



- (51) International Patent Classification: Not classified
- (21) International Application Number: PCT/US2016/027288
- (22) International Filing Date: 13 April 2016 (13.04.2016)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data: 14/686,381 14 April 2015 (14.04.2015) US
- (71) Applicant: SQUARE, INC. [US/US]; 1455 Market St., Suite 600, San Francisco, CA 94103 (US).
- (72) Inventors: RENKE, Christopher, Philip; c/o Square, Inc., 1455 Market St., Suite 600, San Francisco, CA 94103 (US). WHITE, Michael, Wells; c/o Square, Inc., 1455 Market St., Suite 600, San Francisco, CA 94103 (US). MULLER, Eric, Dickeson; c/o Square, Inc., 1455 Market St., Suite 600, San Francisco, CA 94103 (US). WILSON, Mathew; c/o Square, Inc., 1455 Market St., Suite 600, San Francisco, CA 94103 (US).
- (74) Agents: CUNNINGHAM, Aaron et al.; Lee & Hayes, PLLC, 601 W. Riverside Ave, Suite 1400, Spokane, WA 99201 (US).

- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:
— without international search report and to be republished upon receipt of that report (Rule 48.2(g))

(54) Title: OPEN TICKET PAYMENT HANDLING WITH OFFLINE MODE

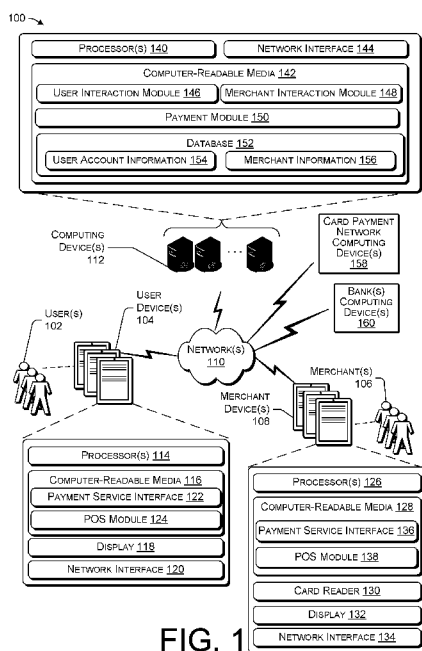


FIG. 1

(57) Abstract: A payment handling system may operate to handle payments for the cost of a transaction in which one or more devices operate in an off-line mode. The payment handling system may allow for merchant devices to make modifications to local versions of an open ticket while in an off-line mode. When a merchant device subsequently enters an online mode, the systems and techniques may operate to reconcile differences between the local versions of the open ticket.



OPEN TICKET PAYMENT HANDLING WITH OFFLINE MODE

RELATED APPLICATIONS

5 [0001] The present application claims priority to U.S. Patent Application No. 14/686,381 filed on April 14, 2015, entitled "Open Ticket Payment Handling with Offline Mode," which is incorporated by reference herein in its entirety.

BACKGROUND

10 [0002] In today's commerce, merchants often utilize an array of different point-of-sale (POS) devices, including mobile POS devices. Merchants may use these mobile POS devices to engage in transactions with customers at different locations. For instance, a waiter of a restaurant may use a mobile POS device to take and process customers' orders for food and drinks. In another example, a bartender may use a mobile POS device to charge a customer for items purchased over the course of a visit by the customer.

15 [0003] In some instances, the customers may wish to request items from the merchant over the span of a visit without paying multiple times. In such instances, the ticket for the transaction may be maintained and updated each time the customers request additional items. However, some customers may request additional items from a different waiter if the original waiter is not present. Such an occurrence may result in issues that may cause errors, delay, inconvenience, poor customer experience and loss of revenue. For example, requesting items from the different waiter may result in items being left off of the final ticket (e.g. the different waiter forgets to have the original waiter update the ticket), duplicate items may be delivered to the customer or the customer may be double charged an item, the customers may not receive the additional items requested from the different waiter (e.g. the different waiter becomes distracted by other customers and forgets to the cause the update to the ticket) and so on.

BRIEF DESCRIPTION OF THE DRAWINGS

25 [0004] The detailed description is set forth with reference to the accompanying figures, in which the left-most digit of a reference number identifies the figure in which the reference number first appears. The use of the same reference numbers in the same or different figures indicates similar or identical items or features.

[0005] FIG. 1 illustrates an example system for handling payments among customers and merchants.

30 [0006] FIGS. 2A-2B are example illustrations of two scenarios that provide context for FIGS. 3A-4C.

[0007] FIGS. 3A-3B are sequence diagrams illustrating the operations of the offline mode functionality executed by the system shown in FIG. 1 in the context of FIG. 2A.

[0008] FIGS. 4A-4C are sequence diagrams illustrating the operations of the offline mode functionality executed by the system shown in FIG. 1 in the context of FIG. 2B.

5 [0009] FIG. 5 is a flow diagram illustrating an example process for handling open ticket transactions including offline mode functionality in the system shown in FIG. 1 and as discuss with regard to FIGS. 2A-4C.

DETAILED DESCRIPTION

[0010] This disclosure describes systems and processes that provide for operation of merchant
10 devices in an off-line mode when handling payments among customers and merchants for open tickets. In some examples herein, the systems and techniques may provide for merchant devices to make modifications to local versions of an open ticket while in an off-line mode. When the merchant device subsequently enters an online mode, the systems and techniques may resolve differences between the versions of the open ticket. The functionality for providing off-line mode operation when handling payments may be implemented in a
15 variety of ways, examples of which are discussed below.

[0011] In an example implementation utilized by a restaurant, a waiter of the restaurant may input a request to a merchant device to create an open ticket for a transaction with customers at a table. The merchant device may create the open ticket or request that a payment service create the open ticket. After the ticket is created, the merchant may input information for the transaction with the customers. For
20 example, the waiter may enter initial food items requested by the customers and/or one or payment instruments to be associated with the open ticket. Subsequently, copies of the open ticket are distributed to other merchant devices in the restaurant. Depending on the implementation, the distribution of the open ticket among the other merchant devices may be performed via a payment service and/or by peer-to-peer communications (e.g. using a local area network (LAN) or an ad hoc network).

[0012] When a request for additional items is entered into one of the merchant devices or other
25 changes are made to the open ticket by one of the merchant devices, the changes may be propagated and reconciled among the local versions of the open ticket. However, when a change is made to a local version of a merchant device in an off-line mode, the changes may not be propagated until after changes are made to other local versions. Similarly, an offline device may not receive changes to an open ticket made by other
30 devices. As a result, when the device enters an online mode, the system may reconcile the changes to ascertain an actual state of the open ticket transaction. Example ways in which such reconciliation may be performed are provided below with regard to FIGS. 3A-5.

[0013] Throughout this disclosure, reference is made to devices operating in online mode and offline mode. Whether a device is considered to be in an offline mode or an online mode may vary depending on the implementation. In some examples, offline mode refers to a mode in which a device is unable to connect to or communicate with at least one other device (e.g. offline with respect to the other device with which the device is not connected or able to communicate). In some examples, the offline mode may be total or selective. Some examples of a device in an offline mode are a device which is out of range of a radio base station, a device that is unplugged from a hardline network, a device that has been intentionally put into an offline or "airplane" mode, a device that is part of an ad-hoc network that is experiencing a temporary network partition through a connectivity issue of another device in the ad-hoc network, and so on. In some examples, a device is in an online mode when it is able to communicate or connect with at least one other device. Similarly to an offline mode, the online mode may be total or selective. Some examples of a device being in an online mode include a device that can communicate with another device in an ad hoc or peer-to-peer network, a device that can communicate with at least some devices in a LAN, a device that can communicate with all devices in a LAN, and/or a device that can communicate with a payment service system, and so on.

[0014] At the end of the open ticket transaction (i.e. the customers leave or indicate they are ready to leave), the waiter may input a close ticket request. In response, the system may perform a final reconciliation of the local versions and execute the payment transaction.

[0015] Additional details and scenarios are described below. This brief introduction is provided for the reader's convenience and is not intended to limit the scope of the claims. Furthermore, the techniques described above and below may be implemented in a number of ways and in a number of contexts. Several example implementations and contexts are provided with reference to the following figures, as described below in more detail. However, the following implementations and contexts are but a few of many. For example, though discussed herein in the context of a payment handling system, implementations are not so limited.

[0016] FIG. 1 illustrates an example system 100 for handling open ticket transactions among customers and merchants. More particularly, FIG. 1 provides a framework for providing functionality for offline mode operations. Additional details of offline mode operations are provided with respect to FIGS. 2A-5.

[0017] As shown in FIG. 1, the system 100 may include one or more user(s) 102 (e.g. customers), one or more user device(s) 104 associated with the user(s) 102, one or more merchants 106, one or more merchant devices 108 associated with the one or more merchants 106, one or more network(s) 110, and one or more computing device(s) 112. In various implementations, the user(s) 102 may operate the user device(s) 104, which may include one or more processor(s) 114, computer-readable media 116, a display 118 and a network interface 120. The computer-readable media 116 may store a payment service interface 122

and a POS module 124. Similarly, the merchant(s) 106 may operate the merchant device(s) 108, which may include one or more processor(s) 126, computer-readable media 128, a card reader 130, a display 132 and a network interface 134. The computer-readable media 126 may store a payment service interface 136 and a POS module 138. The computing device(s) 112 may also include one or more processor(s) 140, computer-readable media 142 and a network interface 144. The computer readable media 142 may store a user interaction module 146, a merchant interaction module 148, a payment module 150 and a database 152.

[0018] In some implementations, one of the users 102 may operate a user device 104 to perform various functions associated with the user device 104. For example, a user of the user(s) 102 may utilize the user device 104, and particularly the payment service interface 122 thereof, to interact with the computing devices 112 via the network interface 120 to establish a user account with the payment service of the computing devices 112. In addition, a user of the user(s) 102 may utilize POS module 124 of the user device 104 to interface with the POS module 138 of the merchant device(s) 108, e.g. as part of a transaction using the payment service of the computing devices 112. For example, the user device 104 may communicate via the network interface 120 with the merchant device(s) 108 and the network interface 134. As an example of such a payment operation, the POS module 138 of the merchant device 108 may communicate with the POS module 124 of the user device 104 to obtain information for processing a payment from the user 102 to the merchant 106 using the payment service of the computing devices 112.

[0019] In some implementations, the user device 104 may be any type of device that is capable of interacting with the merchant device(s) 108 and/or the computing device(s) 112. For instance, the user device 104 may include a personal computer, a laptop computer, a cellular telephone, a PDA, a tablet device, or any other device. The user device 104 shown in FIG. 1 is only one example of a user device 104 and is not intended to suggest any limitation as to the scope of use or functionality of any user device 104 utilized to perform the processes and/or procedures described herein. For example, the user device 104 may include various other applications or modules, such as a module for a user dashboard to enable the user to control information in a user's profile, set user preferences, and so forth.

[0020] The processor(s) 114 of the user device 104 may execute one or more modules and/or processes to cause the user device 104 to perform a variety of functions, as set forth above and explained in further detail in the following disclosure. In some implementations, the processor(s) 114 may include a central processing unit (CPU), a graphics processing unit (GPU), both CPU and GPU, or other processing units or components known in the art. Additionally, each of the processor(s) 114 may possess its own local memory, which also may store program modules, program data, and/or one or more operating systems.

[0021] Depending on the exact configuration and type of the user device 104, the computer-readable media 116 may include volatile memory (such as RAM), non-volatile memory (such as ROM, flash memory, miniature hard drive, memory card, or the like), or some combination thereof.

[0022] In various implementations, the user device 104 may also have input device(s) such as a keyboard, a mouse, a pen, a voice input device, a touch input device, etc. The user device 104 may also include the display 118 and other output device(s), such as speakers, a printer, etc. The user 102 may utilize the foregoing features to interact with the user device 104, merchant device(s) 108 or the computing device(s) 112 via the network(s) 110. More particularly, the display 118 of the user device 104 may include any type of display 118 known in the art that is configured to present (e.g., display) information to the users 102.

[0023] In various implementations, the one or more merchants 106 may be any individual, entity, or machine that offers products, services or the like according to the examples herein. Moreover, each of the merchants 106 may be associated with one or more merchant devices 108, which may be the same as, similar to, or different from the user devices 104. The merchant devices 108 may include any number of components such as the one or more processor(s) 126, the computer-readable media 128, the card reader 130, the display 132 and/or network interface 134. The merchants 106 may utilize the merchant devices 108 to interact with the user device(s) 104 and/or computing device(s) 112 in any manner. For instance, the merchant devices 108 may be used to access an interface associated with the computing device(s) 112 (e.g. the payment service interface 136). Continuing the above example, a merchant device 108 may utilize information obtained from interacting with the POS module 124 of the user device 104 to execute the payment from the user 102 to the merchant 106 through the payment service of the computing devices 112. Further, the POS module 138 may control the operation of the card reader 130 to read payment information from credit cards, debit cards, gift cards and the like. The POS module 138 may then utilize the information in operations for allocating portions of a transaction cost between a plurality of payment instruments. Moreover, the POS module 138 may operate to interact with the card payment network computing devices(s) 160 and/or bank(s) computing device(s) 160 to execute payments from the user 102 to the merchant 106.

[0024] While the user devices 104 and merchant devices 108 are shown as including different modules, this is merely for ease of illustration and not intended as limiting. In various implementations, the user devices 104 and merchant devices 108 may be identical, similar or distinct. Moreover, the modules shown and described for the user devices 104 and merchant devices 108 may be implemented as more modules or as fewer modules and functions described for the modules may be redistributed depending on the details of the implementation. Further, in some implementations, the user devices 104 and/or merchant devices 108 may vary from device to device. In general, the user devices 104 and the merchant devices 108 can each be any appropriate device operable to send and receive requests, messages, or other types of information over the one or more networks 110 or directly to each other. Additionally, in some implementation, there may be thousands, hundreds of thousands, or more, of the user devices 104 and the merchant devices 108.

[0025] In some implementations, the network(s) 110 may be any type of network known in the art, such as a local area network or a wide area network, such as the Internet, and may include a wireless network, such as a cellular network, a local wireless network, such as Wi-Fi and/or close-range wireless communications, such as Bluetooth® and Bluetooth® low energy, near field communications (NFC), a wired network, or any other such network, or any combination thereof. Accordingly, the one or more networks 110 may include both wired and/or wireless communication technologies, including Bluetooth®, Bluetooth® low energy, Wi-Fi and cellular communication technologies, as well as wired or fiber optic technologies. Components used for such communications can depend at least in part upon the type of network, the environment selected, or both. Protocols for communicating over such networks are well known and will not be discussed herein in detail. Consequently, the user devices 104, the merchant devices 108, and the computing device(s) 112 may communicatively couple to the network(s) 110 in any manner, such as by a wired or wireless connection. The network(s) 110 may also facilitate communication between the user devices 104, the merchant devices 108, and the computing device(s) 112. In turn, the network interfaces 120, 134 and 144 of the user devices 104, the merchant devices 108, and the computing device(s) 112 may be any network interface hardware components that may allow user devices 104, the merchant devices 108, and the computing device(s) 112 communicate over the network(s) 110. For example, in a particular implementation, the network interfaces 120 and 134 of the user devices 104 and merchant devices 108 may include near field communication capabilities for performing the communications there between involved in POS operations.

[0026] In addition, and as mentioned previously, the computing device(s) 112 may include the one or more processor(s) 140, the computer-readable media 142 and network interface 144. The computing device(s) 112 may also include additional components not listed above that may perform any function associated with the computing device(s) 112. In various implementations, the computing device(s) 112 may be any type of computing device, such as a network-accessible server, and may be one of multiple servers included in a server cluster or server farm. In other implementations, the processor(s) 140 and the computer-readable media 142 of the computing device(s) 112 may be the same as, similar to, or different from the processor(s) 114 and the computer-readable media 116, respectively, of the user device(s) 104. As discussed above, the computer-readable media 142 may store the user interaction module 146, the merchant interaction module 148, the payment module 150 and the database 152. The database 152 may store various information including user account information 154 and merchant information 158.

[0027] The user interaction module 146 and merchant interaction module 148 operate to interface with the user devices 104 and merchant devices 108, respectively. For example, the modules 146 and 148 may operate in accordance with instructions from the payment module 150 to request or provide information on behalf of the payment module 150. The payment module 150 may handle the processing of payments. For example, the payment module 150 may utilize the user interaction module 146 and the merchant

interaction module 148 to handle communication with the user 102 and merchant 106, respectively. In addition, the payment module 150 may utilize information from the database 152, such as the user account information 154 and merchant information 158 to provide handling of payments between merchants and users. In some implementations, user account information 154 may include information regarding electronic payment accounts of the customers (e.g. users 102).

[0028] As mentioned above, the payment module 150 may handle payments between merchants and users. When paying for a transaction, a user 102 can provide the amount of payment that is due to a merchant 106 using cash, check, a payment card, NFC, or by electronic payment through a payment service of the computing devices 112. The merchant 106 can interact with the merchant device 108 to process the transaction. In some examples, the service of the computing device 112 may handle some payments while other payments may at least at times be handled by point of sale (POS) transactions. In such cases, the point of sale may be the place where the user 102 with user device 104 interacts with the merchant 106 with merchant device 108 and executes a transaction (e.g. purchases items from a street vendor merchant or a restaurant merchant). During point-of-sale (POS) transactions, the merchant device 108 can determine and send data describing the transactions, including, for example, services provided, item(s) being purchased, the amount of the services or item(s), buyer information, and so forth.

[0029] In some implementations, the payment service enables card-less payments, i.e., electronic payments, for transactions between the users 102 and the merchants 106 based on interaction of the user 102 with the user device 104 and interaction of the merchant 106 with the merchant device 108. Accordingly, in some examples, a card-less payment transaction may include a transaction conducted between a user 102 and a merchant 106 at a POS location during which an electronic payment account of the user 102 is charged without the user 102 having to physically present a payment card to the merchant 106 at the POS location. Consequently, the merchant 106 need not receive any details about the financial account of the user 102 for the transaction to be processed. As one example, the electronic payment may be charged to a credit card issuer or credit card number that the user 102 provided when signing up with the service of the computing devices 112 for an electronic payment account. As another example, the user 102 may have a quantity of money pre-paid in an account maintained for use in making the electronic payments. Other variations will also be apparent to those of skill in the art having the benefit of the disclosure herein.

[0030] Before conducting an electronic payment transaction, the user 102 typically creates a user account with the service of the computing devices 112. The user 102 can create the user account, for example, by interacting with an application of the user device 104 that is configured to perform electronic payment transactions and that may execute on the user device 104 (e.g. the payment service interface 122). When creating an electronic payment account with the service of the computing devices 112, the user 102 may provide an image including the face of the user, data describing a financial account of the user 102 (e.g., a credit card number, expiration date), and a billing address. This user information can be securely stored by

the computing devices 112, for example, in the user account information 154 in the database 152. Further, the user account information 154 may be created for each user 102, which may include information about the user and transactions conducted by the user.

5 [0031] To accept electronic payments for POS transactions, the merchant 106 may create a merchant account with the service of the computing devices 112 by providing information describing the merchant including, for example, a merchant name, contact information, e.g., telephone numbers, the merchant's geographic location address, and one or more financial accounts to which funds collected from users will be deposited. This merchant information 156 can be securely stored by the service, for example, in the database 152 along with the user account information 154. Further, a merchant profile may be created
10 for each merchant, which may include information about the merchant and transactions conducted by the merchant.

[0032] The service of the computing devices 112 may be configured to enable electronic payments for transactions. The computing devices 112 can include one or more servers that are configured to perform secure electronic financial transactions, e.g., electronic payments for transactions between a user and a
15 merchant, for example, through data communicated between the user device 104 and the merchant device 108. Generally, when a user and a merchant enter into an electronic payment transaction, the transaction is processed by electronically transferring funds from a financial account associated with the user account to a financial account associated with the merchant account. Alternatively, the user may have a balance of funds maintained by the payment service as part of the user account which may be used in transactions.

20 [0033] The payment module 150 may be configured to send and receive data to and from the user device 104 and the merchant device 108. For example, the payment module 150 can be configured to send information describing merchants to an application on the user device 104 using, for example, the information stored in the database 152. For example, the payment module 150 can communicate data describing merchants 106 that are within a threshold geographic distance from a geographic location of the
25 user device 104. The data describing the merchants 106 can include, for example, a merchant name, geographic location, contact information, and an electronic catalogue, e.g., a menu that describes items that are available from the merchant.

[0034] In some embodiments, the payment module 150 is configured to determine whether a geographic location of the user device 104 is within a threshold geographic distance from a geographic
30 location of the merchant device 108. The payment module 150 can determine a geographic location of the user device 104 using, for example, geolocation data provided by the user device 104. Similarly, the payment module 150 can determine a geographic location of the merchant device 108 using, for example, geolocation data provided by the merchant device 108 or using a geographic address, e.g., street address, provided by the merchant. Depending on the implementation, the threshold geographic distance can be
35 specified by the payment module 150, by the user, or by the merchant.

[0035] Determining whether the user device 104 is within a threshold geographic distance of the merchant device 108 can be accomplished in different ways including, for example, determining whether the user device 104 is within a threshold geographic radius of the merchant device 108, determining whether the user device 104 is within a particular geofence, or determining whether the user device 104 can communicate with the merchant device 108 using a specified wireless technology, e.g., Bluetooth® or Bluetooth® low energy (BLE). In some embodiments, the payment module 150 restricts electronic payment transactions between the user 102 and the merchant 106 to situations where the geographic location of the user device 104 is within a threshold geographic distance from a geographic location of the merchant device 108.

[0036] The computing devices 112 can also be configured to communicate with one or more computing devices 158 of a card payment network (e.g., MasterCard®, VISA®) over the one or more networks 110 to conduct financial transactions electronically. The computing devices 112 can also communicate with one or more bank computing devices 160 of one or more banks over the one or more networks 110. For example, the computing devices 112 may communicate with an acquiring bank, and/or an issuing bank, and/or a bank maintaining user accounts for electronic payments.

[0037] An acquiring bank may be a registered member of a card association (e.g., Visa®, MasterCard®), and may be part of a card payment network. An issuing bank may issue payment cards to users, and may pay acquiring banks for purchases made by cardholders to which the issuing bank has issued a payment card. Accordingly, in some examples, the computing device(s) of an acquiring bank may be included in the card payment network and may communicate with the computing devices of a card-issuing bank to obtain payment. Further, in some examples, the user may use a debit card or gift card instead of a credit card, in which case, the bank computing device(s) of a bank or other institution corresponding to the debit card or gift card may receive communications regarding a transaction in which the user is participating. Additionally, there may be computing devices of other financial institutions involved in some types of transactions or in alternative system architectures, and thus, the foregoing are merely several examples for discussion purposes. In addition, the merchant device(s) 108 may perform interactions similar to those described above with regard to the computing devices 158 of a card payment network and the bank computing devices 160 when processing transactions for payment instruments that do not involve the payment service of the computing devices 112.

[0038] The user 102 operating the user device 104 that is within a threshold geographic distance of the merchant device 108 can interact with an application executed on the user device 104 to conduct an electronic payment transaction with the merchant 106. While interacting with the application, the user 102 can select the merchant 106, from a listing of merchants 106, with whom the user wants to enter into an electronic payment transaction. The user 102 can select the merchant 106, for example, by selecting a “check in” option associated with the merchant 106. The user device 104 can communicate data to the computing devices 112 indicating that the user 102 has checked in with the merchant 106. In response, the

computing devices 112 can communicate data to notify the merchant device 106 that the user has checked in. An application executing on the merchant device 108 can notify the merchant 106 that the user has electronically checked in with the merchant 106 through a display of the merchant device 108.

[0039] Once checked in, the user 102 can receive, obtain or request items, services or appointments that are available to be acquired from the merchant 106. When the user 102 is ready to enter into the card-less payment transaction, the user 102 can, for example, approach a point of sale for the merchant 106 and identify him or herself. For example, the user 102 can verbally notify the merchant 106 that the user 102 wants to enter into a card-less payment transaction and can provide the merchant 106 with the user's name. The merchant 106 can then interact with the application executing on the merchant's device to select the user 102, from a listing of users that have checked in with the merchant 106, to initiate an electronic payment transaction for the item(s) being acquired by the user 102. For example, the merchant 106 can determine a total amount to charge the user for the item(s) being acquired. The user can verbally approve the total amount to be paid and, in response, the merchant 106 can submit a request for an electronic payment transaction for the total amount of the transaction to the computing devices 112. In response, the computing devices 112 can obtain, for example, from the user account information 154, data describing a financial account associated with the electronic purchase account of the user 102 to which the total amount will be charged.

[0040] The computing devices 112 can then communicate with the computing device 158 of a card payment network to complete an electronic payment transaction for the total amount to be charged to user's electronic payment account. Once the electronic payment transaction is complete, the computing devices 112 can communicate data describing the electronic payment for the transaction to the user device 104, e.g., as an electronic receipt, which can, for example, notify the user 102 of the total amount charged to the user for the electronic payment for the transaction with the particular merchant. Further, while a mobile user device 104 is described in this example for purposes of explanation, additional or alternative types of devices may be used in other examples.

[0041] The payment module 150 of the computing devices 112 and the POS module 138 of the merchant devices 108 may include synchronization functionality for the open ticket data to enable the offline mode. In the examples discussed herein, the synchronization functionality is provided by using version vectors.

[0042] A version vector is a mechanism for tracking changes to data in a distributed system, where multiple agents might update the data at different times. The version vector allows the devices to determine if one update preceded another (happened-before), followed it, or if the two updates happened concurrently (and therefore might conflict with each other). In this way, version vectors enable tracking among data replicas (i.e. versions of open tickets).

[0043] In the update rules for version vectors discussed herein; replica open tickets can either experience local updates (e.g., the merchant device adding an item to the local version of the open ticket on the merchant device) or can synchronize with another replica open ticket.

[0044] In operation, the device creating the open ticket also creates a version vector which initially includes one or more counters set to zero. Each counter corresponds to a device operating within the system, and, in some implementations, the vector also includes a label for each counter to identify the corresponding device. In some implementation, the version vector includes a counter for all devices in the system at initiation. In other implementations, the version vector includes a counter for those devices that have had interactions that included the open ticket associated with the vector.

[0045] In operation, each time a replica open ticket experiences a local update event, the device performing the local update increments the counter of the local vector corresponding to the device by one. In other words, each time a merchant updates the local version of the local ticket stored on the merchant device, the counter for the device within the version vector is incremented on the local device.

[0046] Each time two devices synchronize open tickets, the synchronizing devices both set the counters in the local version vectors to the maximum of the counter across both vectors. After synchronization, the two open tickets have identical version vectors.

[0047] Pairs of replica open tickets can be compared by inspecting their version vectors and may be determined to be either: identical, concurrent, or ordered. In the case that one or both of the vectors contain counters associated with device(s) not included in the other vector, the vector missing the counter is assumed to include the counter with a value of zero (i.e. the device associated with that counter has made zero changes to the history of the version of the open ticket having the zero counter or missing counter).

[0048] The ordered relation is one in which a first vector is an ancestor of a second vector. Such an ancestor/descendent relation exists if and only if every counter of the first vector is less than or equal to the corresponding counter in the second vector and at least one of the counters is strictly less than the corresponding counter in the second vector. If neither vector is an ancestor of the other, but the vectors are not identical, then the two vectors must be concurrent (from divergent branches). In the case that the open tickets being synchronized are concurrent versions, the system may reconcile the differences automatically or flag the open ticket for manual intervention.

[0049] As will be discussed in more detail with regard to FIGS. 2A-5, the merchant devices 108 and/or the payment service of the computing devices 112 may implement such synchronization functionality depending on the networking structure. As would be understood by one of ordinary skill in the art, the above discussed synchronization method is merely examples and implementations are not so limited. Many other synchronization mechanisms such as Lamport clocks would be apparent to one of ordinary skill in the art in view of this disclosure.

[0050] As mentioned above, the operations of modules 122-124, 136-138 and 146-150 may vary depending on functionality provided by the particular implementation. As such, the implementations are not limited to the example provided above. Additional details and example functionalities of the user device(s) 104, merchant device(s) 108 and the computing devices 112 as a whole are discussed below with regard to
5 FIGS. 2A-5.

[0051] FIGS. 2A-2B are example illustrations that provides context for the discussion of various scenarios with respect to FIGS. 3A-4C. In particular, FIGS. 2A and 2B are an illustration of two scenarios 200 and 250 occurring at a restaurant in which open ticket transactions including offline mode operations. In general, the scenario is as follows.

10 [0052] Four groups of customers are seated at the restaurant at respective tables 202-208. Merchants 210-214 are waiting on the customers of the tables 202-208. Each of the merchants 210-214 has a respective merchant device, MD A 216, MD B 218 and MD C 220. The merchant devices receive input from the merchants regarding open ticket transactions. For example, the merchant 210 may create an open ticket for items requested by the customers 202 using the merchant device A 216. The merchant devices
15 216-220 operate to connect to the access point 222 when the devices are within the range of the access point 224. While connected, the merchant devices 216-220 may communicate with each other through a LAN network provided by the access point 222 and/or a payment service (not shown) such as that provided by computing devices 112. While outside of the range of the access point 224, the merchant devices 216-220 operate in an off-line mode. While in the off-line mode, the off-line merchant device may not be able to
20 synchronize open ticket transaction data with the other merchant devices and/or the payment service. As such, changes to the local version of the open ticket may cause the open ticket to branch or diverge from the version of the open ticket maintained by the other merchant devices and/or the payment service.

[0053] As stated previously, in some implementations, the merchant devices may operate together and/or with a payment service to implement the off-line mode functionality such that a plurality of merchant
25 devices may alter respective local versions of the same open ticket and then synchronize the altered local version such that differences are reconciled.

[0054] In the particular example illustrated in FIG. 2A, the merchant device A 216 is operating in an online mode while merchant device B 218 and merchant device C 220 are in an off-line mode. As such, in implementations in which a payment service operates to synchronize differences between local versions of
30 open tickets stored by the merchant devices, the open ticket transaction data on merchant device A 216 may be synchronized with the payment service as changes occur. On the other hand, changes to open tickets occurring at merchant device B 218 and merchant device C 212 may not be synchronized because they are outside the range of the access point 224. When an offline device, such as merchant device B 218 or merchant device C 212, enters the range of the access point 224 and reconnects to the access point 222,
35 changes to the local versions stored thereon may be synchronized with the payment service.

[0055] The following example scenario illustrates a divergence in open ticket transaction data that may need to be reconciled by the system. At the start of the scenario, the merchant 214 creates an open ticket for the customers at table 206 using the merchant device C 220. The open ticket transaction data is then synchronized with the payment service after the merchant device C entered the range of the access point 224. Through the payment service, the open ticket data for table 206 is synchronized to the merchant device B 218 while merchant device B is within range of the access point 224. Subsequently, the merchant 214 returns to the table 206 to take the customers' order. Because the merchant device C 220 is outside the range of the access point 224, the modifications to the local version of the open ticket associated with table 206 by the merchant device C 220 may not be synchronized with the payment service. Next, at some point prior to the merchant device C 220 reentering the range of the access point 224, the merchant device B 218 exits the range of the access point 224 and goes into the offline mode. At this point, the merchant device B 218 has a local version of the open ticket for table 206 that does not include the changes made at merchant device C 220.

[0056] After the merchant 214 has left table 206, a customer at the table 206 asks merchant 212 to add one or more items to the open ticket for table 206. The merchant 212 adds the item to the local version of the open tickets stored at merchant device B 218. As such, two divergent local versions exist in the system, one with the items added by merchant 214 and the other including the items added by merchant 212. When the merchant device B 218 subsequently reenters online mode, the payment service may synchronize the payment service's local version of the open ticket with the local version stored in merchant device B 218. Similarly, when the merchant device C 220 reenters the online mode, the payment service may synchronize the payment service's local version of the open ticket with the local version stored on merchant device C 220 such that the version stored at the payment service includes all of the items requested by the customers.

[0057] Although discussed in the context of wait staff of a restaurant, the system of FIG. 2A is not so limited and may be extended or apply to other scenarios. For example, an extended scenario of FIG. 2A may include additional staff that deliver items to customers (e.g. pizza delivery or other item delivery). In such a scenario, the customer may call the restaurant and place an order. An open ticket may be created and distributed among devices of the merchant by the payment service via a Wi-Fi network operating at the restaurant. In this way, the merchant device of the delivery person may be provided with a copy of the open ticket. Once the items are ready for delivery, the delivery person may leave the restaurant to make the delivery. While en route, the merchant device of the delivery person may enter an offline mode. However, using the system described above, either the delivery person or the restaurant may make changes to the open ticket. For example, the customer may contact the restaurant to add an item (e.g. a beverage) to the order. The restaurant may make this change to the open ticket. The merchant may understand that the open ticket of the delivery person will not be updated in offline mode. Assuming the delivery person is carrying the additional item anyway, the merchant may not take further action because the delivery person will be able to

provide the beverage upon arrival. When the delivery person arrives and the customer requests the beverage, the delivery person may update the local version of the open ticket and provide the additional item. When the delivery person returns to the restaurant, the different versions of the open ticket may be reconciled using the techniques disclosed herein.

5 [0058] FIG. 2B includes a similar illustration of another example scenario 250 according to this disclosure. In particular, FIG. 2B illustrates a scenario at the a restaurant that differs from scenario 200 in that an additional merchant 252 is included who has a wide area network (WAN) enabled merchant device D 254. In addition, each of the merchant devices 216-220 and 254 is operating in an ad hoc networking mode. As shown, shown, each merchant device has an ad hoc range associated there with. In particular, merchant
10 device A 216 is associated with ad hoc range A 256, merchant device B 218 is associated with ad hoc range B 258, merchant device C 220 is associated with ad hoc range C 260 and merchant device D 254 is associated with ad hoc range D 262. Rather than or in addition to synchronizing through a payment service as described with regard to FIG. 2A, the merchant devices 216-220 and 254 synchronize using peer-to-peer networking. Further, in some implementations, rather than or in addition to communicating with the
15 payment service through the access point 222, the merchant devices 216-220 may communicate with the payment service through a shared Internet connection provided by the merchant device D 254. In other implementations, the merchant device D 254 may be the only merchant device that communicates directly with the payment service and the payment service may not include synchronization functionality for the off-line mode operations; e.g., all synchronization functionality is handled within the context of the peer-to-peer
20 network by merchant devices 216-220 and 254. In still other implementations, the merchant devices 216-220 may communicate through a shared Internet connection provided by the merchant device D 254 and through the access point 222. An example of the operation of the devices of FIG. 2B is provided by FIGS. 4A-4C.

[0059] The scenarios provided above are merely examples and the scope of this disclosure is not limited to such scenarios. For example, the devices in the scenario 200 may communicate using a peer to
25 peer networking scheme through the LAN provided by the access point 222. Also, the scenario 250 may also include an access point 222 and the communication mechanisms described with regard to scenarios 200 and 250 may be combined.

[0060] In a particular example of such a scenario, the system illustrated in FIG. 2B may be utilized
in, for example, a pop-up or farmer's market restaurant. In an example of such a utilization, a manager or
30 owner operator may have a mobile device with Wi-Fi and cellular connectivity (e.g. similar to merchant device D 254). At the same time, the merchant devices 216-220 may be devices operated by one or more waiters or staff. The manager or owner operator may have his employees bring their own devices for work. In such a scenario, the employees may desire to leave cellular data turned off to save money on their personal phone bills. In such a case, the devices the waiters brought from home may obtain internet access through
35 the device of the manager or owner operator. Alternative, the manager or owner operator may provide Wi-

Fi-only devices for his employees to operate. In either case, as discussed above regarding FIG. 2B, each time a waiter device enters WiFi range of the manager or owner operator device, the waiter device may synchronize with the server. Alternatively, the manager or owner operator device may synchronize with the waiter device then synchronize with the server.

5 [0061] Many other variations and examples based on the scenarios shown in FIGS. 2A and 2B would be apparent to one of skill in the art in view of this disclosure. In view of this general description of the scenario illustrated in FIGS. 2A-2B, various techniques for performing the associated tasks for open ticket payment handling with an offline mode will be discussed with regard to FIGS. 3A-4C.

[0062] The example processes are described in the context of the environment of FIG. 1 but are not
10 limited to those environments. Each process described in this disclosure is illustrated as a logical flow graph, each operation of which represents a sequence of operations that can be implemented in hardware, software, or a combination thereof. In the context of software, the operations represent computer-executable instructions stored on one or more computer-readable media or embodied as one or more computer transmission media that, when executed by one or more processors, perform the recited operations.
15 Generally, computer-executable instructions include routines, programs, objects, components, data structures, and the like that perform particular functions or implement particular abstract data types.

[0063] The computer-readable media may include non-transitory computer-readable storage media, which may include hard drives, floppy diskettes, optical disks, CD-ROMs, DVDs, read-only memories (ROMs), random access memories (RAMs), EPROMs, EEPROMs, flash memory, magnetic or optical cards,
20 solid-state memory devices, or other types of storage media suitable for storing electronic instructions. In some implementations, the computer transmission media may include a transitory computer-readable signal (in compressed or uncompressed form). Examples of computer-readable signals, whether modulated using a carrier or not, include, but are not limited to, signals that a computer system hosting or running a computer program can be configured to access, including signals downloaded through the Internet or other networks.
25 Finally, the order in which the operations are described is not intended to be construed as a limitation, and any number of the described operations can be combined in any order and/or in parallel to implement the process.

[0064] FIGS. 3A-3B collectively illustrate a sequence diagram 300 illustrating an example open ticket transaction in a point-of-sale (POS) context that is executed by a payment service and a plurality of
30 merchant devices which have offline mode functionality. The process of FIGS. 3A-3B is discussed in the context of the FIG. 2A described above. Also, the process shown in FIG. 3A-3B utilizes version vectors to handle synchronization among the various devices shown in FIG. 3A-3B. However, these details are merely to aid in comprehension and the process may be utilized in many other contexts and may utilize other synchronization mechanisms.

[0065] At 302, the merchant inputs a request to merchant device A 216 to create an open ticket for the customers of a table. For example, the merchant may tap an “open ticket” button displayed by a UI to instruct the merchant device A 216 to create an open ticket. In response, the merchant device A 216 may create a data record to maintain information regarding the open ticket. In addition, the merchant device A 216 may create a data record for a vector associated with the open ticket or may include the vector in the data record of the open ticket. As shown in the example of FIG. 3A, once the open ticket and vector have been created, the vector includes one entry, “A : 1”. In some implementations, the merchant device A 216 creates the vector with one entry, “A : 0” and then updates the entry to “A : 1” to represent that the merchant device A 216 has made one change to the open ticket. In this case, the one change is the creation of the ticket and/or the taking of initial item requests.

[0066] At 304, when the merchant device A 216 is within the network range and therefore able to operate in an online mode, the merchant device synchronizes the open ticket and the vector for the open ticket transaction with the payment service operated by the computing devices 112. In this synchronization operation, because the payment service does not have a copy of the open ticket being synchronized, the payment service may be considered to have a vector including the entry “S : 0; A : 0”. Because this is an ancestor of the vector “S : 0; A : 1” that the merchant device A 216 is considered to include under the version vector rules discussed above, the open ticket stored by the merchant device A 216 is copied to the payment service and both adopt the vector “S : 0; A : 1”.

[0067] At 306, the payment service synchronizes the open ticket and vector with the merchant device B 218. This synchronization is performed in a similar manner to that described 304 and results in each storing the vector “S : 0; A : 1; B : 0”. Similarly, at 308, the payment service synchronizes the open ticket and vector with the merchant device C 220. This synchronization is also performed in a similar manner to that described 304 and results in each storing the vector “S : 0; A : 1; B : 0; C : 0”. In practical terms, the operations at 306 and 308 may represent sequential synchronizations performed with devices that are in online mode with the payment service following the synchronization performed at 304.

[0068] At 310, the merchant device B 218 enters off-line mode, for example, by leaving the range of the access point 224. Subsequently, while in the off-line mode, the merchant device B 218 receives input of one or more first items requested by the customers associated with the open ticket. In response, the merchant device B 218 updates its local version of the open ticket to include the first items and increments the counter of the vector associated with device B by one. As a result, the vector stored at merchant device B 218 is updated to “S : 0; A : 1; B : 1”.

[0069] At 312, the merchant device C 220 enters off-line mode, for example, by leaving the range of the access point 224. In the illustrated example of FIG. 3A, the merchant device C 220 enters the off-line mode prior to the merchant device B 218 reentering online mode. While in the off-line mode, the merchant device C 220 receives input of one or more second items requested by the customers associated with the

open ticket. In response, the merchant device C 220 updates in its own local version of the open ticket to include the second items and increments the counter of the vector associated with device C by one. As a result, the vector stored at merchant device C 220 is updated to “S : 0; A : 1; B : 0, C: 1”. This mirrors the example provided above with regard to FIG. 2A in which a customer associated with the open ticket asks a second waiter for additional items after the first waiter has already taken a request for items and walked away. At this point, two divergent versions of the open ticket exist, one stored by merchant device B 218 and one stored by merchant device C 220. The process then continues in FIG. 3B.

[0070] At 314, the merchant device C 220 is illustrated as having returned to online mode, for example, by entering the range of the access point 224. In the illustrated example of FIG. 3B, upon entering the online mode, the merchant device C initiates a synchronization with the payment service. As a result of synchronizing the open ticket and vector with the payment service, the open ticket stored by the payment service is determined to be an ancestor of the open ticket stored by the merchant device C 220 and the payment service could be considered to adopt the open ticket and the vector of the merchant device C 220.

[0071] At 316, the payment service determines that the open ticket has been updated. As such, the payment service initiates synchronization with the other merchant devices that are online. In this case, merchant device A 216 is online and the synchronization is performed.

[0072] At 318, the merchant device B 218 is illustrated as having returned to online mode, for example, by entering the range of the access point 224. In the illustrated example of FIG. 3B, upon entering the online mode, the merchant device B 218 initiates a synchronization with the payment service. At this point in the process, the merchant device B 218 includes the vector “S : 0; A : 1; B : 1” and the payment service includes the vector “S : 0; A : 1; B : 0, C: 1”, because, under the previously discussed update rules of version vectors, neither vector is an ancestor of the other and the vectors are not identical, the synchronization process determines the open tickets associated with the vectors to be concurrent versions. Depending on the implementation, the synchronization process may resolve the differences in the concurrent versions of the open ticket using an automated process and/or by flagging the open ticket for manual review. For example, if the second items included in the version of the open ticket stored by the payment service are distinct from the first items included in the version of the open ticket stored by the merchant device B 218, the synchronization process may automatically add the first items to the version stored by the payment service and add the second items to the version stored by merchant device B. For example, if the first items were a coke and fries and the second items were a beer and hamburger, no risk of duplication is apparent and such an automatic process may be used. In other scenarios, manual intervention may be requested. For example, if the first items were a beer and fries and the second items were also a beer and fries that are distinct based on the time and device at which the second items were entered, there is a risk that the second items are a duplicate request by the customers for the first items. Such a situation may occur if the customer believes that the first request was forgotten and makes a second request to another waiter based on that

belief. Where the risk of duplication exists, the synchronization process may flag the open ticket for manual review by, for example, one or more of the waiters associated with merchant devices B 218 and C 220. For example, the synchronization process may cause the merchant device B 218 to display a user interface requesting the waiter associated with the device determine if the items are duplicates or represent requests for distinct items. As previously stated, the vectors are both updated to include the maximum value of the counters of the two vectors. As such, the merchant device B and payment service each include the vector “S : 0; A : 1; B : 1, C: 1”.

[0073] At 320, following the update of the payment service version of the open ticket, the payment service initiates a synchronization with the other online merchant devices (i.e. merchant device A and merchant device C). As both these synchronizations are of ancestor and descendent versions of the open ticket and vector, the synchronizations may be performed in a manner similar to the operations at 304. As such, the merchant device A and merchant device C each store the version of the open ticket stored by the payment service and include the vector “S : 0; A : 1; B : 1, C: 1”.

[0074] At 322, the payment service receives a request to close the open ticket, for example, from one of the merchant devices. In response, the payment service checks with the merchant devices to determine if the version of the open ticket stored by the payment service is fresh. Depending on the implementation, the payment service may check with those devices that have been involved in the history of the version vector stored by the payment service or the payment service may check with all known devices. In the case that a merchant device is an off-line mode and the payment service is unable to check with the device for freshness, the payment service may wait for the device to enter online mode or request authorization from the merchant device initiating the close ticket request.

[0075] Once freshness is ensured or authorization is received to proceed, the payment service processes the transaction associated with the open ticket. For example, the computing devices 112 of the payment service may interact with the card payment network computing devices 158 and/or bank computing devices 158 to complete the financial transaction.

[0076] The sequence of operations described above is only an example provided for discussion purposes. Numerous variations are possible. Additional example variations are provided below in the discussion of FIGS. 4A-4C and FIG. 5.

[0077] FIGS. 4A-4C collectively illustrate a sequence diagram 300 illustrating an example open ticket transaction in a point-of-sale (POS) context that is executed by a payment service and a plurality of merchant devices which have offline mode functionality. The process of FIGS. 4A-4C differs from that of FIGS. 3A-3B in that the merchant devices also perform peer-to-peer synchronization as previously discussed in the context of the FIG. 2B. In particular, FIGS. 4A-4C illustrate interaction between the payment service of the computing devices 112, the merchant device A 216, the merchant device B 218 and the WAN enabled merchant device D 254. In the particular example illustrated in FIGS. 4A-4C, the merchant devices A 216

and B 218 may communicate with the payment service via a LAN connection provided by the access point 222 or through a shared WAN connection provided by merchant device D 254. As with FIGS. 3A-3B, the process shown in FIG. 4A-4C utilizes version vectors to handle synchronization among the various devices shown in FIG. 4A-4C. However, these details are merely to aid in comprehension and the process may be utilized in many other contexts and may utilize other synchronization mechanisms.

[0078] At 402, the merchant inputs a request to merchant device A 216 to create an open ticket for the customers of a table and enters one or more first items requested by the customers. For example, the merchant may tap an “open ticket” button displayed by a UI to instruct the merchant device A 216 to create an open ticket. Further, the merchant may utilize a UI to enter the one or more first items. In response, the merchant device A 216 may create a data record to maintain information regarding the open ticket and add the one or more first items to the data record. In addition, the merchant device A 216 may create a data record for a vector associated with the open ticket or may include the vector in the data record of the open ticket. As shown in the example of FIGS. 4A, once the open ticket and vector have been created the vector includes one entry, “A : 1”. In some implementations, the merchant device A 216 creates the vector with one entry, “A : 0” and then updates the entry to “A : 1” to represent that the merchant device A 216 has made one change to the open ticket. In this case, the one change is the creation of the ticket and the adding of the first item to the open ticket. However, implementations are not so limited. In other implementations, these operations may be considered to be separate and the counter for device A may instead be incremented twice.

[0079] At 404, the merchant device A 216 and the merchant device B 218 pass into ad hoc networking range. Upon detecting the ad hoc connection, the devices initiate a synchronization of the open ticket data stored by the merchant devices. Because the devices determine that the merchant device B does not include the open ticket, the merchant devices synchronize the open ticket and the vector for the open ticket transaction. In this synchronization operation, because the merchant device B 218 does not have a copy of the open ticket being synchronized, the merchant device B 218 may be considered to have a vector including the entry “A : 0; B : 0”. Because this is an ancestor of the vector “A : 1; B : 0” that the merchant device A 216 is considered to include under the version vector rules discussed above, the open ticket stored by the merchant device A 216 is copied to the merchant device B 218 and both adopt the vector “A : 1; B : 0”.

[0080] At 406, the merchant device A 216 and the merchant device D 254 pass into ad hoc networking range. Upon detecting the ad hoc connection, the devices initiate a synchronization of the open ticket data stored by merchant devices A 216 and D 254. This synchronization process may be performed in a similar manner to that discussed above with regard to the operations at 404. As a result, the merchant devices A and D each store the version of the open ticket previously stored by the merchant device A and include the vector “D : 0; A : 1; B : 0”.

[0081] At 408, the merchant device D 254 enters WAN service range. Upon detecting the connection to the payment service of the computing devices 112 via the WAN connection, the merchant device D 254 initiates synchronization of the open ticket and vector with the payment service. This synchronization process may be performed in a similar manner to that discussed above with regard to the operations at 404. As a result, the merchant devices D and the payment service each store the version of the open ticket previously stored by the merchant device D and include the vector “S : 0; D : 0; A : 1; B : 0”.

[0082] At 410, the merchant device B 218 passes out of the range of the other merchant devices and the access point, entering an off-line mode. Subsequently, while in the off-line mode, the merchant device B 218 receives input of one or more second items requested by the customers associated with the open ticket. In response, the merchant device B 218 updates its local version of the open ticket to include the second items and increments the counter of the vector associated with device B by one. As a result, the vector stored at merchant device B 218 is updated to “A : 1; B : 1”.

[0083] At 412, the merchant device B 218 and the merchant device D 254 pass into ad hoc networking range. Upon detecting the ad hoc connection, the devices initiate a synchronization of the open ticket data stored by merchant devices B 218 and D 254. This synchronization process may be performed in a similar manner to that discussed above with regard to the operations at 404. As a result, the merchant devices A and D each store the version of the open ticket previously stored by the merchant device B and include the vector “S : 0; D : 0; A : 1; B : 1”. The process then continues to FIG. 4B.

[0084] At 414, the merchant device A 216 passes out of the range of the other merchant devices and the access point, entering an off-line mode. While in the off-line mode, the merchant device A 216 receives input of one or more third items requested by the customers associated with the open ticket which includes one or more duplicates of the second items. In response, the merchant device A 216 updates its local version of the open ticket to include the third items and increments the counter of the vector associated with device A by one. As a result, the vector stored at merchant device A 216 is updated to “D : 0; A : 2; B : 0”.

[0085] At 416, the merchant device A 216 is illustrated as having entered the range of the access point 224 thereby entering an online mode and gaining connection to the payment service. In the illustrated example of FIG. 4B, upon entering the online mode with the payment service, the merchant device A initiates a synchronization with the payment service through the LAN connection with the access point 222. As a result of synchronizing the open ticket and vector with the payment service, the open ticket stored by the payment service is determined to be an ancestor of the open ticket stored by the merchant device A 216 and the payment service could be considered to adopt the open ticket of the merchant device A 216 and each updates its vector to “S : 0; D : 0; A : 2; B : 0”.

[0086] At 418, the merchant device A 216 and the merchant device B 218 pass into ad hoc networking range. Upon detecting the ad hoc connection, the devices initiate a synchronization of the open ticket data stored by the merchant devices. However, the attempted synchronization of the open ticket and

vector may fail due to the conflict between the second items and third items which were previously stated to include duplicates of the second items. At 420, because there is a risk of duplicate items, the merchant device A 216 and the merchant device B 218 notify the merchants of the duplicate item risk between the second items and the third items. At 422, the merchant device A 216 receives input from the merchant to resolve the conflict. Based on the input, the merchant device A 216 and the merchant device B 218 synchronize their respective open tickets to include the items not indicated as being duplicates by the merchant input and update their vectors to “S : 0; D : 0; A : 2; B : 1”.

[0087] At 424, the merchant device D 254 passes out of the range of the other merchant devices, the access point and the WAN service, entering an off-line mode. Subsequently, while in the off-line mode, the merchant device D 254 receives input of one or more fourth items requested by the customers associated with the open ticket. In response, the merchant device D 254 updates its local version of the open ticket to include the fourth items and increments the counter of the vector associated with device D by one. As a result, the vector stored at merchant device D 254 is updated to “S : 0; D : 1; A : 1; B : 1”. The process then continues to FIG. 4C.

[0088] At 426, the merchant device A 216 and the merchant device D 254 pass into ad hoc networking range. Upon detecting the ad hoc connection, the devices initiate a synchronization of the open ticket data stored by the merchant devices. As discussed above, the vectors stored by the merchant device A 216 and the merchant device D 254 are concurrent vectors. As such, the synchronization process may attempt and automatic reconciliation of the differences in the tickets or may request manual intervention. In this case, the fourth items are not duplicates of the third items and automatic reconciliation is possible. In addition, information about the previous manual intervention performed at 422 may be maintained by the version stored at merchant device A 216 to allow for automatic reconciliation in the future for the third items. Once the reconciliation is performed, the vectors of both devices are updated to “S : 0; D : 1; A : 2; B : 1”.

[0089] At 428, merchant device D 254 enters WAN service and synchronizes the open ticket and vector with the payment service. This synchronization is between an ancestor and descendent and therefore may be performed as discussed above.

[0090] At 430, the merchant device A 216 receives a request to close the open ticket. In response, the merchant device A 216 causes a check-in with the other merchant devices to ensure the freshness of the open ticket stored at merchant device A 216. At 432, because the merchant device A 216 did not receive responses to the freshness check-in requests from all the merchant devices, the merchant device A 216 prompts the merchant of merchant device A 216 for authorization to proceed without information from one or more of the other merchant devices. For example, merchant device B 218 may have passed out of the range of the other merchant devices, the access point and the WAN service, entering an off-line mode. As such, the check-in with device B will fail due to the lack of connection. As such, the merchant device A 216

may request authorization to proceed without check-in confirmation with device B. If granted, the process moves to 434.

[0091] At 434, once freshness of the version stored at merchant device A is ensured or authorization is received to proceed without freshness responses from one or more other merchant devices, the merchant device A 216 sends a request to the payment to process the transaction associated with open ticket. The payment service may execute the transaction in the manner discussed above.

[0092] The sequence of operations described above is only an example provided for discussion purposes. Numerous variations are possible.

[0093] FIG. 5 is a flow diagram illustrating an example process 500 for handling open ticket transactions including the offline mode functionality disclosed herein. The following actions described with respect to FIG. 5 may be performed by the merchant device(s) 108 and/or the payment service of the computing devices 112 as shown in FIG. 1 and may be used in the context of FIGS. 2-4C.

[0094] At 502, the synchronization process receives a trigger event associated with an open ticket transaction. In the example illustrated in FIG. 5, three general types of trigger events are shown, a modify open ticket trigger, a synchronization trigger and a complete transaction trigger. At 504, the synchronization process determines the type of event. If the trigger event is a modify ticket trigger, the process continues to 506. If the trigger event is a synchronization trigger, the process continues to 508. If the trigger event is a complete transaction trigger, the process continues to 510.

[0095] At 506, the synchronization process implements the modification to the open ticket. In the restaurant example, the modify ticket request may include a request to add items to the ticket or a request to add payment instruments to the open ticket. The process then continues to 512.

[0096] At 512, the synchronization process increments the current device's counter in the local vector associated with the open ticket. This may be performed in the manner discussed above with regard to FIGS. 1-4C. The process then continues to 514.

[0097] At 514, the synchronization process determines whether the current device is operating in an online mode (i.e. if the current device is connected to any other device that, in the particular implementation, the current device performs synchronizations with). If the current device is in an online mode, the process continues to 508. Otherwise, if the current device is in an off-line mode, the process returns to 502 and awaits a new trigger event.

[0098] Returning to 508, which may be reached from 504 in response a synchronization trigger or from 514 in response to a determination that the current device is in an online mode, the synchronization process determines the other devices with which to synchronize. Depending on the implementation, various events may be considered synchronization event triggers. Some examples include returning to online mode (e.g. by gaining a general network connection or a peer-to-peer connection), synchronization with another device that results in an update of the local vector while connected to other devices and a refresh timer which

triggers synchronization on a regular basis. The determination of the devices to synchronize with may vary based on the role of the current device executing the synchronization process and the networking structure of the implementation. For example, the payment service in an implementation such as that shown in FIG. 2A attempts to synchronize with all online devices other than a device with whom a synchronization has changed the local vector stored by the payment service. On the other hand, the merchant devices in FIG. 2A may only synchronize with the payment service in some implementations. On the other hand, the merchant devices in the implementation illustrated in FIG. 2B may attempt to synchronize with both the payment service and any other merchant device which is in connection with the current device. After determining which other devices to synchronize with, the process then continues to 516.

10 [0099] At 516, the synchronization process exchanges vectors with a current synchronization partner device and determines the relationship between the open tickets based on the vectors. At 518, if the vectors are related as ancestor and descendent, the synchronization process may operate to cause both devices keep the descendent version of the open ticket. At 520, if the vectors are concurrent, the synchronization process may attempt an automatic reconciliation of the open ticket changes. If the automatic
15 reconciliation is not successful or the synchronization process does not handle the type of conflict that is present, the synchronization process may prompt the merchant for manual intervention. Once the open tickets are synchronized, the process continues to 522.

[0100] At 522, the synchronization process determines if any devices remain with which to synchronize. If so, the process continues to 524. Otherwise, the process returns to 502.

20 [0101] At 524, the synchronization process determines a new synchronization partner and the process returns to 516 to synchronize with the new synchronization partner.

[0102] Returning to 510, which may be reached from 504 in response to a complete transaction trigger event, the synchronization process attempts to determine whether the local version of the open ticket is fresh (i.e. accounts for all changes throughout the system). In some implementations, the synchronization process of the current device may attempt to communicate with all devices in the system or those devices which are known to have “touched” the open ticket. In some scenarios, the synchronization process may be unable to check with all the other devices (e.g. the other device may be offline). In such a case, the synchronization process may prompt the merchant to obtain authorization to proceed without checking in with one or more of the other devices with which it cannot communicate. The process then continues to 526.

30 [0103] At 526, the synchronization process may cause the transaction to be closed and completed. For example, the payment service may be caused to communicate with the card payment network computing devices 158 and/or the bank computing devices 160 to process payments for the cost of the transaction from any payment instruments associated with the open ticket transaction.

[0104] The process 500 described above is only an example provided for discussion purposes.
35 Numerous variations are possible.

[0105] As previously stated, each of the above discussed scenarios is merely an example and many variations are possible. Moreover, many variations of the techniques discussed above are possible as well without departing from the scope of this disclosure. For example, the techniques and systems disclosed herein may be utilized in contexts other than open ticket transactions. In a particular variation, the techniques and systems disclosed herein may be utilized in the context of inventory synchronization between merchant devices in a manner similar to that discussed above.

[0106] Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described. Rather, the specific features and acts are disclosed as illustrative forms of implementing the claims.

Clauses Section

[0107] Clause 1, one or more computer-readable media storing instructions executable by one or more processors, wherein the instructions program the one or more processors to perform acts comprising: receiving, by a first point-of-sale (POS) device of a plurality of POS devices of a merchant, a first input of a first customer order associated with a transaction between the merchant and one or more customers, the first POS device configured with a cellular data connection and a WiFi connection; storing the first customer order as an open ticket with an associated versioning data structure in the memory of the first POS device; receiving, by a second POS device of the plurality of POS devices of the merchant from the first POS device while the first POS device and second POS device are within WiFi range of each other, a copy of the open ticket and a copy of the associated versioning data structure, the second POS device configured without a cellular data connection and with a WiFi connection; receiving, by the second POS device, a second input of a second customer order associated with the one or more customers while the first POS device and the second POS device are in an offline mode with respect to one another in which the first POS device and the second POS device are out of WiFi range of each other; updating, by the second POS device, the copy of the open ticket and associated versioning data structure based on the second input while the first POS device and second POS device are in the offline mode with respect to one another; receiving, by the first POS device from the second POS device, the updated copy of the open ticket and the updated copy of the associated versioning data structure when the first POS device and second POS device are in an online mode with respect to one another in which the first POS device and the second POS device are back in WiFi range and have reconnected; reconciling, by the first POS device, the open ticket with the updated copy of the open ticket based on the associated versioning data structure and the updated copy of the associated versioning data structure to generate a reconciled open ticket; receiving, by the payment service from the first POS device at least partly via the cellular data connection, the reconciled open ticket; receiving, by the payment

service from the first POS device at least partly via the cellular data connection, a request to process payment for a cost of the transaction between the merchant and the one or more customers; and processing the payment for the cost of the transaction between the merchant and the one or more customers based on the reconciled open ticket.

5 [0108] Clause 2, the one or more computer-readable media as recited in clause 1, wherein the instructions program the one or more processors to further perform acts comprising: receiving, by the first POS device, a third input of a third customer order associated the one or more customers prior to the first POS device and second POS device reconnecting; and updating, by the first POS device, the open ticket with the associated versioning data structure in the memory of the first POS device based on the third input, and
10 wherein the reconciling comprises: determining one or more items of the third customer order match one or more of the items of the second customer order; requesting the merchant to provide fourth input regarding whether the one or more items of the third customer order are duplicate requests by the one or more customers for the one or more items of the second customer order; and generating the reconciled open ticket based on the fourth input received from the merchant.

15 [0109] Clause 3, the one or more computer-readable media as recited in clause 1 or clause 2, wherein the instructions program the one or more processors to further perform acts comprising: receiving, by the first POS device, a third input of a third customer order associated the one or more customers prior to the first POS device and second POS device reconnecting; and updating, by the first POS device, the open ticket with the associated versioning data structure in the memory of the first POS device based on the third
20 input, and wherein the reconciling comprises: determining one or more items of the third customer order do not match one or more items of the second customer order; generating the reconciled open ticket such that the reconciled open ticket includes at least the one or more items of the second order and the one or more items of the third order that do not match.

[0110] Clause 4, the one or more computer-readable media as recited in any of clauses 1-3, wherein
25 the instructions program the one or more processors to further perform acts comprising: determining that the open ticket in the memory of the first POS device is an ancestor of the updated copy of the open ticket received from the second POS device based on the associated versioning data structure in the memory of the first POS device and the updated copy of the associated versioning data structure received from the second POS device; and wherein the reconciling is based at least the determination that the that the open ticket in the
30 memory of the first POS device is an ancestor of the updated copy of the open ticket received from the second POS device.

[0111] Clause 5, the one or more computer-readable media as recited in any of clauses 1-4, wherein the instructions program the one or more processors to further perform acts comprising: receiving, by the first POS device, a third input of a third customer order associated the one or more customers prior to the
35 first POS device and second POS device reconnecting; and updating, by the first POS device, the open ticket

with the associated versioning data structure in the memory of the first POS device based on the third input, and determining that the open ticket in the memory of the first POS device and the updated copy of the open ticket received from the second POS device are concurrent versions based on the associated versioning data structure in the memory of the first POS device and the updated copy of the associated versioning data structure received from the second POS device; and wherein the reconciling is based at least the determination that the open ticket in the memory of the first POS device and the updated copy of the open ticket received from the second POS device are concurrent versions.

[0112] Clause 6, one or more non-transitory computer-readable media storing instructions executable by one or more processors, wherein the instructions program the one or more processors to implement a service to perform acts comprising: receiving a first version of an open ticket associated with a transaction between a merchant and one or more customers from a first point-of-sale (POS) device of a plurality of POS devices of the merchant, the first version of the open ticket including one or more first items requested by the one or more customers from the merchant; generating, by the service, a second version of the open ticket based on the first version of the open ticket such that the one or more first items are included in the second version of the open ticket; receiving a third version of the open ticket from a second POS device of the plurality of POS devices of the merchant including one or more second items requested by the one or more customers from the merchant, the second version of the open ticket generated while the second POS device was in an offline mode; and reconciling the second version of the open ticket and the third version of the open ticket.

[0113] Clause 7, the one or more non-transitory computer-readable media as recited in clause 6, wherein the reconciling of the second version of the open ticket and the third version of the open ticket comprises: determining that the third version of the open ticket is an ancestor of the second version of the open ticket; and maintaining the second version of the open ticket.

[0114] Clause 8, the one or more non-transitory computer-readable media as recited in clause 7, wherein the second version of the open ticket has an associated first vector and the third version of the open ticket has an associated second vector, the first vector and second vector including one or more counters corresponding to POS devices of the plurality of POS devices of the merchant in a history of the respective version of the open ticket; and wherein the third version of the open ticket is determined to be an ancestor of the second version of the open ticket based on the counters of the second vector being less than or equal to the corresponding counters of the first vector and at least one of the counters of the second vector being less than the corresponding counters of the first vector.

[0115] Clause 9, the one or more non-transitory computer-readable media as recited in any of clauses 6-8, wherein the reconciling of the second version of the open ticket and the third version of the open ticket comprises: determining that the second version of the open ticket is an ancestor of the third version of

the open ticket; and generating a fourth version of the open ticket to include the one or more first items and one or more second items.

[0116] Clause 10, the one or more non-transitory computer-readable media as recited in clause 9, wherein the reconciling of the second version of the open ticket and the third version of the open ticket comprises: receiving a fifth version of the open ticket from a third POS device of the plurality of POS devices of the merchant including one or more third items requested by the one or more customers from the merchant, the fifth version of the open ticket created while the third POS device was in an offline mode; and reconciling the fourth version of the open ticket and the fifth version of the open ticket, the reconciling comprising: determining one or more of the third items match one or more of the first items or one or more of the second items; requesting the merchant to provide input regarding whether the one or more third items that match the one or more of the first items or the one or more of the second items are duplicate requests by the one or more customers for the one or more of the first items or the one or more of the second items; and generating a sixth version of the open ticket based on the input received from the merchant.

[0117] Clause 11, the one or more non-transitory computer-readable media as recited in clause 10, wherein the fourth version of the open ticket has an associated first vector and the fifth version of the open ticket has an associated second vector, the first vector and second vector including one or more counters corresponding to POS devices of the plurality of POS devices of the merchant in a history of the respective version of the open ticket; and wherein the fourth version of the open ticket is determined to be a concurrent version of the fifth version of the open ticket based on at least one counter of the first vector being greater than the corresponding counter of the second vector and at least one counter of the first vector being less than the corresponding counter of the second vector.

[0118] Clause 12, the one or more non-transitory computer-readable media as recited in any of clauses 6-11, wherein the instructions program the one or more processors to further implement the service to perform acts comprising: receiving a fifth version of the open ticket from a third POS device of the plurality of POS devices of the merchant including one or more third items requested by the one or more customers from the merchant, the fifth version of the open ticket created while the third POS device was in an offline mode; and reconciling the fourth version of the open ticket and the fifth version of the open ticket, the reconciling comprising: determining the one or more third items are distinct from the one or more first items and the one or more second items; generating a sixth version of the open ticket that includes the one or more first items, the one or more second items and the one or more third items.

[0119] Clause 13, the one or more non-transitory computer-readable media as recited in any of clauses 6-12, wherein the instructions program the one or more processors to further implement the service to perform acts comprising: receiving a request to close the open ticket and process payment for a cost of the open ticket; and attempting to communicate with the plurality of POS devices to verify that changes to the

open ticket have been reconciled in a current version stored by the service; processing the payment for the cost of the current version of the open ticket upon verifying the changes have been reconciled.

[0120] Clause 14, the one or more non-transitory computer-readable media as recited in any of clauses 6-13, wherein the instructions program the one or more processors to further implement the service to perform acts comprising: receiving a request to close the open ticket and process payment for a cost of the open ticket; and attempting to communicate with the plurality of POS devices to verify that changes to the open ticket have been reconciled in a current version stored by the service; requesting authorization from the merchant to process the payment without verification from one or more of the plurality of POS devices; processing the payment for the cost of the current version of the open ticket upon receiving the requested authorization.

[0121] Clause 15, one or more non-transitory computer-readable media storing instructions executable by one or more processors, wherein the instructions program the one or more processors to perform acts comprising: receiving, by a first point-of-sale (POS) device of a plurality of POS devices of a merchant, a first copy of a first version of an open ticket associated with a transaction between the merchant and one or more customers; receiving, by a second POS device of the plurality of POS devices of the merchant, a second copy of the first version of the open ticket; receiving, by the first POS device, a first input specifying one or more first items requested by the one or more customers from the merchant while the first POS device is in an offline mode; generating, by the first POS device, a second version of the open ticket by updating the first copy of the first version of the open ticket to include the one or more first items; receiving, by the second POS device, a second input specifying one or more second items requested by the one or more customers from the merchant while the second POS device is in an offline mode; generating, by the second POS device, a third version of the open ticket by updating the second copy of the first version of the open ticket to include the one or more second items; and reconciling, by one of the plurality of POS devices of the merchant, the second version of the open ticket and the third version of the open ticket.

[0122] Clause 16, the one or more non-transitory computer-readable media as recited in clause 15, wherein the reconciling comprises: determining one or more of the second items match one or more of the first items; requesting the merchant associated with the first POS device or the second POS device to provide input regarding whether the one or more second items that match the one or more of the first items are duplicate requests by the one or more customers for the one or more of the first items; and generating a fourth version of the open ticket based on the input received from the merchant.

[0123] Clause 17, the one or more non-transitory computer-readable media as recited in clause 16, further comprising: reconciling, by the first POS device, the first version of the open ticket and the fourth version of the open ticket, the reconciling comprising: determining that the first version of the open ticket is an ancestor of the fourth version of the open ticket; and generating a fifth version of the open ticket that is a copy of the fourth version of the open ticket.

[0124] Clause 18, the one or more non-transitory computer-readable media as recited in any of clauses 15-17, wherein the reconciling comprises: determining the one or more second items are distinct from the one or more first items; generating a fourth version of the open ticket that includes the one or more first items and the one or more second items.

5 [0125] Clause 19, the one or more non-transitory computer-readable media as recited in clause 18, wherein the second version of the open ticket has an associated first vector and the third version of the open ticket has an associated second vector, the first vector and second vector including one or more counters corresponding to POS devices of the plurality of POS devices of the merchant in a history of the respective version of the open ticket; and wherein the reconciling comprises: determining the second version of the open ticket to be a concurrent version of the third version of the open ticket based on at least one counter of
10 the first vector being greater than the corresponding counter of the second vector and at least one counter of the first vector being less than the corresponding counter of the second vector.

[0126] Clause 10, the one or more non-transitory computer-readable media as recited in any of clauses 15-19, wherein the one of the plurality of POS devices of the merchant reconciling the second
15 version of the open ticket and the third version of the open ticket is a wide area network (WAN) enabled device that connects to a payment service through the WAN and that connects to the at least one other POS device of the plurality of POS devices of the merchant through an ad hoc network connection.

20

CLAIMS

WHAT IS CLAIMED IS:

1. One or more computer-readable media storing instructions executable by one or more processors, wherein the instructions program the one or more processors to perform acts comprising:
- 5 receiving, by a first point-of-sale (POS) device of a plurality of POS devices of a merchant, a first input of a first customer order associated with a transaction between the merchant and one or more customers, the first POS device configured with a cellular data connection and a WiFi connection;
- 10 storing the first customer order as an open ticket with an associated versioning data structure in the memory of the first POS device;
- receiving, by a second POS device of the plurality of POS devices of the merchant from the first POS device while the first POS device and second POS device are within WiFi range of each other, a copy of the open ticket and a copy of the associated versioning data structure, the second POS device configured without a cellular data connection and with a WiFi connection;
- 15 receiving, by the second POS device, a second input of a second customer order associated with the one or more customers while the first POS device and the second POS device are in an offline mode with respect to one another in which the first POS device and the second POS device are out of WiFi range of each other;
- 20 updating, by the second POS device, the copy of the open ticket and associated versioning data structure based on the second input while the first POS device and second POS device are in the offline mode with respect to one another;
- receiving, by the first POS device from the second POS device, the updated copy of the open ticket and the updated copy of the associated versioning data structure when the first POS device and second POS device are in an online mode with respect to one another in which the first POS device and the second POS device are back in WiFi range and have reconnected;
- 25 reconciling, by the first POS device, the open ticket with the updated copy of the open ticket based on the associated versioning data structure and the updated copy of the associated versioning data structure to generate a reconciled open ticket;
- receiving, by the payment service from the first POS device at least partly via the cellular data connection, the reconciled open ticket;
- 30 receiving, by the payment service from the first POS device at least partly via the cellular data connection, a request to process payment for a cost of the transaction between the merchant and the one or more customers; and

processing the payment for the cost of the transaction between the merchant and the one or more customers based on the reconciled open ticket.

2. The one or more computer-readable media as recited in claim 1, wherein the instructions
5 program the one or more processors to further perform acts comprising:
receiving, by the first POS device, a third input of a third customer order associated the one or more customers prior to the first POS device and second POS device reconnecting; and
updating, by the first POS device, the open ticket with the associated versioning data structure in the memory of the first POS device based on the third input, and wherein the reconciling comprises:
10 determining one or more items of the third customer order match one or more of the items of the second customer order;
requesting the merchant to provide fourth input regarding whether the one or more items of the third customer order are duplicate requests by the one or more customers for the one or more items of the second customer order; and
15 generating the reconciled open ticket based on the fourth input received from the merchant.

3. The one or more computer-readable media as recited in claim 1 or claim 2, wherein the instructions program the one or more processors to further perform acts comprising:
receiving, by the first POS device, a third input of a third customer order associated the one or more
20 customers prior to the first POS device and second POS device reconnecting; and
updating, by the first POS device, the open ticket with the associated versioning data structure in the memory of the first POS device based on the third input, and wherein the reconciling comprises:
determining one or more items of the third customer order do not match one or more items of the second customer order;
25 generating the reconciled open ticket such that the reconciled open ticket includes at least the one or more items of the second order and the one or more items of the third order that do not match.

4. The one or more computer-readable media as recited in any of claims 1-3, wherein the instructions program the one or more processors to further perform acts comprising:
30 determining that the open ticket in the memory of the first POS device is an ancestor of the updated copy of the open ticket received from the second POS device based on the associated versioning data structure in the memory of the first POS device and the updated copy of the associated versioning data structure received from the second POS device; and

wherein the reconciling is based at least the determination that the that the open ticket in the memory of the first POS device is an ancestor of the updated copy of the open ticket received from the second POS device.

5 5. One or more non-transitory computer-readable media storing instructions executable by one or more processors, wherein the instructions program the one or more processors to implement a service to perform acts comprising:

 receiving a first version of an open ticket associated with a transaction between a merchant and one or more customers from a first point-of-sale (POS) device of a plurality of POS devices of the merchant, the
10 first version of the open ticket including one or more first items requested by the one or more customers from the merchant;

 generating, by the service, a second version of the open ticket based on the first version of the open ticket such that the one or more first items are included in the second version of the open ticket;

 receiving a third version of the open ticket from a second POS device of the plurality of POS devices
15 of the merchant including one or more second items requested by the one or more customers from the merchant, the second version of the open ticket generated while the second POS device was in an offline mode; and

 reconciling the second version of the open ticket and the third version of the open ticket.

20 6. The one or more non-transitory computer-readable media as recited in claim 5, wherein the reconciling of the second version of the open ticket and the third version of the open ticket comprises:

 determining that the third version of the open ticket is an ancestor of the second version of the open ticket; and

 maintaining the second version of the open ticket.

25 7. The one or more non-transitory computer-readable media as recited in claim 6, wherein the second version of the open ticket has an associated first vector and the third version of the open ticket has an associated second vector, the first vector and second vector including one or more counters corresponding to POS devices of the plurality of POS devices of the merchant in a history of the respective version of the open
30 ticket; and

 wherein the third version of the open ticket is determined to be an ancestor of the second version of the open ticket based on the counters of the second vector being less than or equal to the corresponding counters of the first vector and at least one of the counters of the second vector being less than the corresponding counters of the first vector.

35

8. The one or more non-transitory computer-readable media as recited in any of claims 5-7, wherein the reconciling of the second version of the open ticket and the third version of the open ticket comprises:

5 determining that the second version of the open ticket is an ancestor of the third version of the open ticket; and

generating a fourth version of the open ticket to include the one or more first items and one or more second items.

9. The one or more non-transitory computer-readable media as recited in claim 8, wherein the reconciling of the second version of the open ticket and the third version of the open ticket comprises:

10 receiving a fifth version of the open ticket from a third POS device of the plurality of POS devices of the merchant including one or more third items requested by the one or more customers from the merchant, the fifth version of the open ticket created while the third POS device was in an offline mode; and

15 reconciling the fourth version of the open ticket and the fifth version of the open ticket, the reconciling comprising:

determining one or more of the third items match one or more of the first items or one or more of the second items;

20 requesting the merchant to provide input regarding whether the one or more third items that match the one or more of the first items or the one or more of the second items are duplicate requests by the one or more customers for the one or more of the first items or the one or more of the second items; and

generating a sixth version of the open ticket based on the input received from the merchant.

10. The one or more non-transitory computer-readable media as recited in any of claims 5-9, wherein the instructions program the one or more processors to further implement the service to perform acts comprising:

receiving a request to close the open ticket and process payment for a cost of the open ticket; and

25 attempting to communicate with the plurality of POS devices to verify that changes to the open ticket have been reconciled in a current version stored by the service;

30 processing the payment for the cost of the current version of the open ticket upon verifying the changes have been reconciled.

11. One or more non-transitory computer-readable media storing instructions executable by one or more processors, wherein the instructions program the one or more processors to perform acts comprising:

receiving, by a first point-of-sale (POS) device of a plurality of POS devices of a merchant, a first copy of a first version of an open ticket associated with a transaction between the merchant and one or more customers;

receiving, by a second POS device of the plurality of POS devices of the merchant, a second copy of the first version of the open ticket;

receiving, by the first POS device, a first input specifying one or more first items requested by the one or more customers from the merchant while the first POS device is in an offline mode;

generating, by the first POS device, a second version of the open ticket by updating the first copy of the first version of the open ticket to include the one or more first items;

receiving, by the second POS device, a second input specifying one or more second items requested by the one or more customers from the merchant while the second POS device is in an offline mode;

generating, by the second POS device, a third version of the open ticket by updating the second copy of the first version of the open ticket to include the one or more second items; and

reconciling, by one of the plurality of POS devices of the merchant, the second version of the open ticket and the third version of the open ticket.

12. The one or more non-transitory computer-readable media as recited in claim 11, wherein the reconciling comprises:

determining one or more of the second items match one or more of the first items;

requesting the merchant associated with the first POS device or the second POS device to provide input regarding whether the one or more second items that match the one or more of the first items are duplicate requests by the one or more customers for the one or more of the first items; and

generating a fourth version of the open ticket based on the input received from the merchant.

13. The one or more non-transitory computer-readable media as recited in claim 12, further comprising:

reconciling, by the first POS device, the first version of the open ticket and the fourth version of the open ticket, the reconciling comprising:

determining that the first version of the open ticket is an ancestor of the fourth version of the open ticket; and

generating a fifth version of the open ticket that is a copy of the fourth version of the open ticket.

14. The one or more non-transitory computer-readable media as recited in any of claims 11-13, wherein the reconciling comprises:

determining the one or more second items are distinct from the one or more first items;

5 generating a fourth version of the open ticket that includes the one or more first items and the one or more second items.

15. The one or more non-transitory computer-readable media as recited in claim 14, wherein the second version of the open ticket has an associated first vector and the third version of the open ticket has an associated second vector, the first vector and second vector including one or more counters corresponding to
10 POS devices of the plurality of POS devices of the merchant in a history of the respective version of the open ticket; and

wherein the reconciling comprises:

determining the second version of the open ticket to be a concurrent version of the third
15 version of the open ticket based on at least one counter of the first vector being greater than the corresponding counter of the second vector and at least one counter of the first vector being less than the corresponding counter of the second vector.

100 →

1/9

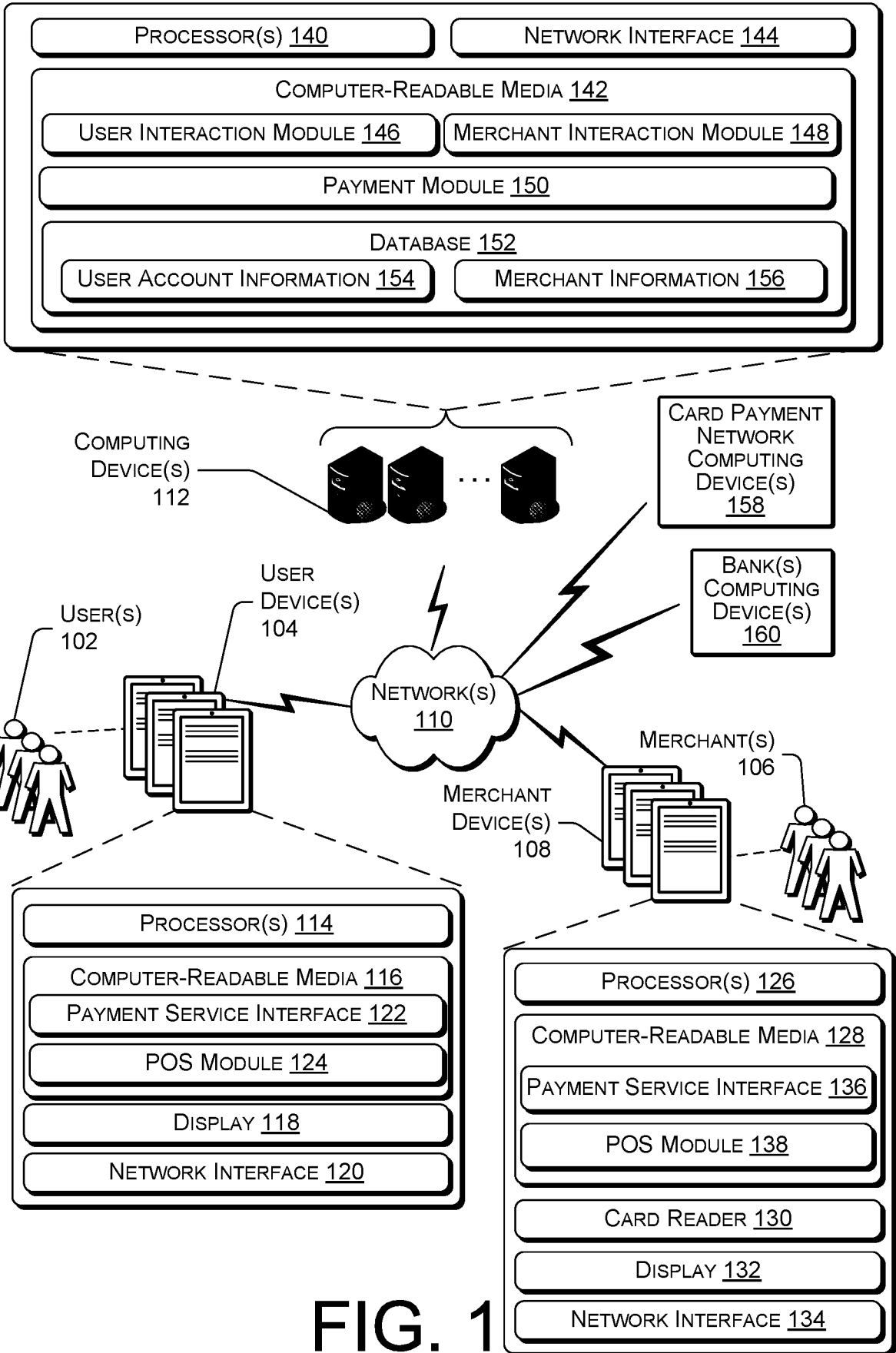


FIG. 1

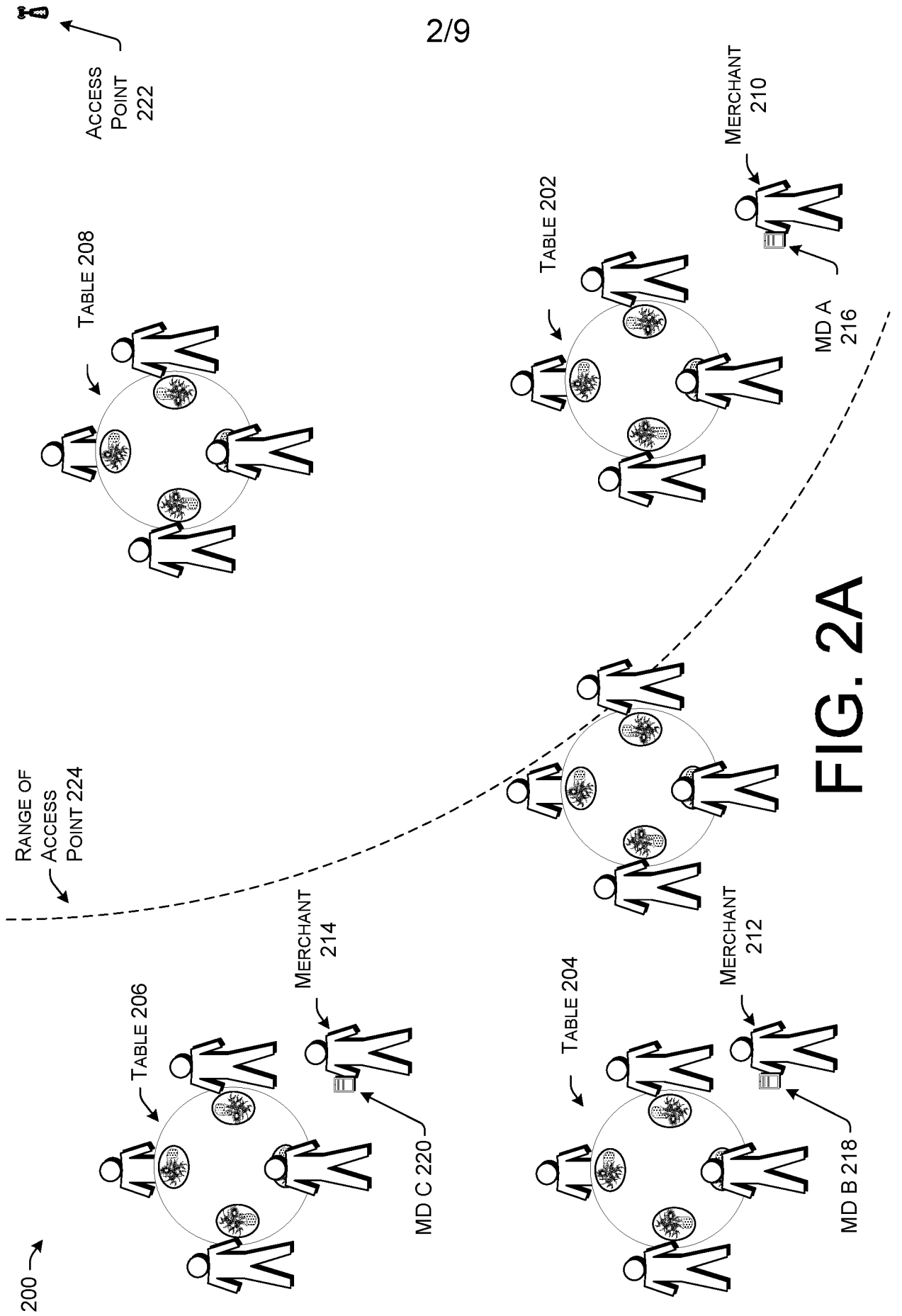
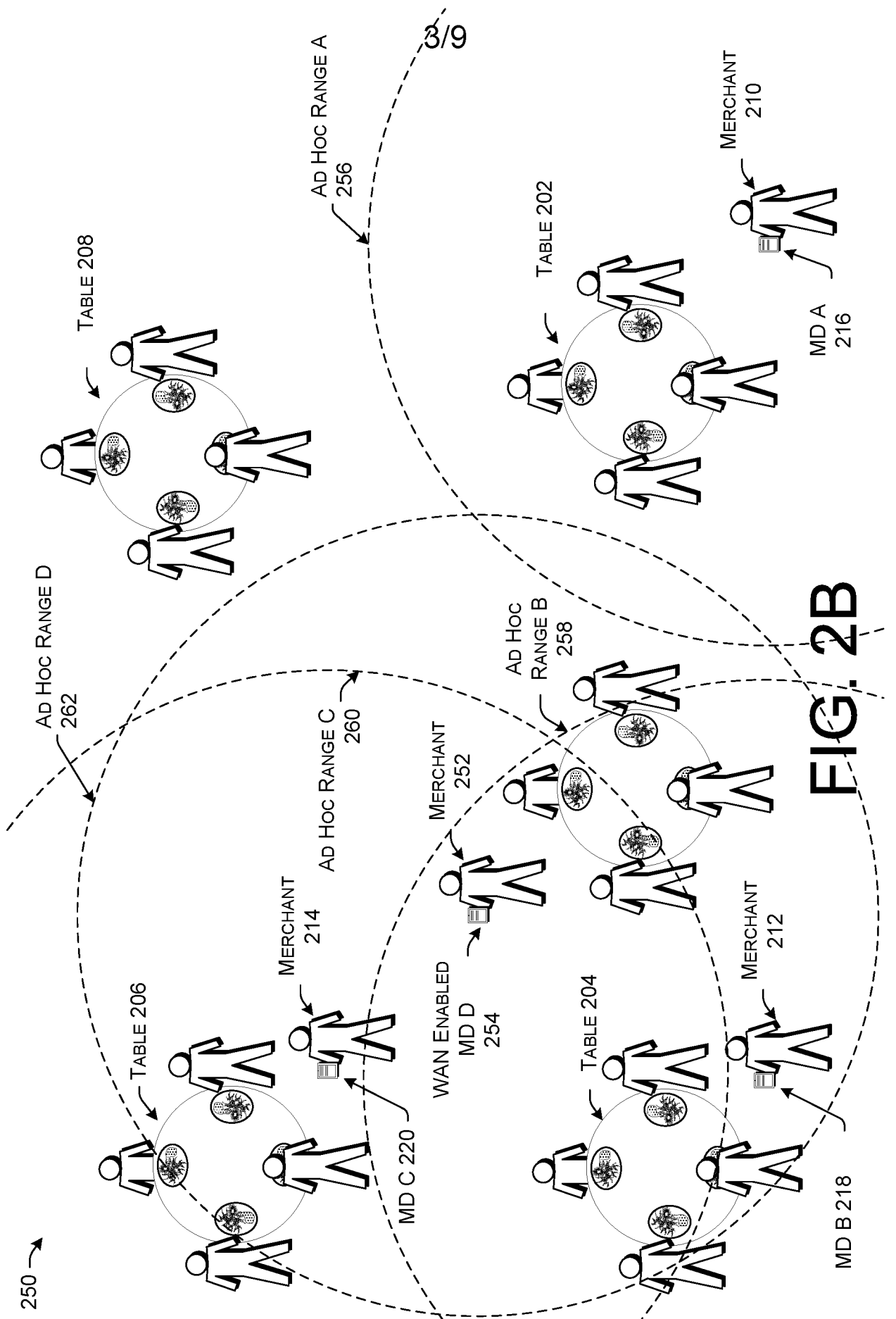
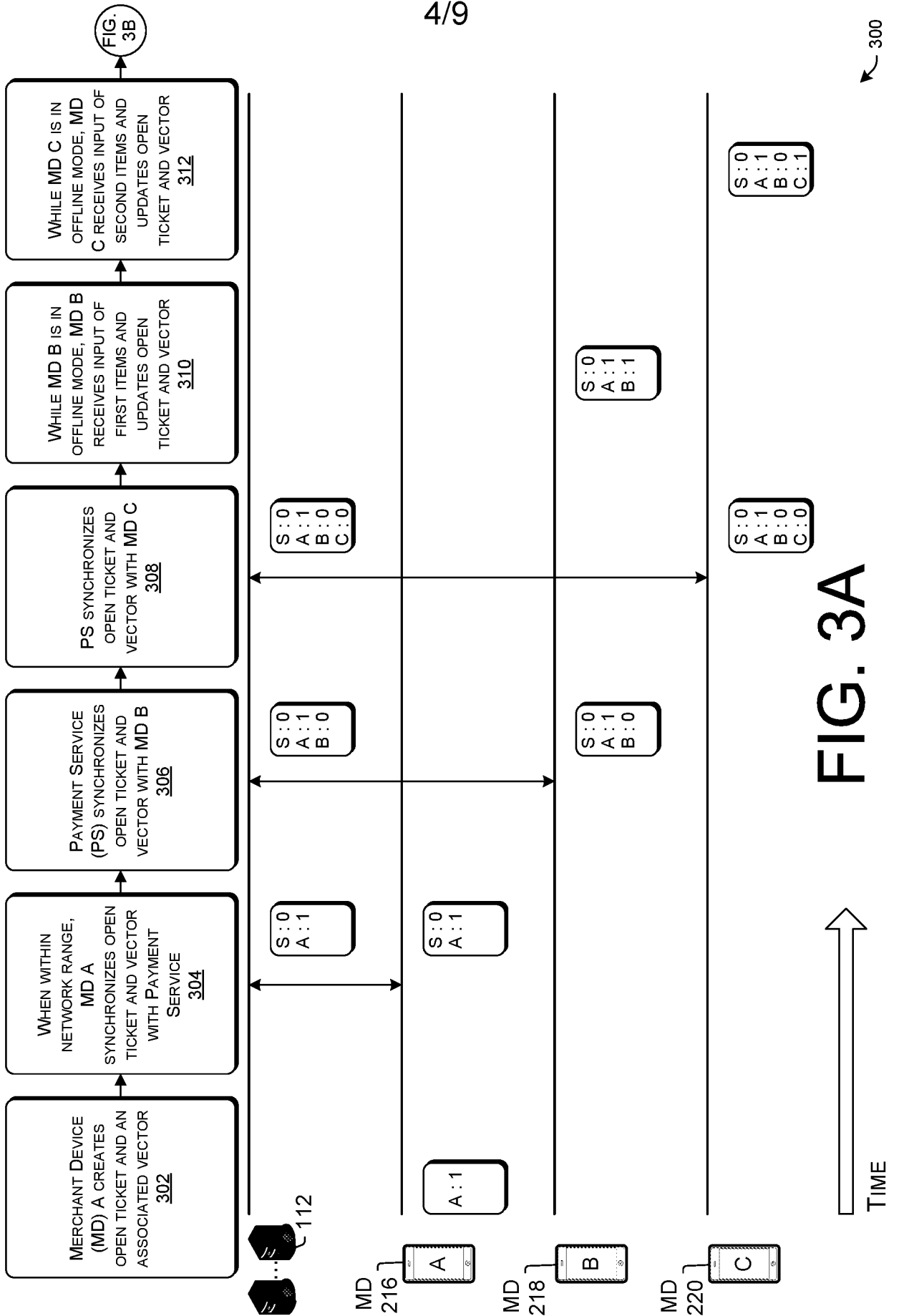


FIG. 2A





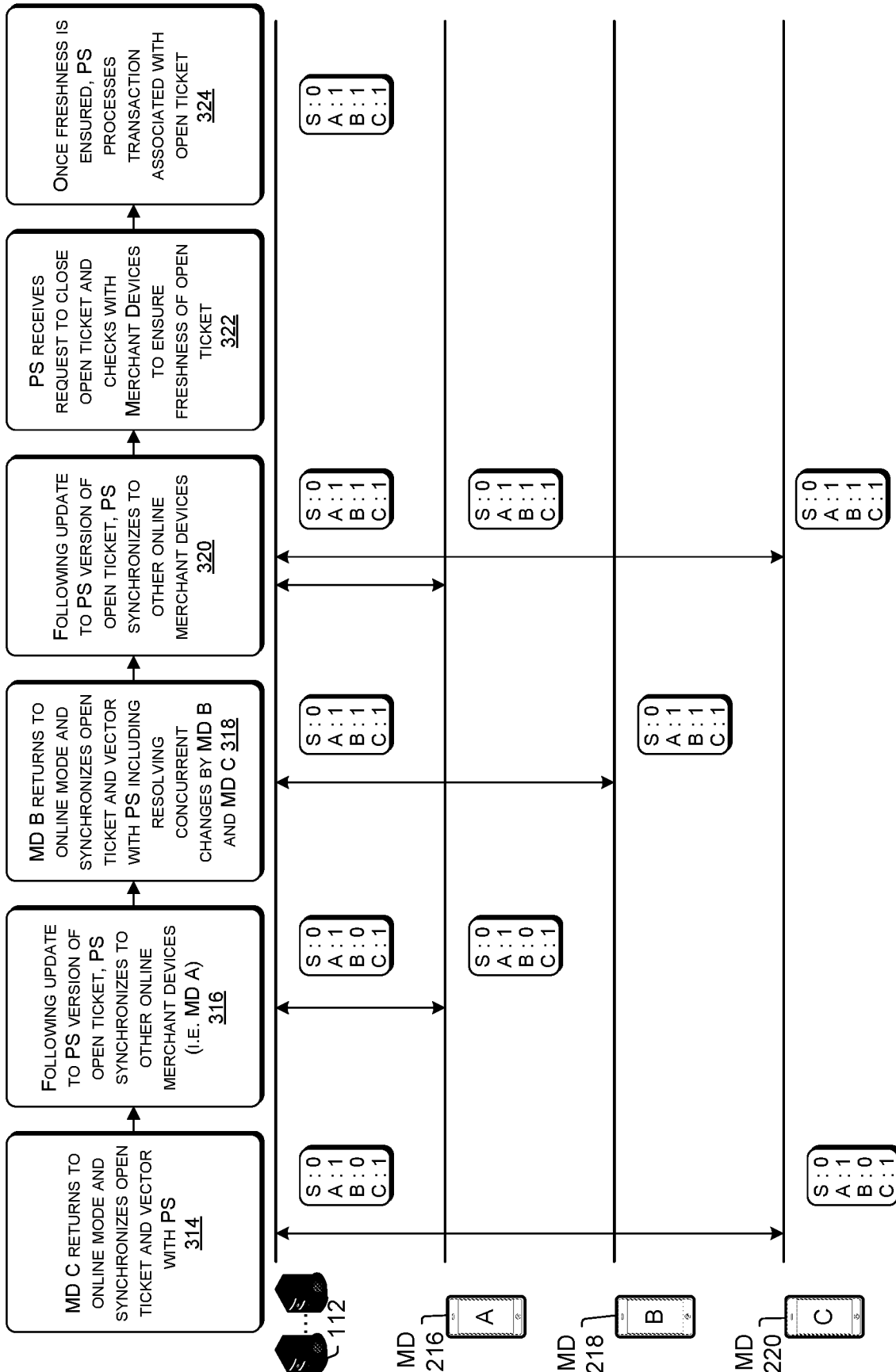
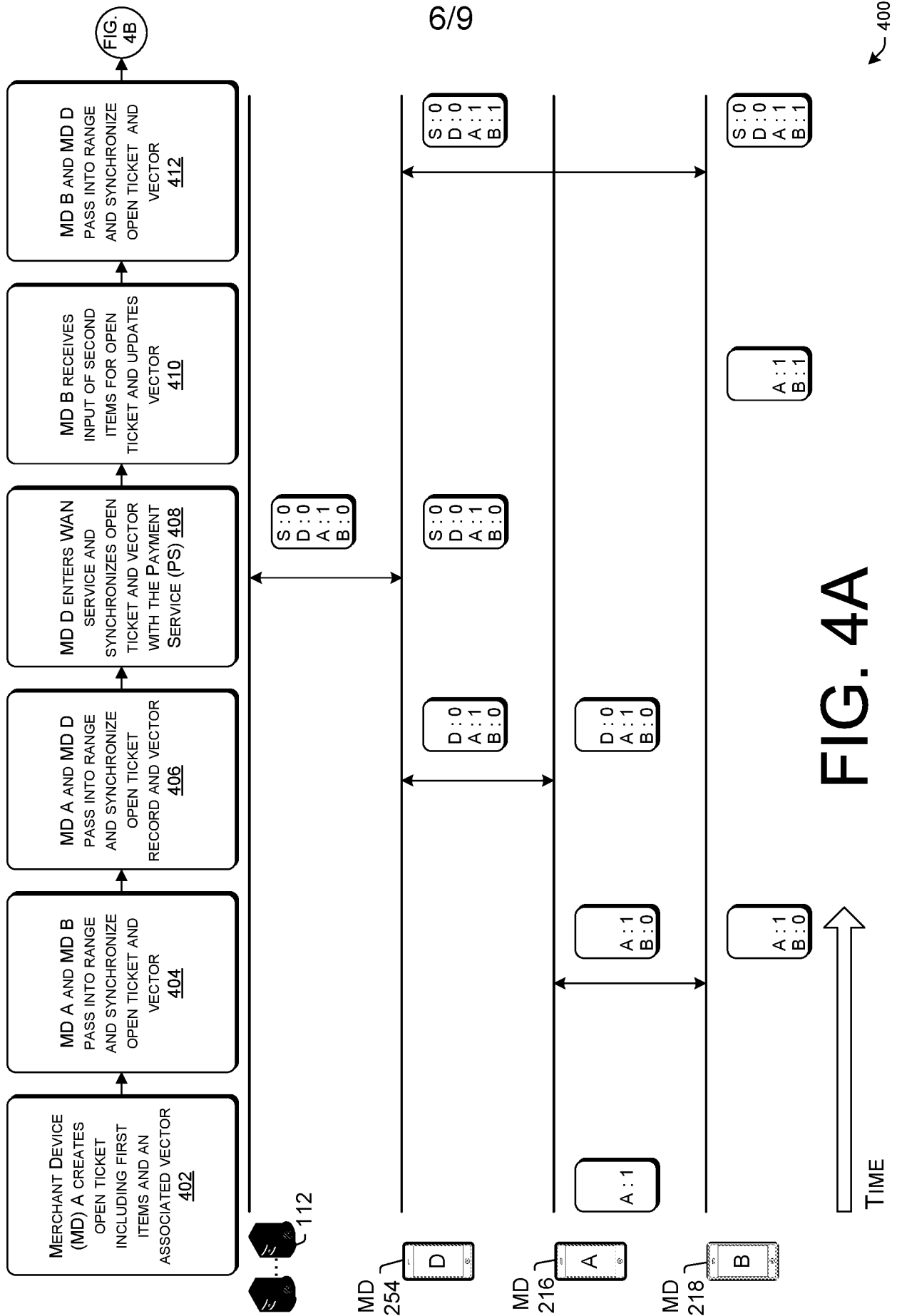


FIG. 3B

300



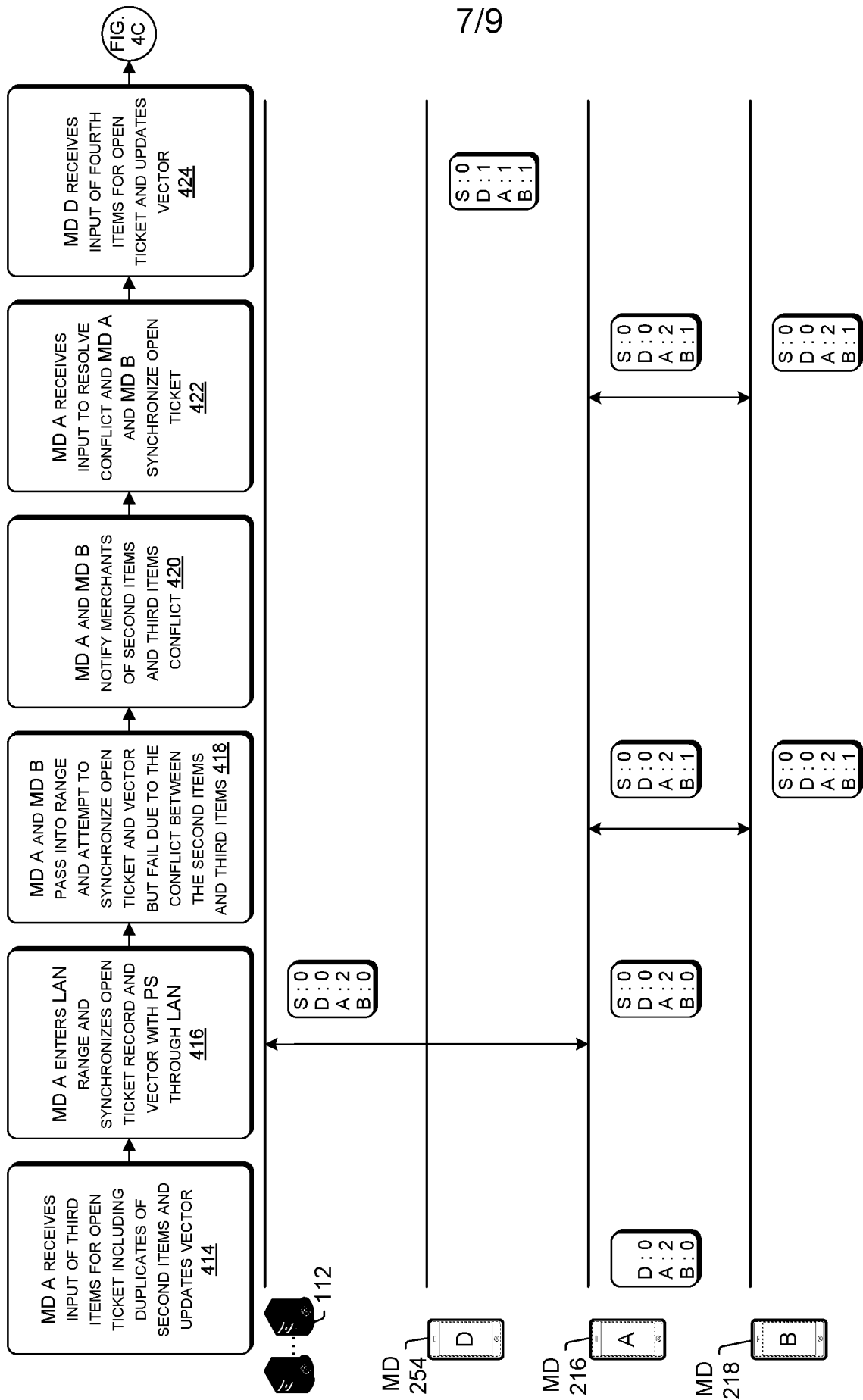


FIG. 4B

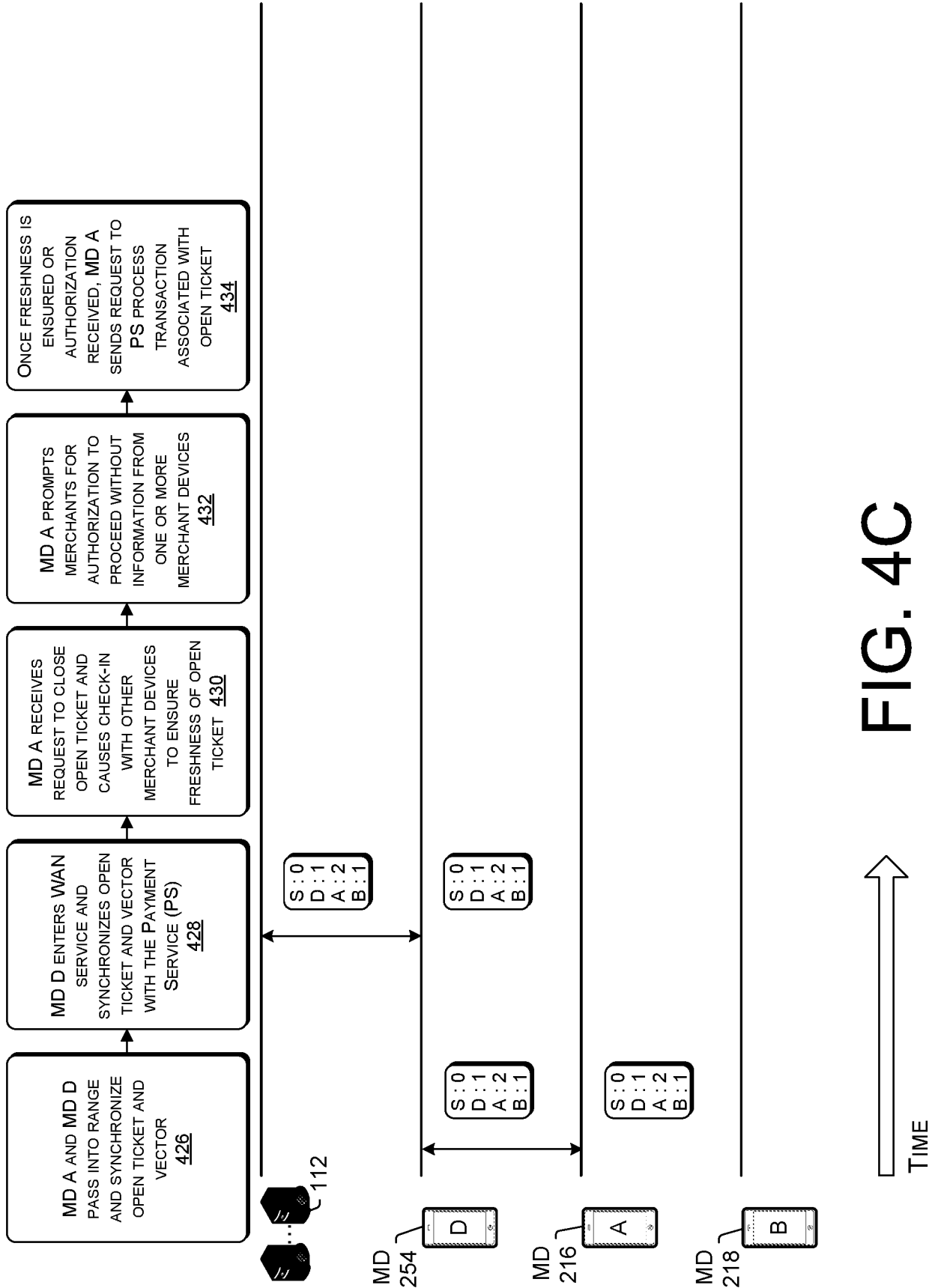


FIG. 4C

500 ↗

9/9

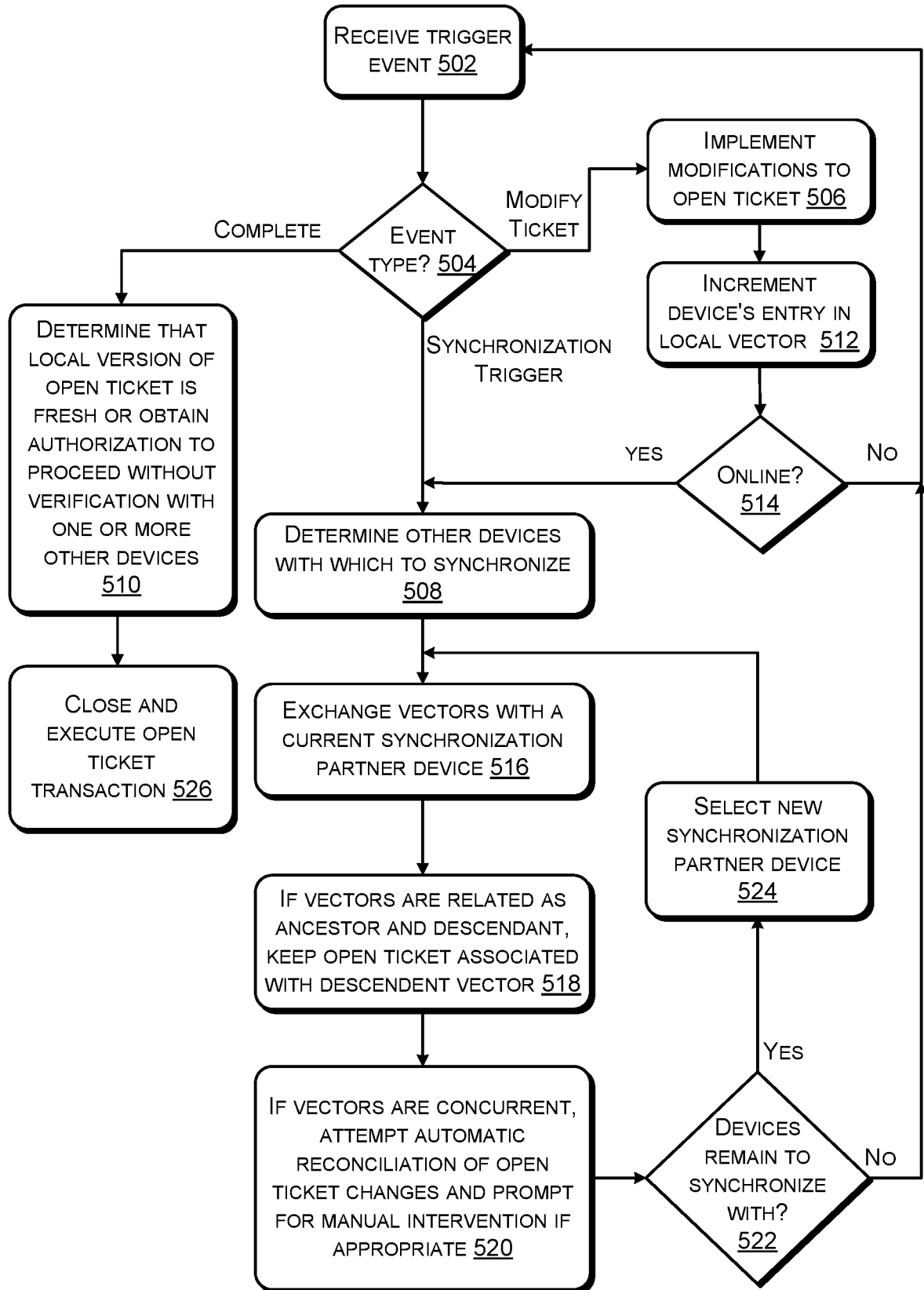


FIG. 5