from a pool that is at the minimum fill percentage; and (4) Swap tickets if necessary, with no limit (see swapping process below). The pool consolidation process 700 is now complete and ends via the Step 790.

Note that the same swapping is performed in Step 720 and Step 740, except the first swap (Step 720) does not allow multiple batches per member per pool but the second swap (Step 740) does. Unlimited swapping is not performed in the Steps 710 and 720 because there is
a chance that doing this will put multiple batches for the same pooling participant within a pool when unnecessary. After the computer system 260 iterates over all pools once with limited swapping, any future swaps that involve placing multiple batches for the same pooling participant in a pool are necessary.

[00135] Note also that with the proper percentages set, it should be unlikely that the Step 750B is performed. For instance, if the overflow percentage is 15% and the minimum fill percentage is 75%, the Step 750B would be performed if there are four or fewer filled pools and the final unfilled pool is 65-70% filled.

[00136] Unless the minimum fill percentage is very high and the number of pools very low, the Step 750B should provide that all pools are filled. The Step 750B may not provide that all pools are filled if removing the extra tickets between the pool maximum and the minimum fill percentage from filled pools would not be enough to take the one unfilled pool up to the minimum fill percentage.

[00137] To determine how low a pool total would be to cause problems, some calculations are illustrative. Assume a pool maximum of 100% and the following variables: (1) P = number of filled pools (this includes consolidation pools); (2) U = total unfilled tickets; (3) M = minimum fill percentage; and (4) O = overflow percentage.

[00138] From the Step 747, we have U > P * O. For the Step 750B to fail, we would need (P * (100 - M)) + U < M. Combining these two and solving yields the following relationship between P and the other variables:

\[ P < \frac{M}{100 - M + O} \]  

(3)

[00139] If we use a starting O of 15% and M of 85%, this becomes:

\[ P < \frac{85}{100 - 85 + 15} = 2.8 \]  

(4)

[00140] So if there are 2 or fewer filled pools (round down), the computer system 260 won't be able to fill the last unfilled pool to the minimum fill percentage using the Step 750B. In
these cases, the computer system 260 fills the last pool as much as possible, moving tickets from filled pools until all filled pools are at the minimum fill percentage.

[00141] Swapping Process: Limited Swapping

[00142] For discussion below, the pool being swapped from is referred to as the "Sending Pool" and the pool being swapped to is referred to as the "Receiving Pool." To swap tickets, the computer system 260 first checks whether the Receiving Pool already has a ticket batch for the pooling participant whose tickets need to be swapped. If so, this Receiving Pool cannot be used, and the next newest pool should be examined. Otherwise, the computer system 260 goes through the Receiving Pool and finds the first batch of tickets belonging to a pooling participant that does not have tickets in the Sending Pool. The computer system 260 exchanges these tickets between the two pools. When searching through ticket batches, the computer system 260 tries to swap tickets added to the pool more recently before swapping tickets added to the pool at an earlier time.

[00143] Swapping Process: Unlimited Swapping

[00144] Unlimited swapping is the same as limited swapping, except that success is guaranteed by increasing the allowed ticket batches per member per pool until a legitimate pool can be found for the tickets being moved.

[00145] When a pooling participant has tickets in every pool, the pooling coordinator 210 allows two or more batches per pool per member. The computer system 260 then attempts to swap the tickets again. Under unlimited swapping, when a batch is added to a pool in which the pooling participant already has a batch, the newly moved batch will be left as a second batch in the pool. The next time a batch for this pooling participant is moved to the same pool, the swapping process will look for the newest swappable pool in which the pooling participant has only one ticket batch. The computer system 260 repeats this same process until a pool is found for the batch of tickets. Note that all pools will receive a second batch of tickets for a given member before a third batch would be allowed, and so on.
[00146] In an exemplary embodiment, each ticket movement during consolidation in accordance with FIGS. 7A and 7B is logged. In an exemplary embodiment, all data is logged to log files separate from the database 265. So if a block of tickets is moved, a log will show exactly what was moved from where, to where and when. In another exemplary embodiment, consolidation data is logged within the records of tickets moved. When a batch of tickets is moved to a new pool ID, new record data is placed into the consolidation field of the record of the batch of tickets. The entire record is hashed and digitally signed. This has the following effect: (1) The original pool designation is left intact, so that it may still be traced from ticket scanning to ticket assigning; and (2) It can be confirmed that the ticket batch was moved only by the server 260 via use of the record hash and the digital signing of the hash by the server 260.

[00147] During the consolidation process, a ticket batch could be moved to one pool, then to another. Also ticket blocks could be moved in a non-fifo or non-lifo sequence. So ticket blocks can be moved in any sequence required by the consolidation algorithm 700. If a ticket block is moved during consolidation, then is moved again, the consolidation fields in the record of the ticket block are overwritten with the new data. The consolidation logs, however, record all movements so that consolidation may be audited.

[00148] In an exemplary embodiment described with reference to FIGS. 1A and IB, the one or more scannable items 140 and 140’ are bar codes. It is to be understood that it is contemplated that the bar codes 140 and 140’ may be line bar codes or any variety of two-dimensional bar codes. In other appropriate embodiments, the one or more scannable items may be RFID chips or other information markers in the lottery ticket(s) 100 and 100’. Although it is described that the unique identification code(s) of the lottery ticket(s) 100 and 100’ is encoded in the bar codes 140 and 140’, it is also contemplated that the unique identification code(s) is encrypted. In such an embodiment, either the pooling coordinator 260 or the lottery organizer 250 decrypts such information.

[00149] Although it is described above that the one or more scannable items are decoded by the computer system 260, it is to be understood that decoding of the one or more scannable items may be performed by the computer system 220, 230, or 250. Thus, in embodiments in
which the computer system 220 or 230 captures an image of the lottery ticket(s) 100, the computer system 220 or 230 may decode the one or more scannable items captured in the image. Thus, computer system 220 or 230 transmits the decoded information to the computer system 260 in the Step 350. Alternatively, the computer system 220 or 230 transmits the captured image, and the computer system 260 transmits the image to the computer system 250 for decoding and validation in the Step 420B.

[00150] Further, although it is described above that the one or more scannable items are decoded, other exemplary embodiments in which the text block 110 is recognized by optical character recognition (OCR) are contemplated. OCRing may be performed by the computer system 220 or 230, which transmits the OCRed information to the computer system 260 in the Step 250. Alternatively, the computer system 220 or 230 transmits the captured image, and the computer system 260 in the Step 410B or the lottery organizer 250 OCRs the text 110.

[00151] As described above, it is contemplated that the lottery ticket 150 may comprise one or more lottery tickets 100', each of which may have its own scannable item or all of which together have one scannable item, e.g., a single bar code 140'. It is contemplated that in the Step 350, the user 220 may scan each lottery ticket 100 or 100' individually or multiple lottery tickets 100 or 100' at one time. In the Step 350, the computer system 220 may upload one image of multiple lottery tickets at a time for decoding, or the computer system 220 may decode the encoded information from the one image and transmit it in the Step 350. One image may correspond to a batch of lottery tickets 100 or 100'. For example, both Powerball® and Mega Millions® can fit five tickets 100' per single ticket print-out 150. Therefore, the user 220, if registering five tickets 100', need only do a single scan in the Step 350; if registering 10 tickets 100, the user 220 performs two scans in the Step 350; and so on.

[00152] Other exemplary embodiments of uploading the lottery tickets into the computer system 260 are contemplated. For example, it is contemplated that in another exemplary embodiment, the user 220 may use a bar code scanner connected to the computer system 220 to scan the bar codes 140 or 140' on the one or more lottery tickets 100 or 100'. Or, it is contemplated that in another exemplary embodiment, the user 220 uses a digital camera to take a
photograph of each of the one or more lottery tickets 100 and uploads the photograph of each lottery ticket 100 to the computer system 260 using the computer system 220.

[00153] As described above, the pooling interface portal may be either or both of a pooling interface website and a pooling interface application. In embodiments in which both are used, it is contemplated that the user 220 may switch from the pooling interface website to the pooling interface application after registering an account with the pooling coordinator 210 online in the Steps 310 through 320. Alternatively, it is contemplated that the pooling coordinator 210 may provide a first secure website through which the user 220 registers with the system 210 in the Steps 310 through 320 and a second secure website, the pooling interface website, through which the user 220 pools the lottery tickets 100 or 100'. It is further to be understood that any of the websites describe herein may be configured to allow the user 220 to download the pooling interface application, before, during, or after registration with the system 210. Furthermore, it is contemplated that the web server 260 may host an unsecured website to provide information on pooling services to unregistered viewers. It is also to be understood that in an exemplary embodiment, the user 220 may purchase the lottery tickets 100 online and use the pooling interface website or the website at the place of purchase to transfer the online lottery tickets 100 to the computer system 260 in the Step 350.

[00154] In another exemplary alternative embodiment of the computer system 210, the computer system 210 comprises an application server 280, separate from the computer system 260 and behind a firewall 270. The application server 280 receives pooling requests from the computer systems 220 and 230 via the computer system 260 over an internal network 290 and performs pooling based on the requests and predetermined rules. The application server 280 maintains the pools in a computer readable storage medium 285 in a database. In an exemplary embodiment, the computer readable storage medium 285 is a hard disk drive. In such an embodiment, the computer system 260 maintains the user accounts in the database 265.

[00155] In an exemplary embodiment, the system 260 only accepts uploads of lottery tickets from the computer system 220 if the computer system 220 is a smart phone that embeds information identifying the smart phone and GPS data of the smart phone in data messages,
including the message(s) uploading the one or more lottery tickets 100 to the computer system 260, so that the computer system 260 is able to identify the smart phone that uploaded the one or more lottery tickets and its location. The GPS data may be used in the validation process to ensure that the user 220 purchased a lottery ticket in compliance with relevant laws or regulations.

[00156] As used herein, a "pool type" is the set of characteristics that distinguish one pool from another. Such characteristics include any of the following: size of batches or blocks that may be added to the pool (e.g., number of lottery tickets in a batch or block), number of batches or blocks permitted in the pool, and payout rules (e.g., percent of jackpot that goes to the pooling participant that submitted the winning ticket, consolidation prize for all other pooling participants).

[00157] Referring now to FIGS. 8A through 8L, there is illustrated an exemplary embodiment of the pooling interface portal, generally designated as 800, in accordance with an exemplary embodiment of the present invention. The interface 800 provides a personal interactive and visual representation of a pooling participant's lottery drawing experience. The purpose of this interface 800 is to display to the pooling participant a representation of tickets that he owns, as well as tickets involved in any pool in which he is included. The interface can work off of the database 265 containing the pooling participant's pooling data. Maps, charts and statistics can be displayed to the pooling participant on the interface.

[00158] In one embodiment, winning lottery numbers are displayed in the interface 800 to the pooling participant as balls rolling out from a side of the display of the interface, as balls dropping down from above, or as popping into existence in any location on the display. After each ball is displayed to the pooling participant, the maps, charts, and statistics dynamically change to reveal updated information, including updated chances of winning one or more lottery amounts, and updated pool information, in which the pooling participant is included, showing the entire pool's remaining chance of winning. This display could occur in real time, relative to a lottery drawing in the Step 540A of the method 500A, or could be displayed at a later time upon request of the pooling participant.
The interface 800 can match the look and feel of the rest of the pooling interface portal, but in an alternative embodiment, the content is separated into two sections or columns. In one section or column, the pooling participant's personalized information is displayed. The pooling participant's personalized information includes the pooling participant's and location, the lottery pool amount (current drawing amount), the size of the pool in which the pooling participant is participating (i.e., the number of tickets in the pool), as well as other relevant statistics and information, such as a calculated chance of winning a lottery amount. An example of the interface illustrating this information 810 is illustrated in FIG. 8A.

In one embodiment, as shown in FIGS. 8A through 81, the displayed statistics and information 810 include two specific animated numbers: 1) the number 812 of tickets the pooling participant personally has in the pool; and 2) the total number 814 of tickets in the pool. The display of these numbers 812 and 814 can be programmed to change and enlarge during the animated click through of the lottery drawing to reflect the pooling participants changing chances of winning.

In another section, or column of the display 800, a map 820 is illustrated showing the relevant geographic region of other ticket holders included in a pool with the pooling participant, as well as the pooling participant's location 822. In the embodiment illustrated in FIGS. 8A and 8B, the pooling participant's location is Philadelphia, and tickets included in the pooling participant's total pool are shown throughout a darkened area 814 (labeled in FIG. 8B). Lightened areas 826 are geographic locations not participating in pool play or are areas in which no ticket holding pooling participants are currently residing. Accordingly, the map 820 shows the relative locations of the other pooling participants holding tickets that are pooled with the pooling participant's tickets. The map 820 indicates the pooling participant's personal tickets and location, and fills in the map 820 with different indicators to represent the amount and location of pooled tickets. The pooled tickets are tickets that are in the pool, but ownership is assigned to other poolers.

The pooling participant is then prompted to start his personal lottery drawing with a clickable button, or it can begin automatically. As the drawing experience progresses, a first
ball 832 rolls on screen from the right-hand side and stops under the map 820, as illustrated in FIG. 8C. In other implementations of the interface 800, each lottery ball could also roll in from left to right, drop from above, or pop into existence. Each ball is labeled with a lottery ball number that reflects the actual lottery number. Again, this can occur in real time, or at a later time as requested by the pooling participant.

[00163] As shown in FIG. 8D, after the lottery ball 832 stops, the map 820 refreshes to reveal the remaining valid tickets (both the pooling participant's personal tickets and the cumulative pooled tickets) that still have a chance of winning, and the statistics and information section (or column) 810 also refreshes, updating the numerical chances remaining to win an amount certain (regarding both the pooling participant's personal tickets and the cumulative tickets of the entire pool in which the pooling participant is included). The number of remaining valid tickets is indicated by the reduced density in the darkened area 824.

[00164] In one variation on the embodiment of the interface 800, a button 840 appears (for user click action) to advance to the next ball. Another variation on the embodiment provides for lottery ball advancement as an automated process not requiring any affirmative action by the pooling participant, e.g., no mouse click. This process (individual lottery ball advancement) continues through the rest of the animation (through the rest of the lottery) until the pooling participant has either been eliminated from winning the lottery jackpot, until the pooling participant has no owned or pooled tickets still active, until the pooling participant requests further animation, or until the last lottery ball is revealed. If the pooling participant has no remaining valid tickets before the last ball is revealed, then either all of the final numbers are revealed at once, or the drawing continues along its normal process, per pooling participant request. FIGS. 8E through 8H illustrate exemplary lottery ball 834, 836, 838, 839 progression.

[00165] In another embodiment of the interface 800, any lottery balls 830 that are not order-dependent (for lottery winning requirements) can be re-ordered. The purpose of this feature of the interface 800 is to show the lottery balls 830 in an order that maximizes the number of tickets available for the longest time possible. Naturally, this feature could not be implemented if the display is in real time. Any balls 830 that are order dependent, e.g.,
Megaball® or Powerball®, are displayed in their required order. This ordering feature could be implemented by user input (request).

[00166] In the event the pooling participant does not win the jackpot, a message 840 appears onscreen informing the pooler that he does not have any personal or pooled tickets that won the jackpot, such as that shown in FIG. 81. If the pooling participant has at least one personally owned or pooled ticket that matches the winning jackpot, a message appears informing the pooling participant that he (or his pool) has won and will inform him how much he will receive in the payout. If the pooling participant has any tickets that have won non-jackpot prizes, the ticket numbers and their redemption value will be displayed to the pooling participant either on the drawing interface 800, or they will be directed to another interface (via hyperlink or button, for example), in which the pooling participant they can see his non-jackpot winning tickets.

[00167] Referring now to FIG. 9, there is illustrated an exemplary alternative embodiment of the ticket 100, generally designated as 100", in accordance with an exemplary embodiment of the present invention. The ticket 100" differs from the ticket 100 in that the ticket 100" is only active when the jackpot for a specified lottery is equal to or greater than a predetermined threshold 135, which is printed on the ticket 100". An exemplary value of the predetermined threshold is $200 million. In an exemplary embodiment, the ticket 100 is active for the next lottery drawing.

[00168] Various ways of setting the predetermined threshold 135 are contemplated. In one exemplary embodiment, the predetermined threshold 135 is predefined by the lottery organizer 250. In another exemplary embodiment, the predetermined threshold 135 is defined by the purchaser at the point of sale. In any of these embodiments, the ticket 100" is activated only after the lottery to which it pertains develops a jackpot that is equal to or greater than the predetermined threshold 135.

[00169] An exemplary use of the ticket 100" is as multi-week ticket. The user 220 could purchase a ticket 100" that is good for multiple drawings during a week. Another exemplary use
of the ticket 100" is as a multi-drawing ticket. The user 220 could purchase a ticket 100" that is
good for 10 drawings with a $200M threshold, for example. Such a ticket 100" would be active
for each of the next 10 drawings in which the jackpot for the lottery 120 is above $200 million.
It is contemplated that the next 10 drawings could be consecutive drawings in the event there are
no winners, or it could be separate drawings spread out over more than one week.

[00170] In an exemplary embodiment of the system 200, the computer system 260 and the
database 265 are configured for storing the ticket 100" in a separate table of the database 265.
In such embodiment, the methods 300 and 400 are provide for the user 220 to upload the ticket
100" in the Step 350 and for the computer system 260 to receive the ticket 100" in the Step
410B, transmit the identification code in the barcode 140 to the lottery operator 250, receive the
ticket information from the lottery operator 250 and validation indication in the Step 410C, and
store the lottery ticket 100" in the Step 450B in a table of the database 265 separate from the
pooling tables. The computer system 260 transmits updates and reminders about whether the
ticket 100" is active or inactive for a given drawing. When the ticket 100 becomes active, the
user 220 may request pooling for the ticket 100" previously uploaded and validated. The
method 300 thereby begins at the Step 370 to pool the ticket 100".

[00171] Several fraud prevention techniques are contemplated to combat the following
scenarios: (1) ticket being scanned by someone other than the purchaser; (2) copy of ticket
scanned by someone other than the purchaser; and (3) double pooling. Such techniques may be
performed by the computer system in an exemplary optional Step 415B in the method 400B.

[00172] In a first embodiment of the Step 415B, to combat the first scenario, the computer
system 260 presents challenge questions to the user 220, and the user 220 must provide correct
answers to these questions. Exemplary questions include one or more of the following: (a) What
date and time was the ticket 100 purchased?; (b) In which town was the ticket purchased?; (c) At
which location was the ticket 100 purchased?; (d) What are the last four digits of the credit or
debit card used to purchase the ticket 100? The last question assumes that the system 210,
specifically the lottery operator 250, permits purchase of the lottery 100 via credit or debit card.
The user 220 provides answers to these questions in an optional Step 355, and the computer
system 260 receives them in the Step 415B. A wrong answer results in a ticket being flagged as possibly fraudulent, and the user 220’s account may also be flagged.

[00173] In a second embodiment of the Step 415B, to combat the first scenario, the computer system 260 could request an authentication code from the user 220. The authentication code may be provided to the user 220 at the point of sale. The user 220 inputs the authentication code in the Step 355, and the computer system 260 receives it in the Step 415B. An incorrect authentication code results in a ticket being flagged as possibly fraudulent, and the user 220’s account may also be flagged.

[00174] In a third embodiment of the Step 415B, to combat the first scenario, the point of sale requires that the user 220 present an authentication code, e.g., a user ID, a PIN, or a scannable ID card, that corresponds to the user 220. The ticket 100 is then linked to the authentication code and transmitted by the point of sale to the pooling coordinator. In the step 415B, the computer system 260 determines whether the scanned ticket 100 was previously linked to the user 220. If not, the ticket 100 is flagged as possibly fraudulent, and the user 220’s account may also be flagged.

[00175] In a fourth embodiment of the Step 415B, to combat the first scenario, the computer system 260 requires that the user 220 scan the ticket 100 within a predetermined time after sale, e.g., 15, 30, 60, 90, etc., minutes. The time window may include a delay before opening to prevent a clerk at the point of sale from scanning the ticket 100. It is unlikely that someone else would be able to steal the ticket, scan it, and return it to the user 220 within the time window. Thus, this fraud prevention technique is desirable and combats fraud scenarios 1-3.

[00176] Fraudulent scenario 2 may be combated by printing the ticket 100 on paper that cannot be photocopied or for which a photocopy would create artifacts that are observable. The barcode 140 may also be printed on the ticket 100 in such a way that a photocopy of the ticket would create a distorted barcode.
Fraudulent scenario 3 may be combated by allowing other pools to check with the computer system 260 to determine whether a ticket 100 is pooled in two places. The ticket 100 may be uploaded to the computer system 260 to check whether it has been already pooled in the system 260. The computer system 260 may also communicate electronically with other pooling computer systems to determine whether pooled tickets are also pooled with such other pooling computer systems.

In an exemplary embodiment, the computer system 260 provides a summary of a pool. One scenario involves a pooling participant collecting money for a pool and purchasing lottery tickets. The pooling participant scans the tickets into the computer system 260, provides a name for the pool, and then prints out a summary of the pool provided by the computer system 260. Such print out may provide a 2D barcode, a URL/short URL, and all of the ticket numbers. The pooling participant distributes this sheet to the pool members, either via email, app push, or hard copy. The pool members can scan the 2D barcode on the sheet or enter the URL or pool ID and transmit such information to the computer system 260. The computer system 260 may then provide the pool member with a summary of the pool, so that the pool member can verify that the pool was properly created. The pooling coordinator 210 is a neutral party that provides accurate pooling information to the pool members.

Several examples of pool types, payouts, pooling, and consolidation are now provided.

**EXAMPLE 1 - POOL TYPES**

In an exemplary embodiment, when a lottery jackpot is equal to or less than $100,000,000, a pooling participant has one option for a pool-type in which to place his one or more lottery tickets 100. In the exemplary embodiment of the method 300 discussed above, the pooling participant selects this pool type in the Step 370. In Table 2 below, there are summarized the characteristics of the single pool type, including the amount of winnings to be distributed in the Step 540A of the method 500A:
<table>
<thead>
<tr>
<th>Size of batch for pool</th>
<th>Number of pooling participants</th>
<th>Total chances</th>
<th>Winnings</th>
<th>Consolidation % of jackpot</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>100</td>
<td>500</td>
<td>50% of jackpot</td>
<td>0.51%</td>
</tr>
</tbody>
</table>

**Table 2**

**EXAMPLE 2 - POOL TYPES**

[00181] In an exemplary embodiment, when a lottery jackpot is greater than $100,000,000, a pooling participant has two options for a pool-type in which to place his one or more lottery tickets 100. In the exemplary embodiment of the method 300 discussed above, the pooling participant selects from these two pool types in the Step 370. In Table 3 below, there are summarized the characteristics of the two pool types, including the amount of winnings to be distributed in the Step 540A of the method 500A:

<table>
<thead>
<tr>
<th>Size of batch for pool</th>
<th>Number of pooling participants</th>
<th>Total chances</th>
<th>Winnings</th>
<th>Consolidation % of jackpot</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>100</td>
<td>500</td>
<td>50% of jackpot</td>
<td>0.51%</td>
</tr>
<tr>
<td>5</td>
<td>500</td>
<td>2500</td>
<td>$30,000,000</td>
<td>(Jackpot-$30m)/499</td>
</tr>
</tbody>
</table>

**Table 3**

**EXAMPLE 3 - JACKPOT PAYOUTS**

[00182] An example of what jackpot payouts look like as the jackpot grows in size for pools of 100 people (500 tickets) is illustrated in Table 4:

<table>
<thead>
<tr>
<th>Jackpot</th>
<th>Number of Poolers</th>
<th>Total Chance</th>
<th>Winner Take Home</th>
<th>Consolation Take Home/Person</th>
</tr>
</thead>
<tbody>
<tr>
<td>$50,000,000</td>
<td>100</td>
<td>500</td>
<td>$25,000,000</td>
<td>$252,525</td>
</tr>
<tr>
<td>$75,000,000</td>
<td>100</td>
<td>500</td>
<td>$37,500,000</td>
<td>$378,788</td>
</tr>
<tr>
<td>$125,000,000</td>
<td>100</td>
<td>500</td>
<td>$62,500,000</td>
<td>$631,313</td>
</tr>
<tr>
<td>$285,000,000</td>
<td>100</td>
<td>500</td>
<td>$137,500,000</td>
<td>$1,388,889</td>
</tr>
</tbody>
</table>
EXAMPLE 4 - JACKPOT PAYOUTS

[00183] An example of what jackpot payouts look like as the jackpot grows in size for pools of 500 people (2,500 tickets) is illustrated in Table 5:

<table>
<thead>
<tr>
<th>Jackpot</th>
<th>Number of Poolers</th>
<th>Total Chance</th>
<th>Winner Take Home</th>
<th>Consolation Take Home/Person</th>
</tr>
</thead>
<tbody>
<tr>
<td>$50,000,000</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>$75,000,000</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>$125,000,000</td>
<td>500</td>
<td>2500</td>
<td>$30,000,000</td>
<td>$190,381</td>
</tr>
<tr>
<td>$285,000,000</td>
<td>500</td>
<td>2500</td>
<td>$30,000,000</td>
<td>$490,982</td>
</tr>
<tr>
<td>$350,000,000</td>
<td>500</td>
<td>2500</td>
<td>$30,000,000</td>
<td>$641,283</td>
</tr>
</tbody>
</table>

Table 5

EXAMPLE 5 - POOLING EXAMPLE

[00184] An example of pooling in accordance with multiple iterations of the method 600 is now described. It is to be understood that the exemplary steps described below are performed by the computer system 260, as described with reference to the method 600.

[00185] The following parameters are assumed: (1) consolidation pools: 3; (2) consolidation percentage: 85%; (3) minimum fill percentage: 85%; (4) overflow percentage: 15%, (5) block size: 5; and (6) pool maximum: 100%.

[00186] Iteration 1: Pooling participant 1 uploads 10 tickets. Two pools are created to hold one block of 5 tickets each to create Pools 1 and 2, which are consolidation pools, as illustrated in Table 6:

```
1 1 1
```

Table 6
The "1" indicates that pooling participant 1 is participating in both Pools 1 and 2.

Iterations 2 and 3: Pooling participant 2 adds 5 tickets, and pooling participant 3 adds 15 tickets. Pooling participant 2's ticket go into the oldest unfilled pool (Pool 1), and pooling participant 3's tickets go into both pools and into a new Pool 3, which is flagged as a consolidation pool, as shown in Table 7:

<table>
<thead>
<tr>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>Pool 1</td>
</tr>
</tbody>
</table>

Table 7

Iterations 4-8: (1) pooling participants 4-7 add 5 tickets each; and (2) pooling participant 1 adds 10 more tickets. The tickets for pooling participants 4-7 go into the oldest unfilled (non-max filled) pool (Pool 1). Pooling participant 1 adds tickets to Pool 3, and a new Pool 4 is created for his remaining tickets, as shown in Table 8:

<table>
<thead>
<tr>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>Pool 1</td>
</tr>
</tbody>
</table>

Table 8

Pool 4 is not marked as a consolidation pool because Pools 1-3 are consolidation pools, and pool 4 has not reached the consolidation percentage.

Iterations 9-18: Pooling participants 8-17 add tickets in the following amounts, respectively: 5, 5, 10, 5, 15, 10, 5, 5, 5, 10. The initial tickets go into the oldest unfilled (non-max filled) pools (Pool 1). Those that have 10 tickets get a batch added to Pool 2, and pooling participant 12 has a batch placed in Pool 3, as depicted in Table 9.
Pool 4 is not marked as a consolidation pool because Pools 1-3 are still consolidation pools, and pool 4 has not reached the consolidation percentage.

Iteration 19: Pooling participant 18 adds 20 tickets. Since the consolidation threshold is 85% and Pool 1 is filled to 85%, pooling participant 19's first batch of tickets goes into Pool 2. His second and third batches go into Pools 3 and 4, respectively. His last batch of tickets creates Pool 5.
Pools 4 and 5 are not marked as consolidation pools because Pools 1-3 are consolidation pools, and Pools 4 and 5 have not reached the consolidation percentage.

Iterations 20-35: Pooling participants 19-34 add tickets in the following amounts, respectively: 10, 10, 15, 15, 20, 15, 15, 15, 20, 20, 15, 10, 10, 15, 15. Pools 2 and 3 are filled to consolidation percentage, as presented in Table 11:

Iteration 36: Pooling participant 35 adds 15 tickets. One batch is added to each of Pools 4-6, as shown in Table 12.
Pool 4 gets filled to 85% capacity. Since this is the consolidation percentage, Pool 4 is now marked as a consolidation pool, and Pool 1 is unmarked as a consolidation pool, i.e., Pool 1 is no longer marked as a consolidation pool.

Iterations 37-40: Pooling participant 36 adds 10 tickets, and pooling participants 37, 38, and 39 each add 5 tickets. Since Pool 1 is not a consolidation pool, i.e., it is a regular pool, and is not at 100%, i.e., it is unfilled, pooling participant 36’s first batch of tickets is placed into Pool 1. His second group of tickets goes to Pool 5. Pooling participants 37 and 38 fill Pool 1 to 100%, so pooling participant 39 has his tickets added to Pool 5, as illustrated in Table 13.
Pools 2-4 are marked as consolidation pools, and Pools 5 and 6 are not marked as consolidation pools because and Pools 5 and 6 have not reached the consolidation percentage.

Pooling is thereby complete in this example.

**EXAMPLE 6 - FIRST CONSOLIDATION EXAMPLE**

An example of consolidation in accordance with multiple iterations of the method 700 is now described. It is to be understood that the exemplary steps described below are performed by the computer system 260, as described with reference to a method 700.

The following parameters are assumed: (1) consolidation pools: 3; (2) consolidation percentage: 85%; (3) minimum fill percentage: 85%; (4) pool maximum: 100%; and (5) overflow percentage: 15%. In this example, the pools are structured as shown in Table 13. Thus, the consolidation pools are Pools 2-4.

Step 710: Tickets in Pool 5 are moved into the consolidation pools to fill them up to the pool maximum (100%). The batches from pooling participants 35, 36, and 39 are moved to Pool 2, and the batches from pooling participants 32, 33, and 34 are moved to Pool 3, as shown in Table 14.
The batch from pooling participant 31 is moved to Pool 4. Since Pool 4 already has tickets for pooling participant 31, these tickets are swapped with eligible tickets in the newest eligible consolidation pool (Pool 2). Pool 3 is not used since it also has tickets for pooling participant 31. Thus, the tickets for pooling participant 31 are moved to Pool 2 from Pool 4, and the tickets for pooling participant 39 are moved to Pool 4 from Pool 2. The tickets for pooling participant 39 are chosen because they were the newest tickets of a pooling participant not already in Pool 4. Similarly, the tickets of pooling participant 30 are moved to Pool 4 from Pool 5, and then swapped with the tickets of pooling participant 36 in Pool 2. And similarly, the tickets of pooling participant 29 are moved to Pool 4 from Pool 5, and then swapped with the tickets of pooling participant 20 in Pool 2 in Pool 2.

The pools now appear as follows:

<table>
<thead>
<tr>
<th>Pool 1</th>
<th>Pool 2</th>
<th>Pool 3</th>
<th>Pool 4</th>
<th>Pool 5</th>
<th>Pool 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>26</td>
<td>29</td>
<td>33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>25</td>
<td>28</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>24</td>
<td>27</td>
<td>31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>23</td>
<td>26</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>22</td>
<td>25</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>21</td>
<td>24</td>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>20</td>
<td>23</td>
<td>27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>22</td>
<td>26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>21</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>17</td>
<td>20</td>
<td>24</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>19</td>
<td>23</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>18</td>
<td>22</td>
<td>29</td>
<td>35</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>12</td>
<td>21</td>
<td>28</td>
<td>34</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td>23</td>
<td>33</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td>29</td>
</tr>
</tbody>
</table>

Table 14
Table 15

<table>
<thead>
<tr>
<th>Pool 1</th>
<th>Pool 2</th>
<th>Pool 3</th>
<th>Pool 4</th>
<th>Pool 5</th>
<th>Pool 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>25</td>
<td>27</td>
<td>31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>24</td>
<td>26</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>23</td>
<td>25</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>22</td>
<td>24</td>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>21</td>
<td>23</td>
<td>27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>22</td>
<td>26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>21</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>17</td>
<td>20</td>
<td>24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>19</td>
<td>23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>18</td>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>12</td>
<td>21</td>
<td>28</td>
<td>34</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td>23</td>
<td>33</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td>29</td>
</tr>
</tbody>
</table>

All consolidation pools are filled to the pool maximum, as shown in Table 15. Thus, Step 710 is complete.

Step 720: All tickets from Pool 6 are moved to Pool 5. The pools now appear as follows:

<table>
<thead>
<tr>
<th>Pool 1</th>
<th>Pool 2</th>
<th>Pool 3</th>
<th>Pool 4</th>
<th>Pool 5</th>
<th>Pool 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>29</td>
<td>34</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>30</td>
<td>33</td>
<td>36</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>31</td>
<td>32</td>
<td>39</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>35</td>
<td>31</td>
<td>35</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>28</td>
<td>30</td>
<td>34</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>27</td>
<td>29</td>
<td>33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>26</td>
<td>28</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>25</td>
<td>27</td>
<td>31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>24</td>
<td>26</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>23</td>
<td>25</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>22</td>
<td>24</td>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>21</td>
<td>23</td>
<td>27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>22</td>
<td>26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>21</td>
<td>25</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>17</td>
<td>20</td>
<td>24</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>19</td>
<td>23</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>18</td>
<td>22</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>12</td>
<td>21</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td>23</td>
<td></td>
</tr>
</tbody>
</table>
Because there is only one unfilled (non-max filled) pool remaining, Step 720 is complete.

Step 730: Skipped because all consolidation pools are filled.

Step 740: Skipped because only one unfilled (non-max filled) pool remains.

Total of unfilled tickets (number of tickets in unfilled pools) is 35. Because this is less than the minimum fill percentage (85%), go to Step 747.

Step 747: Total overflow capacity is 60 tickets (15 in each of four filled pools). Because this is greater than the total of the unfilled tickets, go to Step 750A.

Step 750A: All tickets in Pool 5 are moved into filled pools, starting with the newest filled pool and proceeding to the oldest filled pool, one batch at a time. The tickets are moved as follows:

(1) Batch 35 is moved to Pool 4. Because Pool 4 already has tickets for pooling participant 35, these tickets are swapped with eligible tickets in the newest eligible consolidation pool (Pool 3). Thus, batch 35 from Pool 4 is swapped with batch 19 in Pool 3.

(2) Batch 34 is moved to Pool 3 because batches are distributed evenly among the filled pools. Batch 34 is then swapped with batch 19 from Pool 2. Note that tickets cannot be swapped with Pool 4 because Pool 4 also has tickets for pooling participant 34.

(3) Batch 33 is moved to Pool 2.

(4) Batch 29 is moved to Pool 1.

The result is as follows:
Because all filed pools now have one extra ticket batch, a second batch is added, starting with the newest filled pool:

(1) Batch 28 is moved to Pool 4, then swapped with batch 38 from Pool 1.

(2) Batch 23 is moved to Pool 3, then swapped with batch 37 from Pool 1.

(3) Batch 18 is moved to Pool 2, then swapped with batch 36 from Pool 1.

The result is shown below:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>29</td>
<td>33</td>
<td>19</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>34</td>
<td>35</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>29</td>
<td>34</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>30</td>
<td>33</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>31</td>
<td>32</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>35</td>
<td>31</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>28</td>
<td>30</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>27</td>
<td>29</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>26</td>
<td>28</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>25</td>
<td>27</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>24</td>
<td>26</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>23</td>
<td>25</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>22</td>
<td>24</td>
<td>27</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>21</td>
<td>23</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>22</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>17</td>
<td>21</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>20</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>18</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>12</td>
<td>21</td>
<td>28</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td>23</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>18</td>
</tr>
</tbody>
</table>

Table 17
Consolidation is thereby complete in this example.

EXAMPLE 7 - SECOND CONSOLIDATION EXAMPLE

Another example of consolidation in accordance with multiple iterations of the method 700 is now described. It is to be understood that the exemplary steps described below are performed by the computer system 260, as described with reference to the method 700.

The following parameters are assumed: (1) consolidation pools: 4; (2) consolidation percentage: 85%; (3) minimum fill percentage: 85%; (4) overflow percentage: 15%; and (5) pool maximum: 100. In this example, the pools are structured as shown below in Table 19:

<table>
<thead>
<tr>
<th>Pool 1</th>
<th>Pool 2</th>
<th>Pool 3</th>
<th>Pool 4</th>
<th>Pool 5</th>
<th>Pool 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>35</td>
<td>31</td>
<td>34</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>28</td>
<td>30</td>
<td>33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>27</td>
<td>29</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>26</td>
<td>28</td>
<td>31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>25</td>
<td>27</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>24</td>
<td>26</td>
<td>29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>23</td>
<td>25</td>
<td>28</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>22</td>
<td>24</td>
<td>27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>21</td>
<td>23</td>
<td>26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>22</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>17</td>
<td>21</td>
<td>24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>20</td>
<td>23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>18</td>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>12</td>
<td>21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 18
The consolidation pools are Pools 2-5.

Step 710: Tickets from Pool 6 are moved into the consolidation pools to fill them to the pool maximum (100%), starting with the oldest consolidation pool:

Batches 30 and 29 are moved to Pool 2.

Batch 28 is moved to Pool 2, then swapped with batch 38 from Pool 1.

Batch 23 is moved to Pool 3, then swapped with batch 37 from Pool 1.

Batch 18 is moved to Pool 3, then swapped with batch 36 from Pool 1.

Batch 1 cannot be moved because that pooling participant has tickets in every filled and consolidation pool.

The batches are thus distributed as follows:
Tickets from Pool 7 and 8 are be moved into the consolidation pools to fill them to the pool maximum:

Batch 30 from Pool 7 is moved to Pool 3, then swapped with batch 17 from Pool 1.

Batch 29 from Pool 7 is moved to Pool 4, then swapped with batch 16 from Pool 1.

Batch 28 from Pool 7 cannot be moved because that pooling participant has tickets in every filled and consolidation pool.

Batch 1 from Pool 7 cannot be moved because that pooling participant has tickets in every filled and consolidation pool.

Batch 30 from Pool 8 cannot be moved because that pooling participant has tickets in every filled and consolidation pool.

Batch 1 from Pool 8 cannot be moved because that pooling participant has tickets in every filled and consolidation pool.
The pools appear as follows:

![Table 21]

All tickets that could have been moved into consolidation pools according to the rules have been moved. Thus, the Step 710 is complete.

Step 720: All tickets are moved into Pool 6, if possible. Batch 28 is moved from Pool 7 is moved to Pool 6. Batch 30 from Pool 8 is moved to Pool 6. Batch 1 from either Pool 7 or 8 cannot be moved to Pool 6 since that pool already has a batch of tickets for pooling participant 1. All tickets that could have been moved into unfilled pools according to the rules have been moved. Thus, Step 720 is complete.

Step 730: Step 710 is repeated, but this time 2 batches for a pooling participant will be allowed within the same pool. Batch 1 from Pool 6 is moved to Pool 4. Because 2 batches are allowed, no swap is necessary. Batch 28 from Pool 7 is moved to Pool 4. Because 2 batches are allowed, no swap is necessary. Batch 1 from Pool 7 is moved to Pool 5. Because 2
batches are allowed, no swap is necessary. Batch 30 from Pool 7 is moved to Pool 5. Because 2 batches are allowed, no swap is necessary. Batch 1 from Pool 7 is moved to Pool 5. Because there are already 2 batches for this member in Pool 5, a swap is necessary with a pool that only has one batch for this member. Thus, batch 1 is swapped with batch 37 from Pool 3. The result is shown below:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>29</td>
<td>38</td>
<td>17</td>
<td>28</td>
<td>37</td>
</tr>
<tr>
<td>30</td>
<td>29</td>
<td>36</td>
<td>1</td>
<td>36</td>
</tr>
<tr>
<td>18</td>
<td>30</td>
<td>1</td>
<td>16</td>
<td>1</td>
</tr>
<tr>
<td>23</td>
<td>28</td>
<td>31</td>
<td>35</td>
<td>35</td>
</tr>
<tr>
<td>28</td>
<td>27</td>
<td>30</td>
<td>34</td>
<td>34</td>
</tr>
<tr>
<td>15</td>
<td>26</td>
<td>29</td>
<td>33</td>
<td>33</td>
</tr>
<tr>
<td>14</td>
<td>25</td>
<td>28</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>13</td>
<td>24</td>
<td>27</td>
<td>31</td>
<td>31</td>
</tr>
<tr>
<td>12</td>
<td>23</td>
<td>26</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>11</td>
<td>22</td>
<td>25</td>
<td>29</td>
<td>29</td>
</tr>
<tr>
<td>10</td>
<td>21</td>
<td>24</td>
<td>28</td>
<td>28</td>
</tr>
<tr>
<td>9</td>
<td>20</td>
<td>23</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>22</td>
<td>26</td>
<td>26</td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>21</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>6</td>
<td>17</td>
<td>20</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>19</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>18</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>12</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 22

[00246] Because all pools are filled, no further steps are necessary. Consolidation is thereby complete in this example.

EXAMPLE 7 - THIRD CONSOLIDATION EXAMPLE

[00247] Yet another example of consolidation in accordance with multiple iterations of the method 700 is now described. It is to be understood that the exemplary steps described below are performed by the computer system 260, as described with reference to the method 700.
The following parameters are assumed: (1) consolidation pools: 3; (2) consolidation percentage: 90%; (3) minimum fill percentage: 85%; (4) overflow percentage: 15%; and (5) pool maximum: 100. In this example, the pools are structured as shown below in Table 23.

<table>
<thead>
<tr>
<th></th>
<th>Pool 1</th>
<th>Pool 2</th>
<th>Pool 3</th>
<th>Pool 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>36</td>
<td>17</td>
<td>16</td>
<td>15</td>
<td>14</td>
</tr>
<tr>
<td>37</td>
<td>28</td>
<td>27</td>
<td>26</td>
<td>25</td>
</tr>
<tr>
<td>38</td>
<td>31</td>
<td>30</td>
<td>29</td>
<td>28</td>
</tr>
<tr>
<td>35</td>
<td>34</td>
<td>33</td>
<td>32</td>
<td>31</td>
</tr>
<tr>
<td>30</td>
<td>29</td>
<td>28</td>
<td>27</td>
<td>26</td>
</tr>
<tr>
<td>28</td>
<td>27</td>
<td>26</td>
<td>25</td>
<td>24</td>
</tr>
<tr>
<td>23</td>
<td>22</td>
<td>22</td>
<td>21</td>
<td>20</td>
</tr>
<tr>
<td>27</td>
<td>21</td>
<td>20</td>
<td>19</td>
<td>18</td>
</tr>
<tr>
<td>26</td>
<td>20</td>
<td>19</td>
<td>18</td>
<td>17</td>
</tr>
<tr>
<td>25</td>
<td>18</td>
<td>18</td>
<td>17</td>
<td>16</td>
</tr>
<tr>
<td>24</td>
<td>17</td>
<td>17</td>
<td>16</td>
<td>15</td>
</tr>
<tr>
<td>23</td>
<td>16</td>
<td>16</td>
<td>15</td>
<td>14</td>
</tr>
<tr>
<td>18</td>
<td>15</td>
<td>15</td>
<td>14</td>
<td>13</td>
</tr>
<tr>
<td>17</td>
<td>12</td>
<td>12</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>16</td>
<td>11</td>
<td>11</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>15</td>
<td>10</td>
<td>10</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>14</td>
<td>9</td>
<td>9</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>13</td>
<td>8</td>
<td>8</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>12</td>
<td>7</td>
<td>7</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>11</td>
<td>6</td>
<td>6</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 23

The consolidation pools are Pools 1-3.

Step 710: Batches from Pool 4 will be moved into the consolidation pools to fill them to the pool maximum (100%), starting with the oldest consolidation pool. Batches 35 and 34 are moved to Pool 1; batches 33 and 32 are moved to Pool 2; batch 31 is moved to Pool 2, then swapped with batch 37 from Pool 3; and batch 30 is moved to Pool 2, the swapped with batch 33 from Pool.
Step 720: Skipped. Only one unfilled (non-max filled) pool remains.

Step 730: Skipped. All consolidation pools are filled.

Step 740: Skipped. Only one unfilled (non-max filled) pool remains.

The total unfilled tickets are 55. Because this is less than the minimum fill percentage (85%), go to Step 747.

Step 747: Calculate the total overflow capacity: 45%. Because this is less than the total unfilled tickets, go to Step 750B.

Step 750B: Tickets are moved back into Pool 4 to try to make sure all pools are above or at the minimum fill percentage. Tickets are moved from the newest filled pool first, then the next successive newest filled pool, and so on, one batch at a time:

Batch 38 is moved from Pool 3 to Pool 4.

Batch 32 is moved from Pool 2 to Pool 4.

Batch 36 is moved from Pool 1 to Pool 4.
[00260] All filled pools have had one batch moved. Thus, the process goes back to the newest filled pool and moves to a second batch: Batch 37 is moved from Pool 3 to Pool 4.

[00261] Batch 31 is moved from Pool 2 to Pool 4.

[00262] Batch 35 is moved from Pool 1 to Pool 4.

[00263] Because Pool 4 is now at the minimum fill percentage (85%), no more tickets need to be moved.

[00264] The pools look similar to their arrangement at the start of consolidation:

<table>
<thead>
<tr>
<th></th>
<th>Pool 1</th>
<th>Pool 2</th>
<th>Pool 3</th>
<th>Pool 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>34</td>
<td>30</td>
<td>33</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>28</td>
<td>31</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>27</td>
<td>30</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>26</td>
<td>29</td>
<td>37</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>25</td>
<td>28</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>24</td>
<td>27</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>23</td>
<td>26</td>
<td>38</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>22</td>
<td>25</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>21</td>
<td>24</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>20</td>
<td>23</td>
<td>27</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>22</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>21</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>17</td>
<td>20</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>19</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>18</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>12</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Table 25

[00265] In this example, Pools 1-3 are filled to the consolidation percentage, and Pool 4 is filled to the minimum fill percentage. Thus, no further steps are necessary. Consolidation is thereby complete in this example.

[00266] These and other advantages of the present invention will be apparent to those skilled in the art from the foregoing specification. Accordingly, it is to be recognized by those
skilled in the art that changes or modifications may be made to the above-described embodiments without departing from the broad inventive concepts of the invention. It is to be understood that this invention is not limited to the particular embodiments described herein, but is intended to include all changes and modifications that are within the scope and spirit of the invention.
What is claimed is:

1. A method for pooling lottery tickets comprising:
   receiving information identifying one or more lottery tickets;
   receiving a request to pool the one or more lottery tickets from a pooling participant;
   associating the one or more lottery tickets with one or more pools based on the request,
   predefined pooling rules, and predefined pooling parameters for each of the one or more pools;
   and
   transmitting an indication of the one or more pools with which the one or more lottery
   tickets are associated.

2. The method of claim 1, wherein the request comprises an identification of a type of pool
   into which the one or more lottery tickets are to be pooled.

3. The method of claim 2, wherein the predefined pooling rules proscribe pooling the one or
   more lottery tickets in a pool in which the pooling participant has previously pooled a lottery
   ticket, and wherein the step of associating comprises associating the one or more lottery tickets
   with one or more pools corresponding to the type of pool identified in the request if the pooling
   participant is not already associated with a lottery ticket in the one or more pools.
4. The method of claim 2, wherein the type of pool comprises a maximum pool size and one or more payout rules.

5. The method of claim 2, wherein the step of associating comprises:

   creating one or more new pools pursuant to the received request if the type of pool does not exist; and

   associating the one or more lottery tickets with the one or more new pools based on the request, predefined pooling rules, and predefined pooling parameters for each of the one or more new pools.

6. The method of claim 1, wherein the predefined pooling parameters comprise a maximum size of each of the one or more pools, and wherein the step of associating comprises associating the one or more lottery tickets with the one or more pools if the one or more pools are not filled to the maximum size.

7. The method of claim 1, wherein the predefined pooling rules comprise rules specifying which of the one or more pools are consolidation pools and which of the one or more pools are non-consolidation pools, and wherein the step of associating comprises associating the one or more lottery tickets with one or more of the consolidation pools depending on the predefined pooling rules.

8. The method of claim 1, wherein the predefined pooling rules comprise rules specifying
which of the one or more pools are consolidation pools and which of the one or more pools are non-consolidation pools, and wherein the step of associating comprises associating the one or more lottery tickets with one or more of the non-consolidation pools depending on the predefined pooling rules.

9. The method of claim 1, wherein the predefined pooling rules comprise rules specifying a maximum fill percentage of each of the one or more pools and a consolidation percentage of each of the one or more pools, and wherein the step of associating comprises associating the one or more lottery tickets with a pool that is below its maximum fill percentage and that is not marked as a consolidation pool or with a pool that is below its consolidation percentage and that is marked as a consolidation pool.

10. The method of claim 9, further comprising marking a pool as a consolidation pool after reaching the consolidation percentage after the step of associating.

11. The method of claim 1, wherein the one or more lottery tickets are only active when a jackpot for a specified lottery is equal to or greater than a predetermined threshold.

12. The method of claim 1, further comprising a step of validating the one or more lottery tickets corresponding to the received information as not fraudulent if the information is received within a predetermined period of time after the one or more lottery tickets were purchased.
13. A method for consolidating a plurality of lottery tickets associated with a plurality of pools prior to a lottery drawing, the method comprising:

(1) moving one or more tickets associated with an unfilled pool to a consolidation pool, using limited swapping as needed;

(2) moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using limited swapping as needed;

(3) moving one or more tickets associated with an unfilled pool to a consolidation pool, using unlimited swapping as needed; and

(4) moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using unlimited swapping as needed.

14. The method of claim 13, wherein the step (1) comprises sub-steps:

(a) selecting an oldest pool that is not filled to the pool maximum;

(b) selecting an oldest consolidation pool that is not filled to the pool maximum;

(c) moving each batch of tickets from the oldest pool selected in sub-step (a) to the oldest consolidation pool selected in sub-step (b); and

(d) if a pooling participant associated with a ticket moved in the sub-step (c) to the oldest consolidation pool selected in sub-step (b) is associated with a ticket already in the oldest consolidation pool selected in sub-step (b), using limited swapping.
15. The method of claim 15, wherein the step (1) further comprises a sub-step:

(e) repeating sub-steps (a) through (d), replacing limited swapping in the sub-step (d) with unlimited swapping.

16. The method of claim 13, wherein the step (2) comprises sub-steps:

(a) selecting an oldest pool that is not filled to the pool maximum;
(b) selecting a next oldest pool that is not filled to the pool maximum;
(c) moving each batch of tickets from the next oldest pool selected in sub-step (b) to the oldest pool selected in sub-step (a); and
(d) if a pooling participant associated with a ticket moved in the sub-step (c) to the oldest pool selected in sub-step (b) is associated with a ticket already in the oldest pool selected in sub-step (b), using limited swapping.

17. The method of claim 16, wherein the step (2) further comprises a sub-step:

(e) repeating sub-steps (a) through (d), replacing limited swapping in the sub-step (d) with unlimited swapping.

18. A method for pooling and consolidating lottery tickets comprising:

receiving information identifying one or more lottery tickets;
receiving a request to pool the one or more lottery tickets from a pooling participant;
associating the one or more lottery tickets with one or more pools based on the request, predefined pooling rules, and predefined pooling parameters for each of the one or more pools;
transmitting an indication of the one or more pools with which the one or more lottery tickets are associated; and

prior to a lottery drawing, performing sub-steps of:

(1) moving one or more tickets associated with an unfilled pool to a consolidation pool, using limited swapping as needed;

(2) moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using limited swapping as needed;

(3) moving one or more tickets associated with an unfilled pool to a consolidation pool, using unlimited swapping as needed; and

(4) moving one or more tickets from a next oldest unfilled pool to an oldest unfilled pool, using unlimited swapping as needed.

19. The method of claim 18, wherein the request comprises an identification of a type of pool into which the one or more lottery tickets are to be pooled.

20. The method of claim 18, wherein the one or more lottery tickets are only active when a jackpot for a specified lottery is equal to or greater than a predetermined threshold
FIG. 2
Request account

A

315

Account created?

NO

YES

320

Receive account confirmation and information

Purchase lottery ticket(s) from vendor

Access pooling interface portal and log in

300

Scan and transmit lottery ticket

B

350

Fraud prevention

355

Receive results of validation and pooling options

Validated?

NO

YES

365

Select pooling

C

370

Receive pooling confirmation

380

390

End

FIG. 3
Receive batch of lottery tickets and pool type selection

Does active pool of specified type of pool exist?

Find and select oldest pool

Is this pool filed to maximum?

Add tickets to selected oldest pool

Has pool reached consolidation %?

End

Create pool and place batch of tickets into pool

Number of consolidation pools < N?

Mark new pool as consolidation pool

Does pool have tickets in pool?

Does another active pool exist?

Find and select next oldest pool

FIG. 6A
FIG. 6B
Move tickets to consolidation pools using limited swapping as needed.

Tickets in unfilled, non-consolidation pools?

NO

Move tickets to unfilled pools using limited swapping as needed.

Tickets in unfilled, non-consolidation pools?

NO

Repeat Step 710 but use unlimited swapping as needed

Tickets in unfilled, non-consolidation pools?

YES

End

FIG. 7A
Repeat Step 720 but use unlimited swapping as needed

Tickets in unfilled, non-consolidation pools above min fill %?

Yes

Calculate total overflow capacity

Total overflow capacity >= Total unfilled tickets?

No

Move tickets from remaining unfilled pool to filled pools using unlimited swapping as needed

End

Yes

Move tickets from newest filled pools to the remaining unfilled pool using unlimited swapping as needed

FIG. 7B
Welcome Mike!
Current Drawing: $640,000,000
Pool Size: 100,000

Chances at $20,000,000: 100

Chances at $610,611: 99,999

FIG. 8A

Welcome Mike!
Current Drawing: $640,000,000
Pool Size: 100,000

Chances at $20,000,000: 100

Chances at $610,611: 99,990

FIG. 8B
FIG. 8C

FIG. 8D
Welcome Mike!
Current Drawing: $640,000,000
Pool Size: 100,000

Chance at $20,000,000:
22

Chances at $610,611:
5,007

840

FIG. 8E

Welcome Mike!
Current Drawing: $640,000,000
Pool Size: 100,000

Chance at $30,000,000:
12

Chances at $610,611:
327

840

FIG. 8F
Sorry,
but your Pool did not win this time.

But please keep these tickets.
You have won $150!

FIG. 81
Lottery

###

Date/Time

$\text{Threshold}$

123-456-789
INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER

IPC(8) - A63F 3/06 (2012.01)
USPC - 463/17

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC(8) - A63F 3/06; G07C 15/00 (2012.01)
USPC - 463/17, 25

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

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

PatBase, Google Patents

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>US 2007/0077983 A1 (AMADA et al) 05 April 2007 (05.04.2007) entire document</td>
<td>1-6, 11</td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td>7-10, 12, 18-20</td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td>7-10, 18-20</td>
</tr>
</tbody>
</table>

Further documents are listed in the continuation of Box C.

* Special categories of cited documents:
  'A' document defining the general state of the art which is not considered to be of particular relevance
  'E' earlier application or patent but published on or after the international filing date
  'L' document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
  'O' document referring to an oral disclosure, use, exhibition or other means
  'P' document published prior to the international filing date but later than the priority date claimed

** T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

** X document showing a teaching in the same field of technology and, in the case of prior applications, published in the same country as a result of national stage of nationalization procedure

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

Date of the actual completion of the international search: 13 December 2012

Date of mailing of the international search report: 27 DEC 2012

Name and mailing address of the ISA/US

Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, Virginia 22313-1450
Facsimile No. 571-273-3201

Authorized officer: Blaine R. Copenheaver

Form PCT/ISA/2 10 (second sheet) (July 2009)